UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Towards Reliable and Transparent Models of Language: A Study with Applications in Machine Translation

Nuno Miguel Serrano Guerreiro

Supervisor: Doctor André Filipe Torres Martins
Co-Supervisor: Doctor Pierre Jean Adrien Colombo

Thesis approved in public session to obtain the PhD Degree in
Electrical and Computer Engineering

Jury final classification: **Pass with Distinction and Honour**

2025

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Towards Reliable and Transparent Models of Language: A Study with Applications in Machine Translation

Nuno Miguel Serrano Guerreiro

Supervisor: Doctor André Filipe Torres Martins
Co-Supervisor: Doctor Pierre Jean Adrien Colombo

Thesis approved in public session to obtain the PhD Degree in
Electrical and Computer Engineering

Jury final classification: **Pass with Distinction and Honour**

Jury

Chairperson: Doctor Pedro Manuel Urbano de Almeida Lima, Instituto Superior Técnico, Universidade de Lisboa

Members of the Committee:
Doctor Luke Sean Zettlemoyer, Paul G. Allen School of Computer Science & Engineering, University of Washington, USA
Doctor Marine Carpuat, College of Computer, Mathematical, and Natural Sciences (CMNS), University of Maryland, USA
Doctor André Filipe Torres Martins, Instituto Superior Técnico, Universidade de Lisboa
Doctor Bruno Emanuel da Graça Martins, Instituto Superior Técnico, Universidade de Lisboa

2025

Abstract

Language technologies have progressed rapidly in recent years and are becoming evermore relevant, as computational models of language are adopted across a broad range of sectors of society. These technologies are reshaping how we communicate, process information, and make decisions on a global scale. In particular, machine translation, the key use case explored in this thesis, stands out as an application with widespread global usage, involving the processing billions of words daily across the world.

While this rapid progress has led to exciting opportunities, it has also created pressing challenges. As these technologies become increasingly accessible to millions of users worldwide, ensuring their reliability becomes paramount. Simultaneously, we face a growing *model understanding debt*: as models become larger and more complex, our ability to interpret their predictions, to accurately evaluate their performance, and to debug their failures is diminished. This creates a complex cycle where improved performance through scaling typically comes at the cost of reduced transparency, precisely when the widespread adoption of these technologies demands greater reliability and more sophisticated evaluation methods.

This thesis addresses these challenges by focusing on three main research directions: (i) understanding and safeguarding against "hallucinations"—critical failure modes—in translation systems; (ii) developing novel interpretable models of language and improved fine-grained evaluation metrics; and (iii) advancing the state of the art in machine translation through the development of systems with improved accuracy and reliability.

First, on the topic of hallucinations in machine translation, we present comprehensive studies on detection, analysis, and mitigation of these failures, by investigating classical bilingual and massively multilingual neural translation models, as well as more recent large language models. These studies include a wide range of efforts, from creating human-annotated datasets, comprehensively comparing detection methods, analysing the inner workings of models when they hallucinate, and developing novel on-the-fly mitigation techniques. Second, we investigate interpretable models and evaluation, by introducing new frameworks for extracting explanations in discriminative models and developing fine-grained evaluation metrics for machine translation. We propose the SPECTRA framework, a class of interpretable models that select a sparse set of important document parts that not only serve as explanations but also directly inform the model's decisions. This framework offers greater stability, easier training, and more flexibility in extracting explanations. Then, building on these interpretability ideas, we develop XCOMET, a new state-of-the-art metric for machine translation evaluation. XCOMET goes beyond providing a quality estimation through a single numerical value. The metric is designed to extract text segments with translation errors, along with the severity of these errors, and use this information directly to obtain a final evaluation. Finally, we set out to advancing translation capabilities and develop TOWER, a suite of multilingual large language models up to 70 billion parameters. TOWER models are optimised for translation-related tasks, and trained on carefully curated data and employing more robust quality-informed generation strategies. As a result, they outperform the best existing open-weight and commercial systems such as GPT-4 and Gemini-1.5-Pro across a wide range of language pairs, ranking first in the WMT 2024 General Machine Translation Shared Task according to automatic evaluation and best participating system in human evaluation.

Keywords: Natural language processing, machine translation, interpretability, translation quality evaluation, large language models.

Resumo

As tecnologias de linguagem têm progredido rapidamente nos últimos anos e estão a tornar-se cada vez mais relevantes com a crescente adoção de modelos computacionais de linguagem num vasto leque de sectores da sociedade. Estas tecnologias estão a remodelar a forma como comunicamos, processamos informação, e tomamos decisões a uma escala global. Em particular, a tradução automática, o caso de uso principal estudado nesta tese, destaca-se como uma aplicação com ampla utilização global, processando diariamente milhares de milhões de palavras em todo o mundo.

Embora este rápido progresso tenha proporcionado oportunidades empolgantes, também criou desafios significativos. Com a crescente acessibilidade global destas tecnologias, torna-se crucial garantir que são fiáveis. Simultaneamente, enfrentamos uma crescente *dívida de interpretabilidade* dos modelos: à medida que os modelos de linguagem se tornam maiores e mais complexos, a nossa capacidade para interpretar as suas previsões, avaliar o seu desempenho, e localizar e eliminar as suas falhas é diminuída. Este cenário cria um ciclo complexo em que melhorias no desempenho dos modelos através do aumento de escala vem, geralmente, à custa de uma redução da transparência. Ciclo este que ocorre precisamente quando a adopção generalizada destas tecnologias exige maior fiabilidade e métodos de avaliação mais sofisticados.

Esta tese aborda estes desafios concentrando-se em três direções principais de investigação: (i) compreensão e prevenção de alucinações—modos de falha críticos—em sistemas de tradução; (ii) desenvolvimento de novos modelos interpretáveis de linguagem e métricas de avaliação detalhadas melhoradas; e (iii) avanço do estado-da-arte em tradução automática através de sistemas mais precisos e fiáveis.

Em primeiro lugar, no âmbito das alucinações em tradução automática, apresentamos estudos abrangentes sobre deteção, análise e mitigação destas falhas, investigando modelos de tradução (bilingues e massivamente multilingues) e grandes modelos de linguagem. Estes estudos incluem a criação de dados anotados por humanos, comparação exaustiva de métodos de deteção, análise do funcionamento interno dos modelos ao gerar alucinações, e desenvolvimento de novas técnicas de mitigação de alucinações em tempo real. De seguida, investigamos modelos interpretáveis e avaliação, introduzindo novas estruturas para extrair explicações em modelos discriminativos, e desenvolvendo métricas de avaliação detalhadas para tradução automática. Propomos a estrutura SPECTRA, uma classe de modelos interpretáveis que selecionam conjuntos esparso de partes importantes de documentos que não só são usados como explicações, mas também informam diretamente as decisões do modelo. Esta estrutura oferece maior estabilidade, treino mais fácil, e maior flexibilidade na extração das explicações. Depois, baseando-nos nestas ideias de interpretabilidade, desenvolvemos o XCOMET, uma nova métrica de avaliação de tradução automática que atinge resultados ao nível do estado-da-arte. O XCOMET vai além de fornecer apenas uma estimação de qualidade através de um único valor numérico. A métrica é desenhada para extrair segmentos de texto com erros de tradução e identificar a severidade desses erros. Estas informações são depois usadas diretamente para obter uma avaliação final. Por fim, desenvolvemos o TOWER, uma família de grandes modelos de linguagem multilingues. O maior modelo tem uma capacidade de aproximadamente 70 mil milhões de parâmetros, e é otimizado para tarefas relacionadas com tradução automática. Estes modelos foram treinados em dados cuidadosamente selecionados e recorrem a estratégias de geração de texto mais robustas, informadas por estimativas de qualidade. Como resultado, o TOWER supera os melhores sistemas abertos e comerciais existentes, como o GPT-4 e o Gemini-1.5-Pro, numa ampla gama de pares de idiomas, obtendo o primeiro lugar na competição de Tradução Generalista da WMT 2024 de acordo com a avaliação automática, e sendo considerada a submissão vencedora de acordo com a avaliação humana.

Palavras-chave: Processamento de linguagem natural, tradução automática, interpretabilidade, avaliação automática de tradução, modelos de linguagem de grande escala.

Acknowledgments

Looking back on this journey that began in the midst of the 2020 pandemic, I can't help but smile and feel moved by what an incredible ride it has been. While PhDs are meant to be challenging, and mine certainly had its difficult moments, I was fortunate and privileged to share this path with extraordinary people—brilliant minds, mentors, friends, and family—who made all the difference. Here, I want to express my gratitude to them.

My first words go out to my supervisor, André Martins. I am deeply grateful for his guidance and care. André was instrumental throughout, from the very first to the very last day. At the start, he pushed me and helped me navigate the often treacherous beginning of a PhD. As I progressed and grew more confident, André gave me the space to carry my own research ideas, while consistently challenging me to go a step further. André leads with his insightful guidance, his vision, and, perhaps most importantly, his trust that we can do big things if we dare try. I'm also immensely grateful to my co-supervisor, Pierre Colombo. Pierre is a force of nature and an almost inexhaustible idea generator. He infused me with his enthusiasm for research, making our day-to-day collaboration a unique, fast-moving, and exciting experience. Working with Pierre day in and day out was one of the immense privileges I had during my studies. Through Pierre, I got a spark for research that I carry through in my projects to this day and that I hope I can transfer to everyone I work with. It is thanks to my supervisors and the environment they created that I was able to find my studies a truly enjoyable and fun experience. I am deeply conscious that this is a privilege, and I thank you for that.

Carrying out my studies has been, if nothing else, an immensely humbling experience. In this, there has not been a single biggest contributor but the SARDINE lab as a whole. Our lab is not just a gathering of extremely talented individuals—it is a close-knit group that has been a cornerstone of my experience throughout my studies. I could not have hoped for a more inspirational environment, and I'm certain my PhD journey would have been far less exciting without the support, care, and friendship of my colleagues.

Special thanks to António Farinhas and Patrick Fernandes, my partners in crime, and my dearest friends. You have been a constant presence since day one, and I couldn't have asked for better companions to go through this PhD journey with. Perhaps unknowingly, you have made me into a better researcher and, most importantly, a much better person. You are an endless source of inspiration.

I also want to thank all my co-authors; without your contributions, my research would be much poorer. In particular, I have been fortunate to work with researchers that have also shaped who I am as a researcher: an incredible mentor in Lena Voita, who instilled that if we are to do something, it has to be done properly; Ricardo Rei and Pedro Martins, whose energy, generosity, and shared vision make our day-to-day collaboration exciting; and the bright Duarte Alves, João Alves, and José Pombal, who inspire me with their sharp minds, fresh perspectives, and incredible talent.

I am also fortunate to have completed my last year of the PhD working with the research team at Unbabel. I am very lucky to be able to work day-to-day alongside everyone in the team. I am immensely proud of our achievements during my time at Unbabel.

Many thanks also go to my friends for the support and welcome distraction, and simply for being in my life: Afonso Costa, Afonso Gonçalves, Alexandre Candeias, Alexandre Correia, Bruno Gonçalves, Fatma Ziya, Filipe Quintino, João Santos, Rui Figueiredo, and others.

I am deeply grateful to my parents, my sister, and my brother-in-law, who have always given me the support and love to pursue my goals ever since I was a kid.

Finally, I want to thank, in particular, my partner Morgan Smith, whose unwavering support and encouragement made everything possible. This thesis, in a way, became our tale of two cities — a story written together between Lisbon and London, with countless journeys back and forth. While the distance asked much of us both, Morgan's love and understanding never faltered. I can't thank you enough for always being there even when we were apart. You are my rock and my source of strength.

Contents

List of Figures

List of Tables

Part I

PROLOGUE

1 Introduction

Language technologies have advanced rapidly in recent years and are now being adopted widely by global stakeholders and decision-makers across various societal sectors. For example, individuals and organizations use these technologies for effective and swift communication worldwide through automatic translation systems.[1] Additionally, global law firms use language models to assist with drafting legal documents, management consulting companies utilize them to tailor business solutions for clients, and universities employ them to assess student applications. Soon enough, these systems may help shape public policy and influence political discourse. Yet, despite this remarkable progress, we are still far from understanding how these models come up with their predictions, how to evaluate them properly, and how to reliably safeguard them against potential failure modes.

1: As of 2016, Google claimed that more than 100 billion words were translated daily by their translation engine.

While this rapid progress in language technologies has been driven by contributions from a broad range of natural language processing (NLP) tasks, machine translation— the key use case explored in this thesis— has played a particularly significant role. Historically, machine translation has often served as a testbed for innovations in NLP, contributing to and benefiting from broader developments in the field. Key innovations originating from machine translation research include the attention mechanism (Bahdanau et al., 2014, 2015) and the transformer architecture (Vaswani et al., 2017), which have massively influenced the field of NLP at large. Following these developments, researchers explored various optimization objectives and architectural modifications for transformer-based models. This led to the introduction of influential pretrained models such as BERT (Devlin et al., 2019), the GPT series (Radford and Narasimhan, 2018; Radford et al., 2019; Brown et al., 2020b; Ouyang et al., 2022), and T5 (Raffel et al., 2020a). These models, sometimes referred to as "foundation models" (Bommasani et al., 2022), have shown adaptability across numerous downstream tasks. Crucially, as transformer-based models scale well (Kaplan et al., 2020; Hoffmann et al., 2022), performance improvements were often achieved by merely increasing the number of parameters and size of the training data. As models became exponentially larger and larger (Brown et al., 2020b; Raffel et al., 2020b; Shoeybi et al., 2020; Askell et al., 2021; Chowdhery et al., 2022a; Biderman et al., 2023; Touvron et al., 2023b), they have revealed remarkable generalization and adaptation capabilities. Concurrently, machine translation systems evolved from bilingual models for single-direction translation to *universal* systems capable of translating multiple languages (Aharoni et al., 2019; Arivazhagan et al., 2019; Fan et al., 2020; Zhang et al., 2020a; Wenzek et al., 2021; Goyal et al., 2022; NLLB Team et al., 2022). More recently, in the spirit of a *one-fits-all* solution, today's generative large language models (LLMs) are able to perform a broad set of tasks, including translation (Vilar et al., 2022; Hendy et al., 2023a), specified solely via text interaction with the model (e.g., by providing some demonstrations or examples of the end-task or by *prompting* with instructions to solve the end-task).[2]

These pretrained models have also been called *foundation* models (Bommasani et al., 2022), exactly because they can be easily adapted as a foundation to solve other downstream tasks.

2: It is important to remark that other factors such as improved computing power, increased data availability, and the open-source movement have also played a part in achieving this rapid progress.

Despite these advancements, we argue that we have incurred a critical *model understanding* debt: the increasingly complex, highly non-linear and over-parameterized design of these language models makes them harder and more costly to examine and interpret. Simultaneously, their ever larger training data is more difficult to appropriately inspect and document, complicating tracing back potential model failures back to the data.[3] As such, while this recipe of scaling up both model size and training data has indeed proven useful to push the limits of performance in various tasks, it simultaneously made it harder to understand how models arrive at their predictions, which is particularly relevant to *debug* them, understand when, how and why they fail, and to identify problems in the training data that might have led to these failures. As a result, paired with widespread and increasing adoption of these technologies, there is a pressing need and a clear demand for research towards obtaining more reliable and transparent models of language.

3: For example, the training data of the current best LLMs (both closed and open-weight) has not been publicly released nor documented.

This thesis aims to contribute towards that goal by focusing on **three main research vectors**:

- understanding and safeguarding against critical failure modes of translation;
- proposing novel interpretable models, including the development of more robust metrics that can provide interpretable and rich evaluation reports;
- developing improved state-of-the-art machine translation systems trained on carefully filtered data and whose generations employ more robust, quality-informed generation strategies.

More broadly, the work in the thesis proposes new methods and comprehensive analysis on detection, understanding and mitigation of extremely pathological outputs that are unfaithful to the input data in translation models, commonly referred to as *hallucinations*. Sidestepping momentarily from the study of machine translation models, we also propose new methods for achieving discriminative models that not only produce decisions, but also provide flexible sparse explanations that explicitly inform those very same decisions. This work on interpretability aligns with and informs our approach to automatic translation evaluation, leading to the development of a new metric whose predictions go beyond a single numerical score, and are instead explicitly supported by identified error spans and their severities. Finally, we bring these threads together in our contribution towards advancing the state-of-the-art of neural machine translation by training large language models of up to 70 billion parameters on curated training data and producing translations explicitly informed by quality evaluation metrics, improving downstream performance and robustness. Through these diverse yet interconnected research directions, our work addresses relevant challenges across the lifecycle of real-world systems—from model development and interpretability, to robust evaluation and analysis, to safeguarding against and mitigating critical failures.

1.1 Challenges and Related Work

In this section, we start by presenting a brief high-level overview of related work in the three main research vectors of our thesis (outlined above in bullet points).[4] The first regards the various failure modes of language models, specifically focusing on the phenomenon of hallucinations in the context of machine translation. The second explores methods that provide insights into the decision-making process of language models while simultaneously enhancing their interpretability, and how we can ultimately leverage these ideas to create richer and more interpretable translation evaluation metrics. The third concerns the development of multilingual generative large language models that push the boundaries of machine translation capabilities across a wide range of language pairs and domains. We describe our motivations in each of these directions and highlight some of the key challenges that this thesis addresses.

4: A more thorough description of previous work will be presented in the next chapter.

1.1.1 Failure modes of language models

Generative language models[5] have improved substantially in recent years, with current models being capable of generating fluent and coherent text increasingly harder to distinguish from human-generated text. However, when these models are deployed *in the wild* for public use, they may harm users in hard-to-predict ways (Perez et al., 2022). For example, in 2016, Microsoft took down its chatbot Tay within 24 hours of its launch after users exposed it to various forms of controversial and inflammatory language, which led the chatbot to repeat and generate racist and homophobic tweets (Lee, 2016). Similarly, in 2024, Google rolled back access to an additional generative AI features in Search, AI Overviews, intended to generate a summary of helpful answers to queries (Grant, 2024). Indeed, while episodic, models generated innacurate and even worrying outputs, spanning from generating false information to unsafe text.

5: We use the term *language model* to refer to systems which are trained to predict the probability of a token (typically, a string of text) given its preceding context in an autoregressive fashion.

Beyond the documented and well-known incidents above, there is a large body of research on the shortcomings and pitfalls of language models. As these systems are remarkable at learning and manipulating patterns of strings of words during training, it is no surprise that many of these failures are associated with failures in appropriately curating this data.[6] Language models have been found to produce misinformation (Lin et al., 2022a; Goldstein et al., 2023), harmful text (Dinan et al., 2019; Wallace et al., 2019; Weidinger et al., 2021), and even leakage of private data such as personal phone numbers and social security numbers (Carlini et al., 2021; Huang et al., 2022; Perez et al., 2022; Lukas et al., 2023). In fact, language models may fail in even more subtle, hard-to-detect and seemingly innocuous ways.

6: Importantly, strategies meant to reduce toxic patterns in the data may actually render language models more brittle to distribution shift, especially on languages used by marginalized groups. This highlights the challenge and tension between the controllability and distributional robustness of language models (Bender et al., 2021; Xu et al., 2021).

From the standpoint of the adequacy of their outputs, language model *hallucinations* lie at the extreme end of error severity. A language model is said to hallucinate when it produces text that bears minimal or no relation *at all* with the input data, irrespective of its fluency. In fact, hallucinations produced by recent language models are, more often than not, perfectly fluent text. We show an example of a hallucination

Table 1.1: Example of a hallucination produced by the machine translation system studied in Chapter 3. The output translation is detached from the content in the source document.

A hallucination in machine translation	
Source	Dominique Busserau Staatssekretär für Verkehr
Reference	Mr Dominique BUSSEREAU Minister of State with responsibility for Transport.
Hallucination	For further information, please contact Christopher Fretwell.

produced by an actual translation model in Table 1.1. While this behavior is critical in any task, we echo the concerns outlined in Bender et al., 2021 on the risks of these pathological outputs in the task of machine translation (MT). These systems may hallucinate and produce translations that are seemingly fluent and coherent to users who do not see the source text or cannot understand the source text on their own. As humans themselves have the tendency to perceive increased fluency as an indicator of the adequacy of the output (Martindale and Carpuat, 2018), MT hallucinations can easily lead users to misinterpret the original meaning and intent of the source text. This can lead to potential devastating impacts to users. A case in point involves the wrongful arrest of a Palestinian man shortly after posting a photo on Facebook with the caption "*Good morning*" in Arabic (Hern, 2017). Facebook's translation service incorrectly translated the caption as *"attack them"* in Hebrew and *"hurt them"* in English, causing Israeli police to detain the man.

Although the problem of hallucinations in machine translation has received some attention in recent years, research on this highly pathological phenomenon still lacks solid ground. This happens for two reasons: first, previous work used multiple and often overlapping definitions and categories of hallucinations which makes it hard to draw connections between observations made in different works (Lee et al., 2018b; Raunak et al., 2021; Zhou et al., 2021); second, since hallucinations are extremely rare, previous work focused on settings in which the phenomenon is amplified, e.g. perturbing data either in training or at inference, or evaluating under domain shift (Lee et al., 2018b; Müller et al., 2020; Wang and Sennrich, 2020; Müller and Sennrich, 2021; Raunak et al., 2021; Voita et al., 2021; Zhou et al., 2021). Critically, the analysis on these works mostly relied on the adequacy of the automatic hallucination detection methods they proposed. However, it is not straightforward to understand whether these methods translate well to unperturbed settings. Additionally, most previous studies have been conducted on *small bilingual* models (<100M parameters) trained on a *single English-centric high-resource* language pair (Raunak et al., 2021; Dale et al., 2022; Ferrando et al., 2022b; Xu et al., 2023a). This leaves a knowledge gap regarding the prevalence and properties of hallucinations in large-scale translation models across different translation directions, domains and data conditions. Finally, the problem of on-the-fly mitigation of these pathological translations, although critical in real-world applications, remains largely overlooked in the literature.

Thus, there is a series of key challenges on what comes to detection, understanding, and mitigation of hallucinations in machine translation. They span from developing a consistent and reasonable taxonomy for future studies, the analysis of hallucinations *in the wild* without artificially perturbing either the training nor the test data, developing

more effective and efficient detection methods, and proposing solutions for mitigating these translations on-the-fly.

1.1.2 Explainable models of language and automatic translation evaluation

The increasing demand for transparency in language models has fueled research on explainability and intepretability methods. This pursuit aims to provide insights into how these models make decisions and how their outputs can be interpreted. In contrast to traditional model interpretability methods that relied solely on post hoc inspection of static model weights (Guyon and Elisseeff, 2003), recent research has proposed methods that seek instead to understand how input features (e.g., words in a document) affect the final outcome, often through analysis of the internal workings of the model during runtime.

Popular techniques of generating explanations include analyzing heatmaps obtained via attention weights (Bahdanau et al., 2015; Martins and Astudillo, 2016; Rocktäschel et al., 2016; Jain and Wallace, 2019; Voita et al., 2019; Wiegreffe and Pinter, 2019), computing gradient attributions (Denil et al., 2015; Li et al., 2016; Sundararajan et al., 2017), performing leave-one-out experiments (Li et al., 2017; Feng et al., 2018; Serrano and Smith, 2019), understanding how predictions change with local input perturbations (Ribeiro et al., 2016), and obtaining the global influence of each input feature to the model prediction (Abnar and Zuidema, 2020; Ethayarajh and Jurafsky, 2021; Kobayashi et al., 2021; Voita et al., 2021; Ferrando et al., 2022a,b; Modarressi et al., 2022; Mohebbi et al., 2023). Differently to these methods that extract *post-hoc* explanations[7], the framework of *selective rationalization*—also referred to as *select-predict* models—embeds a trainable generator module in the model that allows for joint explanation generation and prediction (Lei et al., 2016). Put simply, these models are trained to select a *rationale*—the smallest subset of input tokens that would yield the same output as the full sequence—and make predictions based solely on it. We show an example of a rationale in Table 1.2. Rationalization methods are arguably more *faithful* by construction, as the same rationale that explains the decision also explicitly informs it (Jacovi and Goldberg, 2020).

7: *Post-hoc* explanations are interpretations of a model's predictions, generated after the model decision has been made. As they are produced separately from the original model's decision process, they may not always be faithful to the underlying *logic* of the model.

A rationale for review classification

Document	This record is beatifully written and produced. It is much better than the previous albums!
Prediction	5 stars.
Rationale	This record is beatifully written and produced. It is much better than the previous albums!

Table 1.2: Example of a synthetic rationale (text highlight) example.

Extracting rationales in previous work, however, has presented its own set of challenges, as training and controlling the extraction process is not trivial. Early solutions leveraged both stochastic hard attention mechanisms and regularization to obtain rationales with desirable properties (e.g., sparsity and contiguity; Lei et al. (2016) and Bastings et al. (2019)). However, these approaches, and others like top-k mapping of token-level weights to rationales (Chang et al., 2020; Jain et al., 2020; Paranjape et al., 2020), are non-differentiable and require gradient surrogates or score function estimators (Williams, 1992; Niculae et al., 2023) for end-to-end training. This tends to complicate training by requiring cumbersome hyperparameter tuning and often leads to brittle models that exhibit high variance across different runs. Alternatively, Treviso and Martins (2020) relax the constraint on binary input assignments and consider the rationale to comprise of a sparse probability

distribution over the input words, with words receiving non-zero probability being selected as part of the rationale. They do so by leveraging sparse attention mechanisms (Martins and Astudillo, 2016; Peters et al., 2019) that allow for deterministic end-to-end differentiable training. Despite substantially facilitating training, this method lacks a direct way to control desirable properties such as sparsity and contiguity in the rationale extraction. Consequently, a key challenge is devising rationalization methods that successfully bridge the gap between the controllability of rationale extraction offered by hard attention mechanisms, and the simplicity of end-to-end training facilitated by sparse attention mechanisms.

Concurrent to these developments in rationalization, the field of machine translation evaluation has evolved in lockstep with the advancements in machine translation systems themselves.[8] However, such advancements have also come at the cost of interpretability. Traditionally, evaluation relied heavily on lexical-based metrics such as BLEU (Papineni et al., 2002) and chrF (Popović, 2015), which assess surface-level similarities (e.g., simple n-gram overlap) between translation hypotheses and references. However, translation systems have become more refined and moved away from explicit lexical transfer between languages. This has rendered these metrics less adequate to judge quality for the most advanced systems (Mathur et al., 2020; Kocmi et al., 2021a). In response to this and in line with the developments in other NLP tasks, the field moved to *trainable neural* metrics built on top of pre-trained transformer-based models (Rei et al., 2020a; Sellam et al., 2020; Zhang et al., 2020b). While these newer approaches have consistently been found to outperform lexical metrics in terms of correlations with human judgements (Freitag et al., 2021a,b; Kocmi et al., 2021a), they have introduced their own set of challenges, chief among them the issue of interpretability: the black-box nature of many neural metrics hampers our ability to understand and trust their outputs.

8: In the context of NLP tasks, machine translation is certainly among the best served in terms of evaluation. This is evidenced by the availability of data not only for training metrics but also for 'meta-evaluation', the existence of challenge sets designed to test metric robustness. Here, it is important to remark the pivotal role of the yearly Workshop on Machine Translation (WMT) shared tasks, collecting new data annually and conducting rigorous human evaluations. This not only provides a consistent benchmark for assessing new metrics but also ensures that evaluation methods keep pace with rapidly evolving translation systems.

One promising approach to address these challenges is the development of richer, more informative evaluation reports. Current widely used approaches for automatic evaluation, such as COMET (Rei et al., 2020a) and BLEURT (Sellam et al., 2020), only provide a single numerical score, offering limited insight into specific translation errors or their severity. This lack of granularity hinders the ability to pinpoint and understand the nature of translation mistakes. As such, there is a growing demand for evaluation methods that can go beyond providing a single score, for which a potential solution is that of highlighting specific error spans within translations and indicating their severity, similar to how rationalization techniques aim to identify the most informative parts of input text. Additionally, there are concerns about the robustness of evaluation metrics, particularly in identifying critical pathological translations (Sudoh et al., 2021; Takahashi et al., 2021; Amrhein and Sennrich, 2022a). These challenges underscore the need for new approaches that can bridge the gap between high-performance evaluation and robust, interpretable, more fine-grained evaluation of translation quality.

1.1.3 Advancing neural machine translation with generative large language models

Neural machine translation (NMT) models have evolved significantly since their inception, both in scale, architecture, and resources required to train them. When introduced as a new approach for automatic translation (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014), models comprised an encoder and a decoder: the encoder extracted a fixed-length vector representation from a variable-length input, while the decoder, conditioned on this representation, generated a correct, variable-length translation. These early models, based on recurrent neural networks, were trained on datasets of a few million sentences, with preprocessing often limiting sentences to as few as 30 words on both source and target sides for computational efficiency.

Soon after, two seminal works inspired by the machine translation task transformed the landscape of not only translation but also other NLP tasks. First, the introduction of the attention mechanism (Bahdanau et al., 2015) enabled models to dynamically leverage information from different parts of a source text—as opposed to being conditioned on a single fixed representation—during translation generation. Second, the advent of the transformer (Vaswani et al., 2017), took the idea of attention even further by constructing an architecture that dispensed recurrences and convolutions, relying solely on attention mechanisms.

As bilingual machine translation systems—those designed to translate for a single language pair—matured, researchers began investigating multi-way, multilingual neural machine translation, enabling a single model to translate between multiple languages (Firat et al., 2016). The scalability of transformer networks facilitated the development of *massively* multilingual translation systems (Arivazhagan et al., 2019; Fan et al., 2020; Akhbardeh et al., 2021; Wenzek et al., 2021; Bapna et al., 2022; NLLB Team et al., 2022). These models, such as NLLB (NLLB Team et al., 2022) which supports translation for over 40,000 translation directions, relied on sophisticated and expensive data acquisition and filtering data pipelines.[9] Around this time, such models were parameterized by billions of parameters and were being trained on over 10 billion samples of parallel data. While these approaches led to very successful models, researchers began exploring alternatives to these computationally expensive processes.

The emergence of generative large language models (LLMs) introduced a new paradigm for addressing NLP tasks at large, including machine translation, through text interaction, particularly through few-shot learning (Brown et al., 2020a).[10] In this few-shot learning paradigm, models are prompted to translate without explicit training on parallel data.[11] This approach represents a fundamental shift from the previous paradigm that required training on large-scale corpora exclusively comprised of parallel data, and more closely resembles that of unsupervised translation (Ravi and Knight, 2011; Artetxe et al., 2018; Lample et al., 2018). As LLMs improved and started being trained on more and more data, it was in no time that the potential of LLM-based MT was quickly realized and these models were shown to outperform previous state-of-the-art neural machine translation systems (Chowdhery et al., 2022b; Lin et al., 2022c; Vilar et al., 2022; Garcia et al., 2023; Hendy et al., 2023b; Kocmi et al., 2023). However, this performance was primarily

9: A common approach to gather data for training these massively multilingual translation models is *backtranslation* (Sennrich et al., 2016a). This strategy leverages the wide availability of monolingual data in multiple languages to create new parallel data samples by pairing it with an automatic backtranslation. This method allows researchers to augment their training data significantly, especially for language pairs with limited naturally occurring parallel texts. Curiously, such ideas have inspired novel methods to create high quality instruction following datasets for training and aligning LLMs (Li et al., 2024).

10: Large language models have been shown capable of performing arbitrary tasks by exposing a few examples of the task at inference time.

11: The claim of whether good translation capabilities are obtained through no supervision at all from parallel data has been disputed in Briakou et al., 2023. In this work, the authors show that *incidental bilingualism*—the unintentional consumption of bilingual signals, including translation examples—during pre-training of LLMs can have a substantial impact on translation capabilities.

achieved using proprietary, closed commercial systems. In contrast, the development of open-weight large language models—which allow for training and development of new improved models—have largely remained focused on English, with limited data explicitly reserved for any other languages.[12] This disparity presents therefore a relevant challenge: efficiently adapting existing English-centric LLMs to excel and outperform these closed models at multilingual tasks, particularly machine translation, is not trivial, especially under restrictions in training compute.

12: The LLaMA models, the most widely used suite of open models, exemplify this trend. LLaMA has undergone three iterations, each with varying degrees of multilingual capability. While the first two iterations were not intended for multilingual usage at all, the latest iteration has been intentionally trained with multilingual data: 8% of the overall training corpus of 15 trillion tokens of text.

1.2 Summary of Contributions

We summarize the main contributions of the work developed in this thesis, addressing some of the key challenges mentioned above.

1. **We provide, in Chapter 3, a comprehensive study of natural hallucinations in machine translation**, assessing the validity of a broad range of detection methods and claims regarding the inner workings of translation systems when hallucinating. Moreover, we provide a simple and efficient solution to effectively mitigate hallucinations on-the-fly.
2. **In Chapter 4, we propose a novel unsupervised plug-in hallucination detection method that explicitly and effectively uses model-based information.** We hypothesize that as hallucinations—contrary to good translations—are not supported by the source content, they may exhibit model patterns that are statistically different from those found in good quality translations. Based on this hypothesis, we approach the problem of hallucination detection as a problem of anomaly detection with an optimal transport formulation (Kantorovich, 2006; Peyré, Cuturi, et al., 2019). Our method not only outperforms all previous model-based detectors, but is also competitive with external detectors that employ auxiliary models that have been trained on millions of samples.
3. **We investigate hallucinations in massively multilingual translation models, exploring a wide range of translation scenarios spanning over 100 translation directions that remained overlook in previous work (Chapter 5).** Our analysis provides several key insights on the prevalence and properties of hallucinations across models of different scale, translation directions, and data conditions, including: the prevalence of hallucinations across multiple translation directions across different resource levels and beyond English-centric directions; the emergence of toxicity in hallucinations; how model distillation may bring reduced hallucination rates; and, the effect of scaling up within the same model family on the prevalence of hallucinations. We also examine and provide relevant analysis on how fallback systems can help mitigate hallucinations and improve translation quality on-the-fly.
4. **In Chapter 6, we propose a unified framework for deterministic extraction of structured rationales.** Our models are not only easier to train, but also generally outperform previous rationalization solutions. They do so while exhibiting less variability

than stochastic methods and easing regularization of the rationale extraction when compared to previous deterministic approaches.

5. **We introduce XCOMET, a novel suite of evaluation metrics for machine translation that combines regression-based scoring with fine-grained error span detection (Chapter 7).** This metric achieves state-of-the-art performance across sentence-level, system-level, and error span prediction tasks, while offering more interpretable and detailed analysis of translation quality. XCOMET bridges the gap between high-performance evaluation and the need for interpretable, granular reports on translation errors. Our own ensemble based on this suite of metrics won the WMT 2023 Metrics Shared Task.
6. **We present TOWER (Chapter 8), a suite of advanced multilingual large language models that achieve state-of-the-art results in machine translation.** Through continual pretraining and supervised fine-tuning on high-quality instructions, TOWER demonstrates superior performance across a broad range of language pairs. Leveraging the advancements in machine translation evaluation, we present our submission to the WMT 2024 General Translation Shared Task which combines a 70-billion parameter TOWER model with quality-aware decoding strategies that allow for generation of translations directly informed by quality metrics. Our submission has achieved the first place on the shared-task according to the official reference-based automatic evaluation, beating closed commercial systems like GPT-4 and CLAUDE-3.5-SONNET.

Publications

The work in this thesis is based on the following published research papers:[13]

13: *: joint first-authorship.

- **Nuno M. Guerreiro** and André F.T. Martins. *SPECTRA: Sparse Structured Text Rationalization*. In Proc. EMNLP 2021.
- **Nuno M. Guerreiro**, Elena Voita, and André F.T. Martins. *Looking for a Needle in a Haystack: A Comprehensive Study of Hallucinations in Neural Machine Translation*. In Proc. EACL 2023.
- **Nuno M. Guerreiro**, Pierre Colombo, Pablo Piantanida, and André F.T. Martins. *Optimal Transport for Unsupervised Hallucination Detection in Neural Machine Translation*. In Proc. ACL 2023.
- **Nuno M. Guerreiro**, Duarte M. Alves, Jonas Waldendorf, Barry Haddow, Alexandra Birch, Pierre Colombo, and André F.T. Martins. *Hallucinations in Large Multilingual Translation Models*. In Transactions of the Association for Computational Linguistics, 2023.
- **Nuno M. Guerreiro***, Ricardo Rei*, Daan van Stigt, Luísa Coheur, Pierre Colombo, André F.T. Martins. *xCOMET: Transparent Machine Translation Evaluation through Fine-grained Error Detection*. In Transactions of the Association for Computational Linguistics, 2024.

- Duarte M. Alves*, José Pombal*, **Nuno M. Guerreiro***, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G.C. de Souza, André F.T. Martins. *Tower: An Open Multilingual Large Language Model for Translation-Related Tasks*. In Proc. COLM 2024.

- Ricardo Rei*, José Pombal*, **Nuno M. Guerreiro***, João Alves*, Pedro H. Martins*, Patrick Fernandes, Helena Wu, Tânia Vaz, Duarte M. Alves, Amin Farajian, Sweta Agrawal, António Farinhas, José G.C. de Souza, André F.T. Martins. TOWER-V2: *Unbabel-IST 2024 Submission for the General MT Shared Task.* In Proc. WMT 2024.

Finally, to make this dissertation a coherent piece, we omit some of the research work carried out during the doctoral studies. This includes work on several topics such as developing, explaining and understanding metrics for evaluation of machine translation outputs, as well as the development and analysis of LLMs for various translation-related tasks. This work refers to the following publications:

- Marcos Treviso, **Nuno M. Guerreiro**, Ricardo Rei, and André F. T. Martins. *IST-Unbabel 2021 Submission for the Explainable Quality Estimation Shared Task*. In Proc. Eval4NLP 2021. *Winners of Best Explainability approach.*

- Ricardo Rei*, Marcos Treviso*, **Nuno M. Guerreiro***, Chrysoula Zerva*, Ana C. Farinha, Christine Maroti, José G. C. de Souza, Taisiya Glushkova, Duarte M. Alves, Alon Lavie, Luísa Coheur, and André F. T. Martins. *CometKiwi: IST-Unbabel 2022 Submission for the Quality Estimation Shared Task*. In Proc. WMT 2022. *Winners on all tracks.*

- Ricardo Rei, **Nuno M. Guerreiro***, José Pombal, Daan van Stigt, Marcos Treviso, Luisa Coheur, José G.C. de Souza, André F.T. Martins. *Scaling up COMETKIWI: Unbabel-IST 2023 Submission for the Quality Estimation Shared Task*. In Proc. WMT 2023.

- Marcos Treviso, Alexis Ross, **Nuno M. Guerreiro**, and André F.T. Martins. *A Joint Framework for Rationalization and Counterfactual Text Generation.* In Proc. ACL 2023.

- Ricardo Rei*, **Nuno M. Guerreiro***, Marcos Treviso, Luísa Coheur, Alon Lavie, and André F.T. Martins. *The Inside Story: Towards Better Understanding of Machine Translation Neural Evaluation Metrics.* In Proc. ACL 2023.

- Duarte M. Alves, **Nuno M. Guerreiro**, João Alves, José Pombal, Ricardo Rei, José G. C. de Souza, Pierre Colombo, André F. T. Martins. *Steering Large Language Models for Machine Translation with Finetuning and In-Context Learning*. Findings of EMNLP 2023.

- Anas Himmi, Guillaume Staerman, Marine Picot, Pierre Colombo, **Nuno M. Guerreiro**. *Enhanced Hallucination Detection in Neural Machine Translation through Simple Detector Aggregation*. In Proc. EMNLP 2024.

- Manuel Faysse*, Patrick Fernandes*, **Nuno M. Guerreiro***, António Loison*, Duarte M. Alves, Caio Corro, Nicolas Boizard, João Alves, Ricardo Rei, Pedro H. Martins, Antoni Bigata Casademunt, François Yvon, André F.T. Martins, Gautier Viaud, Céline Hudelot, Pierre Colombo*. *CroissantLLM: A Truly Bilingual French-English Language Model*. Under submission at TMLR. 2024.

- Sweta Agrawal, José G. C. de Souza, Ricardo Rei, António Farinhas, Gonçalo Faria, Patrick Fernandes, **Nuno M. Guerreiro**, André F.T. Martins. *Modeling User Preferences with Automatic Metrics: Creating a High-Quality Preference Dataset for Machine Translation*. In Proc. EMNLP 2024.

- Emmanouil Zaranis, **Nuno M. Guerreiro**, André F.T. Martins. *Analyzing Context Contributions in LLM-based Machine Translation*. Findings of EMNLP 2024.

- Marcos Treviso, **Nuno M. Guerreiro**, Sweta Agrawal, Ricardo Rei, José Pombal, Tania Vaz, Helena Wu, Beatriz Silva, Daan van Stigt, André F.T. Martins. *xTower: A Multilingual LLM for Explaining and Correcting Translation Errors*. Findings of EMNLP 2024.

- Hippolyte Gisserot-Boukhlef, Ricardo Rei, Emmanuel Malherbe, Céline Hudelot, Pierre Colombo and **Nuno M. Guerreiro**. *Is Preference Alignment Always the Best Option to Enhance LLM-Based Translation? An Empirical Analysis*. In Proc. WMT 2024.

- Pedro Henrique Martins, Patrick Fernandes, João Alves, **Nuno M. Guerreiro**, Ricardo Rei, Duarte M. Alves, José Pombal, Amin Farajian, Manuel Faysse, Mateusz Klimaszewski, Pierre Colombo, Barry Haddow, José G. C. de Souza, Alexandra Birch, André F. T. Martins. *EuroLLM: Multilingual Language Models for Europe*. arXiv pre-print 2024.

1.3 Organization of the Thesis

This thesis is divided into five parts, which we describe below.

▸ **Part I: Prologue.** The first part includes the present chapter as well as Chapter 2, which presents the necessary and relevant background for the core topics of the thesis. This includes brief overviews of selective rationalization, language modeling, neural machine translation, attention mechanisms and the transformer network, machine translation evaluation and hallucinations in machine translation.

▸ **Part II: Safeguards.** Here we address the problem of hallucinations in machine translation, presenting novel contributions in terms of detection, understanding, and mitigation of these translations. It is comprised of the following chapters:

- Chapter 3 establishes a foundation for studying hallucinations in classical bilingual translation models, comparing detection methods, creating a human-annotated dataset, and proposing a new method for on-the-fly mitigation of these translations.

- Chapter 4 introduces a novel hallucination detection method based on Optimal Transport theory, offering a simple yet effective approach to this challenge.
- Chapter 5 extends our analysis to massively multilingual translation models and generative large language models, examining hallucination properties across over 100 language pairs.

▸ **Part III: Interpretable Evaluation.** This part explores selective rationalization and novel fine-grained evaluation methods for machine translation.

- Chapter 6 introduces SPECTRA, a novel framework for deterministic extraction of structured rationales in text classification tasks that addresses limitations of previous work in terms of rationale extraction and ease of training.
- Chapter 7 presents XCOMET, a novel metric for automatic evaluation of translation quality that provides interpretable reports with error spans and severities. We present results in system-level, sentence-level and error span prediction tasks, and provide a robustness analysis to critical errors and hallucinations.

▸ **Part IV: Adavancing Capabilities.** This part focuses on advancing multilingual generative large language models for translation tasks:

- Chapter 8 presents TOWER, a suite of multilingual large language models of up to 70 billion parameters optimized for translation-related tasks. We provide details on data, training, and decoding strategies that led to our submission to the WMT 2024 General Translation Shared Task. We present results on relevant test sets and compare with state-of-the-art open and close, commercial systems.

▸ **Part V: Epilogue.** This part concludes the thesis, by providing, in Chapter 9, a summary of contributions and drawing possible directions of future work.

Background 2

This chapter provides the necessary background for the core topics of this thesis. The aim is to establish a clear understanding of the methodologies and analysis works that underpin each of the topics in the thesis, as they were at the beginning of the doctoral research. Concurrent developments during the period of this study may be acknowledged, but only briefly, as the focus here is to provide an overview of the relevant knowledge that directly informed and motivated the research presented in the thesis.

We will not constrain the presentation of the topics in this chapter to follow the same order as they will feature in the thesis work.

2.1 Rationalizers

Selective rationalization (Lei et al., 2016; Bastings et al., 2019; Swanson et al., 2020) is an explainability method, in which we construct models (*rationalizers*) that produce an explanation or *rationale* (e.g: text highlights or alignments; Zaidan et al. 2007) along with the decision.

2.1.1 Rationalization for highlights extraction

Rationalization models for highlights extraction, also known as *select-predict* or *explain-predict* models (Jacovi and Goldberg, 2021; Zhang et al., 2021), are based on a cooperative framework between a rationale generator and a predictor: the generator component encodes the input text and extracts a "rationale" (e.g., a subset of highlighted words), and the predictor classifies the input conditioned *only* on the extracted rationale.[1] Typically, this is done by obfuscating the words that are not in the rationale with a binary mask.

▸ **Highlights Extraction.** We consider a standard text classification or regression setup, in which we are given an input sequence $\boldsymbol{x} \in \mathbb{R}^{D \times L}$, where D is the embedding size and L is the sequence length (number of words), and we want to predict its corresponding label $y \in \mathbb{R}$ for regression or $y \in \{1, \ldots, C\}$ for classification. A generator model, gen, encodes the input text $\boldsymbol{x}$ into token-level scores. Then, a rationale $\boldsymbol{z}$, e.g. a binary mask over the tokens, is obtained via an extraction layer based on these scores. Subsequently, the predictor model makes predictions conditioned only on the masked input $\hat{y} = \text{pred}(\boldsymbol{z} \odot \boldsymbol{x})$, where $\odot$ denotes the Hadamard (elementwise) product. We present an illustration of the rationalizer in Figure 2.1.

▸ **End-to-end Training and Testing Procedure.** While most rationalization methods deterministically select the rationale at test time, there are differences on how these models are trained. For instance,

1: This design makes rationalizers *faithful* explainability methods (Jacovi and Goldberg, 2020), as the model's predictions are, by construction, explicitly informed by the explanation.

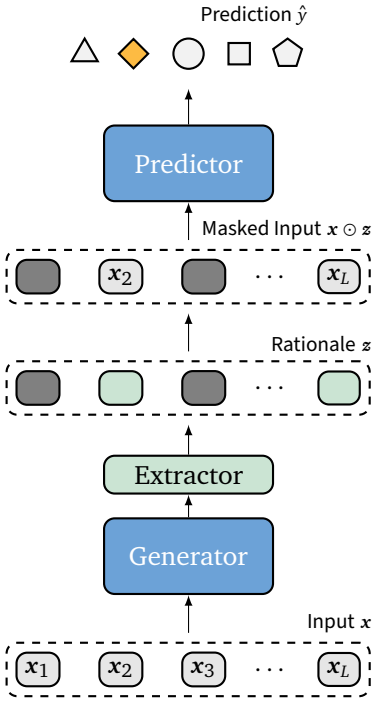


Figure 2.1: Illustration of a rationalizer for highlights extraction.

Lei et al. (2016) and Bastings et al. (2019) use stochastic binary variables (Bernoulli and HardKuma, respectively), and sample the rationale $\boldsymbol{z} \sim \text{gen}(\boldsymbol{x}) \in \{0,1\}^L$, whereas Treviso and Martins (2020) make a continuous relaxation of these binary variables and define the rationale as a sparse probability distribution over the tokens obtained via sparse transformations such as entmax (Peters et al., 2019) and sparsemax (Martins and Astudillo, 2016), e.g., $\boldsymbol{z} = \text{sparsemax}(\text{gen}(\boldsymbol{x}))$. In the latter approach, instead of a binary vector, we have $\boldsymbol{z} \in \triangle^{L-1}$, where $\triangle^{L-1}$ is the $L-1$ probability simplex $\triangle^{L-1} := \{\boldsymbol{p} \in \mathbb{R}^L : \mathbf{1}^\top \boldsymbol{p} = 1, \boldsymbol{p} \geq 0\}$. Words receiving non-zero probability are considered part of the rationale.

Rationalizers that require stochastic sampling or hard heuristics to extract the rationales (e.g., top-k tokens w.r.t some measure of token-level importance) are distinctively difficult to train end-to-end, as they require marginalization over all possible rationales, which is intractable in practice.[2] Thus, recourse to sampling-based gradient estimations is a necessity, either via REINFORCE-style training, which exhibits high variance (Lei et al., 2016; Chang et al., 2020), or via reparameterized gradients (Bastings et al., 2019; Paranjape et al., 2020). This renders training these models a complex and cumbersome task. These approaches are often brittle and fragile for the high sensitivity that they show to changes in the hyperparameters and to variability due to sampling. On the other hand, existing rationalizers that leverage sparse transformations (Treviso and Martins, 2020) such as sparsemax to extract rationales, while being deterministic and end-to-end differentiable, do not enable flexible constrained extraction of the rationale in terms of sparsity and contiguity.

2: Marginalization over all possible rationales requires enumerating an exponential number of assignments of $\boldsymbol{z}$: $\mathcal{O}(2^L)$.

▸ **Constrained Rationale Extraction.** Existing rationalizers are *extractive*: they select and extract words or word pairs to form the rationale. Since a rationalizer that extracts the whole input would be meaningless as an explainer, they must have a length constraint or a sparsity inducing component. Moreover, rationales are idealized to encourage selection of contiguous words, as there is some evidence that this improves readibility (Jain et al., 2020). Some works opt to introduce regularization terms placed on the binary mask such as the ℓ_1 norm and the fused-lasso penalty to encourage sparse and compact rationales (Lei et al., 2016; Bastings et al., 2019). Others use hard constraints through heuristics such as top-k, which is not contiguous but sparse, or select a chunk of text with a pre-specified length that corresponds to the highest total score over all possible spans of that length (Chang et al., 2020; Jain et al., 2020; Paranjape et al., 2020). Sparse attention mechanisms can also be used to extract rationales, but since the rationales are constrained to be in the simplex, controlling the number of selected tokens and simultaneously promoting other desirable properties such as contiguity is non-trivial.

2.1.2 Rationalization for matchings extraction

Swanson et al., 2020 extend the setup of rationalization from text highlights to text matchings. Under this task, the rationale is a linear assignment between the words in two documents. Consider a natural language inference setup in which classification is made based on two input sentences: a premise $\boldsymbol{x}_P \in \mathbb{R}^{D \times L_P}$ and a hypothesis $\boldsymbol{x}_H \in$

$\mathbb{R}^{D \times L_H}$, where L_P and L_H are the lengths of the premise and hypothesis, respectively, and D is the embedding size. A generator model (gen) encodes $\boldsymbol{x}_P$ and $\boldsymbol{x}_H$ separately and then computes pairwise costs between the encoded representations to produce a score matrix $\boldsymbol{S} \in \mathbb{R}^{L_P \times L_H}$. The score matrix $\boldsymbol{S}$ is then used to compute an alignment matrix $\boldsymbol{Z} \in \mathbb{R}^{L_P \times L_H}$, where $z_{ij} = 1$ if the i^{th} premise word is aligned to the j^{th} word in the hypothesis. $\boldsymbol{Z}$ subsequently acts as a sparse mask to obtain text representations that are aggregated with the original encoded sequences and fed to a predictor to obtain the output predictions.

2.2 Neural Autoregressive Language Modeling

An autoregressive language model is a neural network that outputs a probability distribution for the next word based on the context provided by previous words.[3] For a text sequence $\boldsymbol{y} = [\text{START}, y_1, \ldots, y_T, \text{STOP}]$, where each token y_t ($t \in \mathbb{N}$) is a token in a vocabulary $\mathcal{V}$, a language model parameterized by $\boldsymbol{\theta}$ outputs the joint distribution

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}) = \prod_{t=1}^{T} p_{\boldsymbol{\theta}}(y_t | \boldsymbol{y}_{<t}). \tag{2.1}$$

We learn the parameters θ by optimizing the language model to assign high probability to sentences in a vast text corpus. This is achieved by optimizing the log-likelihood function over a set of training sentences $\mathcal{S}$:

$$\mathcal{L}(\boldsymbol{\theta}) = -\sum_{i=1}^{|\mathcal{S}|} \sum_{t=1}^{T_i} \log p_\theta(y_t^i | \boldsymbol{y}_{<t}^i). \tag{2.2}$$

Crucially, note that this objective, commonly referred to as *language modeling objective*, uses the input data to generate labels, without the need for explicit external labels. As such, language models are said to be self-supervised. This is important: obtaining data for training these models is not constrained by any human annotation process; one only needs a corpus of raw text.[4]

At decoding time, the autoregressive nature of the language model is particularly evident. The model generates text one word at a time, conditioning each new word on the sequence of words that have already been generated, commonly in a left-to-right fashion. Generation stops when the model generates the STOP symbol.

2.2.1 Neural machine translation

Neural Machine Translation (NMT) (Kalchbrenner and Blunsom, 2013; Cho et al., 2014; Sutskever et al., 2014) is a key application of autoregressive language models. In essence, NMT aims to solve the task of machine translation—translating a text of arbitrary length N in a source language, denoted as $\boldsymbol{x}$, to text of arbitrary length T in a target language, $\boldsymbol{y}$—with a neural network that directly models the conditional probability $p(\boldsymbol{y}|\boldsymbol{x})$ over an output space of hypotheses $\mathcal{Y}$ conditioned on a source sequence contained in an input space $\mathcal{X}$.

3: In this thesis, we use the terms "autoregressive language model" and "language model" interchangeably. However, it should be noted that other types of language models exist, including nonautoregressive language models (Xiao et al., 2022), bidirectional language models (such as masked language models) (Devlin et al., 2019), among others.

The standard choice to model $p(\cdot|\boldsymbol{y}_{<t})$ is to obtain a vector representation, $\boldsymbol{z}_t \in \mathbb{R}^{|\mathcal{V}|}$, and apply a softmax transformation, $p_\theta(\cdot|\boldsymbol{y}_{<t}) = \text{softmax}(\boldsymbol{z}_t)$, where $[\text{softmax}(\boldsymbol{z})]_i = \frac{\exp(z_i)}{\sum_j \exp(z_j)}$.

4: The self-supervised nature of language models has played an important role in recent years. Scaling the training data is relatively cheap and commonly done by scraping the web for diverse text sources (e.g., Wikipedia, Reddit, books, and others). State-of-the-art language models are now trained on large amounts of unlabeled data, comprising over 500 gigabytes of raw text.

A typical NMT model has two main components: an encoder, which processes the source sentence, and a decoder, which generates the translation. The decoder is essentially an autoregressive language model *conditioned* on the source sentence:[5]

$$p_{\boldsymbol{\theta}}(\boldsymbol{y}|\boldsymbol{x}) = \prod_{t=1}^{T} p_{\boldsymbol{\theta}}(y_t|\boldsymbol{y}_{<t}, \boldsymbol{x}) \tag{2.3}$$

5: As a result, a NMT model is said to be a *conditional language model*.

The model parameters $\boldsymbol{\theta}$ (both encoder and decoder parameters) are learned by maximizing the log-likelihood of reference pairs of source and target sentences. Thus, in contrast to standard language models, NMT does require a specific kind of annotated data: parallel sentences in the source and target languages. In fact, the performance of a NMT model is highly dependent on the availability and quality of this data: high-resource language pairs with abundant available bitext (e.g., German to English) are now translated with remarkable quality, while low-resource language pairs for which bitext is more scarce (e.g., Tamil to English) frequently suffer from mistranslations and other errors (Goyal et al., 2022; NLLB Team et al., 2022).

Traditionally, NMT models have been predominantly bilingual, being trained on abundant parallel data for a single language pair. While this is practical for high-resource language pairs, it is unfeasible for the majority of languages in the world. Moreover, from a practical standpoint, this approach requires $\mathcal{O}(N^2)$ individually trained models to serve N language pairs. When N is large, it becomes extremely difficult to train, deploy and maintain the huge number of models. This motivated research on new ways of reducing this operational burden while simultaneously being able to serve lower-resourced language pairs. One such approach is that of massively multilingual machine translation models, often parameterized by a single large neural network that can handle numerous languages (Arivazhagan et al., 2019; Fan et al., 2020; Akhbardeh et al., 2021; Wenzek et al., 2021; Bapna et al., 2022; Chowdhery et al., 2022a; NLLB Team et al., 2022). These systems translate directly in multiple translation directions without relying on a pivot language.[6]

6: Nevertheless, systems like Google Translate may still use English as a pivot language for some translation directions (Karpinska and Iyyer, 2023). Pivoting to English with bilingual models requires $2N$ models to serve N language pairs: one for each translation into English and one for each translation from English.

The dominant strategy for achieving these systems is training large multilingual models on vast amounts of parallel data often obtained through a combination of data mining and data augmentation strategies, such as backtranslation (Sennrich et al., 2016a; Edunov et al., 2018). Compared to the traditional approach of bilingual models, this approach is not only more efficient and easier to maintain and deploy, but can also bring significant quality improvements, particularly for low-resource and non-English-centric language pairs, as these benefit the most from multilingual transfer (Arivazhagan et al., 2019; Fan et al., 2020).

As an alternative, a novel and promising strategy is to leverage large language models (LLMs).[7] These systems are pretrained on massive *multilingual but nonparallel* corpora with the standard language modeling objective and can be prompted to solve arbitrary tasks (Radford et al., 2019; Brown et al., 2020b)—see Figure 2.2 for an example. Architecture-wise, they differ from standard NMT models, for they do not require an encoder to obtain a source representation: a decoder is simultaneously responsible for processing the source and generating a

7: We understand the term *large language model* as a standard language model (see Equation (2.1)) that is typically parameterized by billions of parameters and is trained on raw text that can reach up to over a trillion words. Crucially, these models exhibit remarkable generalization capabilities and they can be used to generate new text in a variety of tasks, formats, and languages.

translation. In fact, this approach has led to impressive results across a wide variety of NLP tasks (Chowdhery et al., 2022a; Zhang et al., 2022). Translation is no exception: LLMs can produce translations that are competitive with those of supervised translation models without having been trained explicitly on translation data (Vilar et al., 2022; Bawden and Yvon, 2023; Hendy et al., 2023a; Peng et al., 2023).

Translate the following source text from English to French.
Source: The cat sat on the mat.
Target:

Figure 2.2: Prompting a LLM to follow a translation instruction is one of many ways of generating translations with LLMs.

2.3 Attention Mechanisms

Attention networks are now a standard module of the deep learning toolkit, playing a pivotal role in the vast majority of models that are currently being proposed and researched in the field of NLP. Their profound impact spans from contributing to impressive performance gains to allowing better understanding of how models come up with their decisions.

The attention mechanism was originally introduced and motivated for neural machine translation. Previous to the introduction of the attention module, most NMT models were composed by an encoder recurrent neural network (RNN)[8] that read and encoded a source sentence into a fixed-length sentence vector representation, and a decoder RNN that output a translation conditioned on that representation (see Figure 2.3 for an illustration of the architecture). This was subpotimal: for one, the encoder had to compress the entire source sentence in a single vector representation, and then, the decoder could only rely on that single compressed *sentence-level* representation to translate all the different words. The attention mechanism provides a solution for this information bottleneck, by providing a way for the decoder to *attend to* different parts of the input. Instead of just providing a single sentence-level representation, we additionally let the decoder dynamically and selectively retrieve, at each translation step, the most relevant parts of the source sentence. We illustrate the mechanism in Figure 2.4.[9]

Formally, let $[\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N]$ be the hidden states of a RNN, running over the words in the source sequence $\boldsymbol{x}$, $\boldsymbol{q}_t$ is the RNN hidden state of the target decoder at time step t, and z is a categorical latent variable with sample space $\{1, \ldots, N\}$, representing the source position to be attented to for translation. We start by computing the *attention scores*, $s_{i,t}$, using a score function of choice, score, which represent how relevant the source token i is for target step t.[10] Then, we simply apply a softmax transformation to the attention scores to obtain the attention distribution $p(z = i|\boldsymbol{x}, \boldsymbol{q}_t) = \mathsf{softmax}(\boldsymbol{s}_i) \in \triangle^{N-1}$, also commonly known as

8: In a recurrent neural network, for each time step t and corresponding input vector $\boldsymbol{x}_t$, the hidden state $\boldsymbol{h}_t$ is computed and can subsequently used to generate the output $\boldsymbol{y}_t$. This can be concisely described by the equations $\boldsymbol{h}_t = f_\theta(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t)$ and $\boldsymbol{y}_t = g_\phi(\boldsymbol{h}_t)$, where f and g are learned functions parameterized by θ and ϕ, respectively.

9: Figure 2.3 and Figure 2.4 are inspired and adapted from the illustrations in Voita (2020).

10: There are several popular variants for the score function. Bahdanau et al., 2015 propose using a multi-layer perceptron (MLP) as the score function, $\mathsf{score}(\boldsymbol{h}_i, \boldsymbol{q}_t) = \mathsf{MLP}([\boldsymbol{h}_i; \boldsymbol{q}_t])$. Nevertheless, even simpler functions such as the dot-product, $\mathsf{score}(\boldsymbol{h}_i, \boldsymbol{q}_t) = \boldsymbol{h}_i^\top \boldsymbol{q}_t$, have been shown to work well.

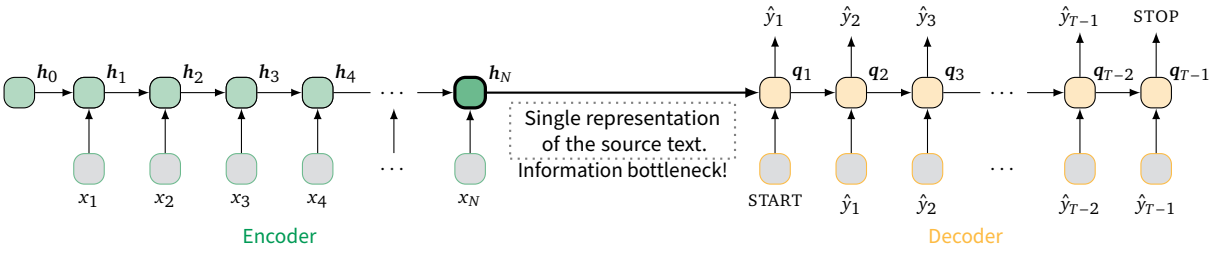


Figure 2.3: Vanilla encoder-decoder model. The decoder generates text based on a single encoder representation of the source text.

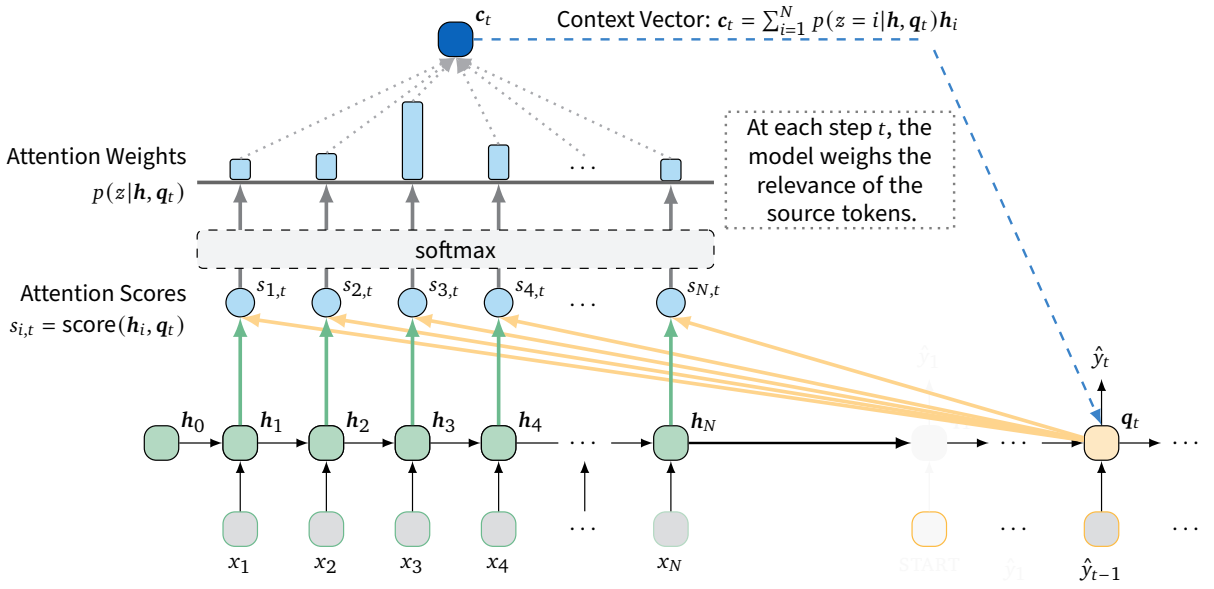


Figure 2.4: Illustration of an encoder-decoder model with a neural attention mechanism. At each step, the decoder dynamically weighs the relevance of the source tokens to generate text.

attention weights. Finally, a *context vector* is obtained by computing the weighted sum of encoder states with the attention weights:[11]

$$\boldsymbol{c}_t = \sum_{i=1}^{N} p(z = i|\boldsymbol{x}, \boldsymbol{q}_t)\boldsymbol{h}_i \tag{2.4}$$

11: The attention mechanism can also be interpreted as taking the expectation of a function $\boldsymbol{h}_z$ with respect to a latent variable $z \sim p$, where p is parameterized to be a function of $\boldsymbol{x}$ and $\boldsymbol{q}_t$ (Bahdanau et al., 2015; Kim et al., 2017).

Then, we incorporate the context vector for step t when obtaining the decoder RNN hidden state for that step:

$$\boldsymbol{q}_t = f(\boldsymbol{q}_{t-1}, \hat{\boldsymbol{y}}_{t-1}, \boldsymbol{c}_t). \tag{2.5}$$

Although initially motivated for machine translation, the attention mechanism proved to be a powerful tool for many sequence-to-sequence tasks, such as question answering, and image captioning. A few years after its introduction, the conceptualization of attention took center stage in the development of the Transformer model (Vaswani et al., 2017): a groundbreaking model architecture that relies solely on attention mechanisms, doing away with previous solutions whose prevalent computational units were recurrences (e.g., LSTMs: Hochreiter and Schmidhuber (1997)) and convolutions (Gehring et al., 2017).

2.3.1 Transformer

The transformer (Vaswani et al., 2017) is an encoder-decoder model that maps the input sequence to the output sequence through a stack of layers of computation. Each layer in the encoder and the decoder is composed by a *multi-head attention mechanism* followed by a feed-forward layer, along with residual connections (He et al., 2016) and layer normalization (Ba et al., 2016). Importantly, it relies solely on attention mechanisms to handle the source-source, target-target and source-target relationships. In the description below, we will focus

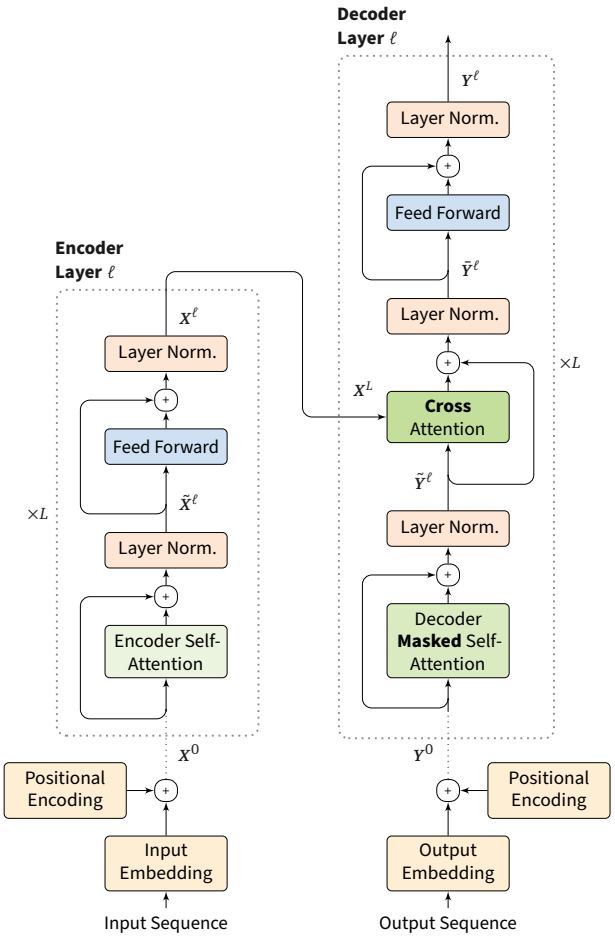


Figure 2.5: Illustration of the (encoder-decoder) transformer network originally proposed in Vaswani et al. (2017).

specifically on describing the multi-head attention (MHA) mechanism and the three distinct purposes it serves in the transformer: (i) encoder self-attention, (ii) decoder self-attention and (iii) cross attention. For a thorough description of the transformer network, please refer to Voita (2020) and Zhang (2022). We illustrate the architecture in Figure 2.5.

▸ **Multi-head attention (MHA).** This attention computation is the core building block of the transformer network, being responsible for combining contextual information in the hidden representations. Different to previous solutions that utilized a single attention computation to obtain a context vector, the multi-head attention mechanism explores several separate attention computations (heads) that run independently and in parallel. This allows for characterizing the inputs in different subspaces with different heads carrying different information and functionalities (e.g., positional information, or tracking syntactic dependencies: Correia et al. (2019) and Voita et al. (2019)).

Formally, given a query $\boldsymbol{Q} \in \mathbb{R}^{|Q| \times d}$, a key $\boldsymbol{K} \in \mathbb{R}^{|K| \times d}$, and value $\boldsymbol{V} \in \mathbb{R}^{|V| \times d}$, where d is the dimensionality of the representations and $|Q|$, $|K|$ and $|V|$ are the respective query, key, value sequence lengths. Often, the keys and values originate from the same source, indicating $|K| = |V|$. The (scaled dot-product) attention mechanism performs the operation:

The scaling term $1/\sqrt{d}$ was added to the *vanilla* dot-product attention mechanism as it was found to improve model stability (Vaswani et al., 2017).

$$\text{Attn}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \underbrace{\text{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d}}\right)}_{\boldsymbol{\Omega} \in [0,1]^{|Q| \times |K|}} \boldsymbol{V} \tag{2.6}$$

where Ω is commonly designated as the attention (weight) matrix.

In the multi-head attention mechanism, the queries, keys and values are split in H ways (denoted as heads) in order to run H independent and separate attention computations, without increasing the model size. We first linearly project the query, key and value matrices to different subspaces by introducing head-specific learnable weight matrices $\boldsymbol{W}_Q^h, \boldsymbol{W}_K^h, \boldsymbol{W}_V^h \in \mathbb{R}^{d \times d_h}$, where $d_h = d/H$. Next, we compute the scaled dot-product attention independently for each head:

$$\text{head}_h = \text{Attn}(\boldsymbol{Q}\boldsymbol{W}_Q^h, \boldsymbol{K}\boldsymbol{W}_K^h, \boldsymbol{V}\boldsymbol{W}_V^h) \tag{2.7}$$

Finally, we concatenate the attention for all H heads, and project the concatenated attention through a linear layer with learnable weights $\boldsymbol{W}_O \in \mathbb{R}^{d \times d}$:[12]

$$\text{MHA}(\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V}) = \text{Concat}\left(\text{head}_1, \text{head}_2, \ldots, \text{head}_H\right) \boldsymbol{W}_O \in \mathbb{R}^{|Q| \times d} \tag{2.8}$$

The attention mechanism offers significant advantages over RNNs and LSTMs, such as improved model parallelization, and better handling of global dependencies compared to convolutional networks.[13] In fact, this mechanism is so powerful and scalable that the need for developing task-specific architectures has diminished. Instead, stacking or transferring multiple attention blocks has proven to be sufficient for various tasks, as demonstrated by influential models like BERT (Devlin et al., 2019), GPT models (Radford et al., 2019; Brown et al., 2020b) and T5 (Raffel et al., 2020a). Even years after its introduction, this multi-head attention mechanism continues to be the foundation of increasingly powerful models. The advancements in these models are driven by training them on ever bigger training datasets, and by simply stacking more multi-head attention mechanisms or increasing the number of heads, the dimensionality of the queries, keys, and values, or of the feed forward blocks.

▸ **Attention in the encoder.** Each encoder layer is constituted by an *attention block* — multi-head attention (MHA), a residual connection (RES), and layer normalization (LN) — and a feed-forward block — feed-forward network, residual connection and layer normalization. We will describe below the computation performed in the attention block.

In the encoder, the attention mechanism is commonly designated as *self-attention*, as each individual part of a sequence attends to all other parts of the same sequence.[14] More formally, in the context of the encoder self-attention, the query, key, and value matrices, $\boldsymbol{Q}$, $\boldsymbol{K}$, $\boldsymbol{V}$, respectively, all originate from the same input sequence representations.

For a transformer network with L encoder layers, the computation in the encoder self-attention block can be summarized as:

$$\tilde{\boldsymbol{x}}_i^\ell = \text{LN}\left(\text{MHA}(\boldsymbol{x}_i^{\ell-1}, \boldsymbol{X}^{\ell-1}) + \boldsymbol{x}_i^{\ell-1}\right) \tag{2.9}$$

12: Moving forward, we will simplify the notation of the MHA module to only receive two arguments, $\text{MHA}(\cdot, \cdot)$, since in most cases, the multi-head attention modules share the same keys and values, i.e, $K = V$. Moreover, we will represent $\text{MHA}(q_i, \cdot)$ as the output of the MHA module corresponding to the query token i.

13: Importantly, while recurrent neural networks naturally account for sequential ordering, the transformer model—relying solely on modules like attention blocks and feed forward networks—treats the input sequences as sets of tokens, thus losing sequence order information. To rectify this, positional embeddings are employed to enable the model to leverage sequential information. This is done in practice by adding a unique learnable vector to each input token that corresponds to its position in the sequence.

14: This is different to the attention mechanism we described in Equation 2.4. In that equation, each decoder state is looking at all encoder states; in self-attention, each state from a set of states is attending at all the other states in the same set.

where $\boldsymbol{x}_i^\ell \in \mathbb{R}^d$ is the i-th input representation at layer $\ell \in \{1, \ldots, L\}$, and $\tilde{\boldsymbol{x}}_i^{\ell-1}$ is the previous layer output representation corresponding to token i. Under this notation, $\boldsymbol{X}^\ell := [\boldsymbol{x}_1^\ell, \ldots, \boldsymbol{x}_N^\ell]$ and $\boldsymbol{X}^0$ corresponds to the input sequence embeddings, $\boldsymbol{X}^0 := [\boldsymbol{x}_1^0, \ldots, \boldsymbol{x}_N^0]$. These representations will then go through the feed-forward block for every layer, such that the final token representations of layer ℓ, $\boldsymbol{X}^\ell$, are obtained via:

$$\boldsymbol{X}^\ell = \mathsf{LN}\left(\boldsymbol{W}_2 \mathsf{ReLU}\left(\boldsymbol{W}_1 \tilde{\boldsymbol{X}}^\ell + \boldsymbol{b}_1\right) + \boldsymbol{b}_2 + \tilde{\boldsymbol{X}}^\ell\right) \tag{2.10}$$

where $\boldsymbol{W}_1 \in \mathbb{R}^{d_{ff} \times d}$, $\boldsymbol{W}_2 \in \mathbb{R}^{d \times d_{ff}}$, $\boldsymbol{b}_1 \in \mathbb{R}^{d_{ff}}$, and $\boldsymbol{b}_2 \in \mathbb{R}^d$ are learnable model parameters. Here, $\mathsf{ReLU}(\cdot) = \max(0, \cdot)$ is the rectified linear unit, i.e., an element-wise maximum operation against zero.

The layer normalization operation, LN, first normalizes the input vector and then applies a transformation with learnable parameters $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^d$:

$$\mathsf{LN}(\boldsymbol{x}) = \frac{\boldsymbol{x} - m(\boldsymbol{x})}{s(\boldsymbol{x})} \odot \boldsymbol{\gamma} + \boldsymbol{\beta} \in \mathbb{R}^d$$

where $m(\boldsymbol{x}) \in \mathbb{R}$ and $s(\boldsymbol{x}) \in \mathbb{R}$ denote the element-wise mean and standard deviation, respectively. The subtraction and division are also performed element-wise.

The residual connection operation, RES, is expressed in Equation 2.9: the original input vector for the multi-head attention $\boldsymbol{x}_i$ is added to its output vector pre layer-normalization.

▸ **Attention in the decoder.** Each decoder layer contains an *attention block* — multi-head attention (MHA), a residual connection (RES), and layer normalization (LN) — followed by a *cross (encoder-decoder) attention block* and a *feed-forward block*, which includes, similarly to the encoder layers, a feed-forward network, a residual connection, and layer normalization. In this section, we focus on the decoder self-attention and cross-attention blocks.

Similar to the encoder, the decoder also employs self-attention. However, unlike the encoder, the decoder self-attention is causal, meaning that it only attends to previous inputs in the output sequence, $Y \in \mathbb{R}^{T \times d}$. This is to ensure that the model does not utilize future information during the generation process.

To enforce causality in the decoder self-attention mechanism, we use a matrix $\boldsymbol{M}$ to mask the positions of the target sequence that the model cannot attend to at each time step, hence why it is commonly designated as *masked* self-attention. To do so, we can simply change the computation of the attention weight matrix in Equation 2.6 to:

$$\boldsymbol{\Omega} = \mathsf{softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d}} + \boldsymbol{M}\right) \tag{2.11}$$

where $\boldsymbol{M} \in \mathbb{R}^{T \times T}$ is a matrix such that its entries above the main diagonal are set to $-\infty$. This forces the corresponding softmax output to 0, thereby enforcing causality. We show a synthetic example of a decoder self-attention weight matrix in Figure 2.6.

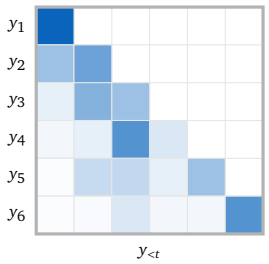


Figure 2.6: A synthetic example of a decoder self-attention weight matrix.

For a transformer network with L decoder layers, the computation in the masked self-attention decoder block can be summarized as:

$$\tilde{\boldsymbol{y}}_i^\ell = \mathsf{LN}\left(\mathsf{MHA}(\boldsymbol{y}_i^{\ell-1}, \boldsymbol{Y}_{<i}^{\ell-1}) + \boldsymbol{y}_i^{\ell-1}\right) \tag{2.12}$$

where $\boldsymbol{y}_i^\ell \in \mathbb{R}^d$ is the i-th input representation at decoder layer $\ell \in \{1, \ldots, L\}$, and $\tilde{\boldsymbol{y}}_i^{\ell-1}$ is the output representation from the previous layer corresponding to the i-th input token. In this notation, $\boldsymbol{Y}^0$ corresponds to the input sequence embeddings in the decoder, $\boldsymbol{Y}^0 := [\boldsymbol{y}_1^0, \ldots, \boldsymbol{y}_N^0]$. The symbol $\boldsymbol{Y}_{<i}^{\ell-1}$ represents, with some abuse of notation, the effective masked output sequence at layer $\ell - 1$, containing the output representations $\tilde{\boldsymbol{y}}_i^{\ell-1}$ for all prefix tokens at time step i.

Additionally, the decoder may contain a *cross-attention* block.[15] This mechanism is responsible for mixing the information from the encoder (e.g., in the use-case of translation, it allows the decoder to access

15: While the decoder in a transformer model was conceived to include a cross-attention block, there are variants of transformer models such as GPT models (Radford and Narasimhan, 2018) that are decoder-only and do not utilize this block. These models generate sequences solely based on previous context, without the need for encoded information from a separate input sequence.

information about the source sentence tokens), and is similar to the approach we illustrated in Figure 2.4. At each time step, the decoder will look at the encoder representations $\boldsymbol{X}^L$. In the context of the query-key-value attention formulation, the representations $\boldsymbol{X}^L$ serve as the keys and values for the cross-attention block. The resulting cross-attention weight matrix will be a matrix $\boldsymbol{\Omega} \in [0, 1]^{T \times N}$.

Incorporating the cross-attention block into the decoder, the computation can be summarized as:

$$\bar{\boldsymbol{y}}_i^\ell = \mathsf{LN}\left(\mathsf{MHA}(\tilde{\boldsymbol{y}}_i^\ell, \boldsymbol{X}^L) + \tilde{\boldsymbol{y}}_i^\ell\right) \tag{2.13}$$

where $\tilde{\boldsymbol{y}}_i^\ell$ is the output representation from the masked self-attention block (Equation 2.12), and $\bar{\boldsymbol{y}}_i^\ell$ denotes the updated representation after layer ℓ, incorporating the cross-attention information from the encoder. These representations are then fed to the feed-forward block similar to that of Equation 2.10 to obtain the final layer representations $\boldsymbol{Y}^\ell$.

For comprehensive surveys on different transformer-based models, please refer to Lin et al. (2022b) and Zhao et al. (2023).

Notably, different transformer-based models employ different combinations of these attention mechanisms. For example, BERT (Devlin et al., 2019), XLM-R (Conneau et al., 2019), and RoBERTa (Liu et al., 2019) are encoder-only models that only utilize encoder self-attention, GPT models (Radford and Narasimhan, 2018; Radford et al., 2019; Brown et al., 2020b; Ouyang et al., 2022) and most LLMs (Chowdhery et al., 2022a; Thoppilan et al., 2022; Zeng et al., 2022; Zhang et al., 2022; Touvron et al., 2023b; Workshop et al., 2023) are decoder-only models that only use decoder self-attention, and (m-)T5 (Raffel et al., 2020a; Xue et al., 2021), as well as most neural machine translation models (Fan et al., 2020; NLLB Team et al., 2022), follow the standard encoder-decoder architecture that use all three different attention mechanisms.

2.3.2 Attention as a means for interpretability

Attention mechanisms, since their inception, have provided an opportunity to analyze, in an interpretable format, the inner workings of neural models (Bahdanau et al., 2015; Martins and Astudillo, 2016; Rocktäschel et al., 2016). For example, in the context of translation, the simple intuition is that through observation of the cross-attention weights, we can assess which positions in the source sentence were considered more important when generating the target word.

The widespread adoption of the transformer architecture, which, as we have seen in the previous section, leverages the attention mechanism as the main computational unit , has led to many research studies on leveraging attention as a means for tracking the flow of information through the model, and to understand how transformer-based models make their predictions (Voita et al., 2018; Clark et al., 2019; Correia et al., 2019; Jain and Wallace, 2019; Serrano and Smith, 2019; Vig and Belinkov, 2019; Voita et al., 2019; Wiegreffe and Pinter, 2019; Abnar and Zuidema, 2020; Kobayashi et al., 2020; Ethayarajh and Jurafsky, 2021; Ferrando and Costa-jussà, 2021; Kobayashi et al., 2021; Ferrando et al., 2022a,b; Modarressi et al., 2022; Mohebbi et al., 2023, *inter alia*).

Although many works debate whether attention can be used as explanation (Jain and Wallace, 2019; Serrano and Smith, 2019), attention has been found to play relevant, interpretable, and recognizable roles in transformer models.[16] These include discerning syntactical structures, merging of subwords, handling of rare words, capturing linguistic phenomena (e.g., anaphora resolution), and others (Voita et al., 2018; Clark et al., 2019; Vig and Belinkov, 2019; Voita et al., 2019). Perhaps most importantly to our work, attention patterns have also been connected to failure modes in neural models, such as generation of hallucinations in machine translation (Berard et al., 2019; Raunak et al., 2021; Ferrando et al., 2022b; Xu et al., 2023b). We will analyse these connections in more depth later in this chapter.

16: Refer to Bibal et al. (2022) for a broader picture on that debate.

2.4 Machine Translation Evaluation

As MT systems have become increasingly sophisticated and widely adopted, the need for accurate and meaningful evaluation methods has grown correspondingly.

Overall, the evaluation of MT quality serves two primary purposes: guiding the development and selection of MT models during the research and development phase, and monitoring the performance of deployed systems in real-world applications (one such relevant case will be introduced in the next chapter on detection of hallucinated translations). While human evaluation remains the gold standard for assessing translation quality, its resource-intensive nature limits scalability for modern MT applications. This limitation has led to a growing reliance on automatic evaluation metrics, which offer speed and scalability but face their own set of challenges in accurately capturing the nuances of translation quality.

Here, we will cover three key areas: methodologies for human assessment of translation quality; an overview on automatic metrics for translation evaluation and current challenges; and modes of evaluation (e.g., reference-free and reference-based evaluation).

2.4.1 Methodologies for human assessment of translation quality.

Human evaluation of machine translation is primarily conducted through three distinct approaches: post-edits (PE), direct assessments (DA), and the Multidimensional Quality Metrics (MQM) framework.

- **Post-edits (PE).** *Professional translators* are tasked with "fixing" a given translation, making minimal edits to improve its quality. Using this edited translation — often termed *post-edit* — we can evaluate the machine translation output by quantifying the number of edits, thus gauging the initial translation's quality (Snover et al., 2006).

- **Direct assessments (DA).** DAs (Graham et al., 2013) are a simple and widely-used evaluation method. Annotators — *non-expert bilingual speakers* or *professional translators* — are asked to annotate each translation with a score ranging from 0 to 100 to reflect its adequacy and

Table 2.1: MQM error hierarchy as illustrated in Freitag et al. (2021a).

Error Category		Description
Accuracy	Addition	Translation includes information not present in the source.
	Omission	Translation is missing content from the source.
	Mistranslation	Translation does not accurately represent the source.
	Untranslated text	Source text has been left untranslated.
Fluency	Punctuation	Incorrect punctuation (for locale or style).
	Spelling	Incorrect spelling or capitalization.
	Grammar	Problems with grammar, other than orthography.
	Register	Wrong grammatical register (eg, inappropriately informal pronouns).
	Inconsistency	Internal inconsistency (not related to terminology).
	Character encoding	Characters are garbled due to incorrect encoding.
Terminology	Inappropriate for context	Terminology is non-standard or does not fit context.
	Inconsistent use	Terminology is used inconsistently.
Style	Awkward	Translation has stylistic problems.
Locale convention	Address format	Wrong format for addresses.
	Currency format	Wrong format for currency.
	Date format	Wrong format for dates.
	Name format	Wrong format for names.
	Telephone format	Wrong format for telephone numbers.
	Time format	Wrong format for time expressions.
Other		Any other issues.
Source error		An error in the source.
Non-translation		Impossible to reliably characterize distinct errors.

fluency, where a score of 100 corresponds to a perfect translation, and 0 corresponds to a completely inadequate one.

▸ **Multidimensional Quality Metrics (MQM).** The MQM framework (Lommel et al., 2014a), on the other hand, offers a more comprehensive and systematic approach to machine translation evaluation. *Professional translators* highlight errors —typically in the form of error spans — within translations, attributing them severity ratings (e.g., *minor*, *major*, or *critical*) and categorical labels (e.g., *fluency*, *accuracy*) — see Table 2.1 for a comprehensive list of translation errors. Importantly, these error spans and their respective severities can be used to obtain a numerical quality score for the translation.[17] Table 2.2 illustrates one such annotation. MQM annotations have gained prominence in recent years due to their capacity to offer detailed insights into translation errors, facilitating more fine-grained and accurate comparisons between translation systems (Freitag et al., 2021a). As such, the field of Automatic Evaluation of MT has increasingly favoured comparisons using MQM annotations over traditional DA and PE methodologies (Freitag et al., 2021b, 2022c; Zerva et al., 2022a).

17: A common approach is to obtain the error counts for each severity level ($c_{\text{MIN}}, c_{\text{MAJ}}, c_{\text{CRIT}}$) and apply predetermined severity penalty multipliers (e.g., 1 for minor errors, 5 for major errors, and 10 for critical errors) to define the error type penalty total, $e(S)$. Formally:

$$e(S) = c_{\text{MIN}} + 5 \times c_{\text{MAJ}} + 10 \times c_{\text{CRIT}}.$$

2.4.2 Automatic metrics for translation evaluation.

Conventional automatic metrics for machine translation (MT) evaluation rely on *lexical*-based approaches, where the evaluation score is computed through statistics related to lexical overlap between a machine translation and a reference translation. Despite evidence indicating that these lexical metrics (e.g., BLEU (Papineni et al., 2002) and CHRF (Popović, 2015)) do not consistently align with human judgments, particularly when these are obtained through the MQM framework (Freitag et al., 2021b, 2022c), they remain very popular.[18]

18: Regardless of their popularity, lexical metrics have become increasingly less effective with the emergence of more complex translation systems, e.g., LLM-based translation systems. These systems often produce high-quality translations that deviate from literal, word-for-word renderings Raunak et al., 2023. In fact, Hendy et al., 2023a show that lexical metrics fail to capture the strong performance of GPT and exhibit lexical and reference bias.

Table 2.2: Examples of translation candidates and their respective MQM annotations. We highlight minor (`MIN`), major (`MAJ`), and critical (`CRIT`) error spans.

Minor and major errors	
Source	Have a wonderful rest of your day and happy new year!
Reference	Ich wünsche Ihnen noch einen schönen Tag, und ein frohes neues Jahr!
Translation candidate	Habt einen wundervollen Rest des Tages und ein glückliches neues Jahr!
MQM annotation	`MIN:` Habt einen wundervollen Rest des Tages und ein `MAJ:` glückliches neues Jahr!
Critical errors	
Source	Handys, die bis auf Wäschewaschen und Staubsaugen scheinbar alles können.
Reference	Mobile phones that can practically do everything except clean the laundry and vacuum clean.
Translation candidate	The staff were very friendly and helpful.
MQM annotation	`CRIT:` The staff were very friendly and helpful.

In fact, BLEU remains the most widely employed evaluation metric in machine translation to this day (Marie et al., 2021). On the other hand, *neural* metrics (e.g., COMET (Rei et al., 2020a), BLEURT (Sellam et al., 2020), and METRICX (Juraska et al., 2023)) that rely on complex neural networks to estimate the quality of MT outputs — see Figure 2.7 for an illustration of the typical architecture of such metrics — are consistently among the best metrics for MT evaluation according to correlations with human judgments (Freitag et al., 2021b, 2022c).

However, contrary to lexical metrics which offer a straightforward interpretation, it can often prove challenging to explain the score predicted by a *neural* metric to a given translation output. As such, there have been a series of efforts to bring interpretability to neural metrics by focusing on understanding the inner workings of neural metrics (Leiter et al., 2023; Rei et al., 2023c), or on constructing inherently interpretable neural metrics (e.g., MATESE (Perrella et al., 2022) and FG-TED Bao et al., 2023) by assigning a central role to the task of predicting *word-level* errors in a given translation, instead of *just* a sentence-level score.[19]

More recently, with the rise of generative LLMs, some works have tried to frame the MT evaluation problem as a generative problem. This offers great flexibility, as the LLM can be prompted to either score the translation directly (Kocmi and Federmann, 2023b), or to identify errors in the translation (e.g., in line with the MQM framework) (Fernandes et al., 2023a; Xu et al., 2023c).

19: Note that, as we have seen above, word-level errors/error spans can be easily converted to sentence-level scores.

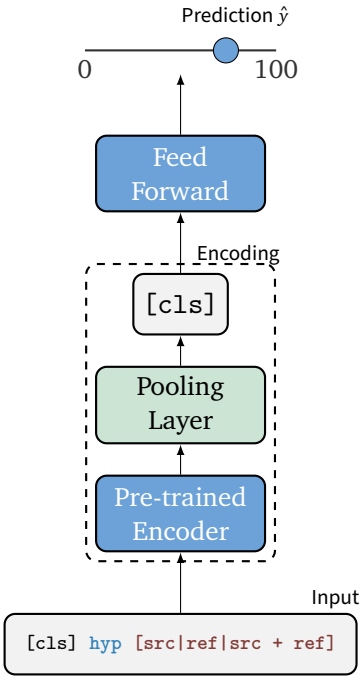


Figure 2.7: Top-level architecture of neural metrics like COMETKIWI and BLEURT. The input to the model starts with a `[cls]` token followed by a translation hypothesis (`hyp`) and an additional input that will have the source, reference or both. After the pooling layer the `[cls]` token is passed to a feed-forward network to produce a quality score $\hat{y}$.

2.4.3 Modes of evaluation

An automatic *metric* for translation evaluation aims at predicting the quality of a translated sentence, $\boldsymbol{t}$, in light of a reference translation, $\boldsymbol{r}$, for a given source sentence, $\boldsymbol{s}$. Here, we focus specifically on neural metrics that make use of a neural model, and typically operate under one of the following evaluation scenarios:

- **reference-only** (REF): the model evaluates the translation by processing it alongside a ground-truth reference sentence (BLEURT is an example of such a metric);
- **source-reference combined input** (SRC+REF): the model evaluates the translation by jointly processing it with both the source and the reference (COMET is an example of such a metric);

- **source-only** (SRC): the model evaluates the translation using only its corresponding source sequence (COMETKIWI (Rei et al., 2022b) is an example of such a model). This mode is commonly termed as *quality estimation* (QE) or *reference-free* evaluation (Specia et al., 2010).

In essence, the model's input sequence consists of the translation $\boldsymbol{t}$ paired with some **additional input**—either $\boldsymbol{r}$, $[\boldsymbol{r}, \boldsymbol{s}]$ or $\boldsymbol{s}$—derived from the scenarios above. Given this input, the model may predict the quality of the translation at different granularities, e.g., sentence-level or word(span)-level.

▸ **SENTENCE-LEVEL PREDICTION.** The model is tasked to predict a single global score—typically between 0 and 1—for the translation, that represents how well it aligns with its context (i.e., source and/or reference sentence). These scores can be used for a broad range of tasks, such as gauging the quality of different translation systems (Freitag et al., 2022c), identifying pathological translations (Guerreiro et al., 2023a), assisting the generation of translations by MT systems (Fernandes et al., 2022b), or even acting as reward models for human alignment of language models (Gulcehre et al., 2023).

▸ **WORD(SPAN)-LEVEL PREDICTION.** In contrast, word-level (or span-level) predictions are more fine-grained, identifying individual words or phrases in the translation that may have errors or discrepancies—typically identifying them as OK/BAD or according to their severity, e.g., MINOR/MAJOR. These granular evaluations are more interpretable and assist in pinpointing specific issues, which can be particularly valuable for feedback and iterative translation improvements.

While both granularities, in particular the error span information, can provide relevant and informative quality signals, the vast majority of widely used neural metrics for evaluation exclusively perform sentence-level prediction (e.g., COMET, BLEURT, METRICX).

2.5 Hallucinations in Machine Translation

Hallucinations lie at the extreme end of translation pathologies and present a critical challenge in machine translation, as they can compromise the safety and reliability of real-world applications that employ these systems.

Importantly, hallucinations in machine translation are unlike hallucinations in other natural language generation tasks (e.g., abstractive summarization and generative question answering) (Ji et al., 2022). While, for these other tasks, models often produce hallucinated outputs (Falke et al., 2019; Cao et al., 2022; Manakul et al., 2023), hallucinations in machine translation, possibly due to the more closed-ended nature of the task, are substantially rarer and hard to observe in clean, unperturbed data.[20] This has led several previous studies to examine their properties by creating artificial scenarios where hallucinations are more likely to occur (e.g., introducing perturbations in the source text (Lee et al., 2018a) or noise in the training data (Raunak et al., 2021)). To distinguish these two scenarios, hallucinations in machine translation

20: It is important to note that the majority of research evaluating translation systems is carried out on clean, high-quality datasets, such as WMT benchmarks. Because hallucinations are relatively infrequent under these conditions, their impact on corpus-level automatic metrics tends to be low. However, this may inadvertently downplay their potential significant influence on the user experience when interacting with translation systems *in the wild*. As such, studies specifically focusing on the phenomenon of hallucinations in machine translation are not only relevant, but essential for improving the reliability of translation systems in practical real-world use cases.

Table 2.3: Example of a hallucination under perturbation.

Good quality translation for the original source sequence		
Source	Nadal bagged 88% net points in the match winning 76 points in the first serve.	
Reference	Nadal acumuló'l 88% de puntos netos nel partíu ganando 76 puntos nel primer serviciu.	
Original MT	Nadal ganó'l partíu con un 88% de puntos netos ganando 76 puntos nel primer serviciu.	BLEU: 57.14
Poor quality translation for the perturbed source sequence		
Perturbed Source	are Nadal bagged 88% net points in the match winning 76 points in the first serve.	
Hallucination	El so debú na Primer División de Bélxica produciόse'l 24 d'abril de 2012 nel Estadiu de Wembley, con un marcador de 3-0 a favor de los locales.	BLEU: 2.09

Table 2.4: Examples of natural (detached and oscillatory) hallucinations.

Detached hallucination	
Source	Handys, die bis auf Wäschewaschen und Staubsaugen scheinbar alles können.
Reference	Mobile phones that can practically do everything except clean the laundry and vacuum clean.
Hallucination	The staff were very friendly and helpful.
Oscillatory hallucination	
Source	In dieser Zeit stürzt sich Murnau bereits wieder in Theaterproben, allerdings widmet er sich nicht mehr der Schauspielerei, sondern der Regie.
Reference	During this period, Murnau once again dedicates his time to theatre rehearsals; however, this time not as an actor, but as a director.
Hallucination	Murnau was born in Murnau, Germany. He was born in Murnau, Germany.

are categorized into two types (Raunak et al., 2021): *hallucinations under perturbation* and *natural hallucinations*.

▸ **Hallucinations under perturbation.** A model generates a hallucination under perturbation when it produces a significantly lower quality translation for a slightly perturbed input[21] compared to the original input (Lee et al., 2018a). Hallucinations under perturbation explicitly reveal the lack of robustness of translation systems to perturbations in the source by finding translations that undergo significant negative shifts in quality due to these changes. We provide an example of a hallucination under perturbation in Table 2.3.

21: Perturbed source sentences are usually obtained via simple and minimal modifications of the original source sentence: introduction of tokens in the beginning of the sentence, capitalization errors or misspellings.

▸ **Natural hallucinations.** These translations occur naturally, without any perturbation. Here, we follow the taxonomy introduced in Raunak et al. (2021). Under this taxonomy, hallucinations are translations that contain *content that is detached from the source*, and are further categorized as *fluent detached hallucinations* or *oscillatory hallucinations*. The former refers to translations that bear no relation at all to the source, while the latter refers to inadequate translations that contain erroneous repetitions of words and phrases. We show examples of both types of natural hallucinations in Table 2.4.

2.5.1 Detection of hallucinations

Hallucinations under perturbation

Hallucinations under perturbation are detected via a simple 2-rule process initially proposed in Lee et al. (2018b). Following work that

Algorithm 1: Detection of hallucinations under perturbation

Data: Translation Model $\mathcal{M}$, Parallel Corpus (X, Y)
Initialize: minimum quality threshold for original translations: t_{orig};
maximum quality threshold for perturbed translations: t_{pert};
for *each pair* $(\boldsymbol{x}, \boldsymbol{y})$ *in* (X, Y) **do**
  translation $\hat{\boldsymbol{y}} = \mathcal{M}(\boldsymbol{x})$;
  perturbed source sentence $\boldsymbol{x}_{\text{pert}} = \text{perturb}(\boldsymbol{x})$;
  perturbed translation $\hat{\boldsymbol{y}}_{\text{pert}} = \mathcal{M}(\boldsymbol{x}_{\text{pert}})$;
  original score = quality($\hat{\boldsymbol{y}}$, $\boldsymbol{y}$);
  perturbed score = quality($\hat{\boldsymbol{y}}_{\text{pert}}$, $\boldsymbol{y}$);
  if *original score* $> t_{orig}$ ***and*** *perturbed score* $< t_{pert}$ **then**
    Hallucination detected;

analysed this type of hallucinations used different variants of this algorithm (Raunak et al., 2021; Ferrando et al., 2022b; Xu et al., 2023a). The algorithm is shown in Algorithm 1: we fix (i) a minimum threshold quality score for the original translations, and (ii) an extremely low maximum quality score for the perturbed translations. A model generates a hallucination under perturbation when both translations meet the thresholds.[22] Crucially, rule (i) ensures that low-quality translations for unperturbed sources are not considered as candidates for hallucinations under perturbation (these translations may fall instead under the scope of natural hallucinations).

22: Previous work has consistently used variants of BLEU (Papineni et al., 2002) to obtain the quality scores. Thresholds for reasonable and very low quality translations, implemented in steps (i) and (ii), have varied across studies. They typically range from a BLEU score of 9 to 30 for reasonably good quality translations, and from 1 to 3 for extremely low-quality translations.

Natural hallucinations

Approaches to detection of natural hallucinations generally aim to find low-quality translations that may also satisfy additional constraints. Previous work either relied only on quality filtering (Müller and Sennrich, 2021; Raunak et al., 2021), or only on heuristics (Berard et al., 2019), or a combination of the two (Raunak et al., 2021). For example, Raunak et al. (2021) proposed two binary heuristics to detect oscillatory and fully detached hallucinations. Given a corpus of source-translation pairs, a translation is flagged as an hallucination if it is in the set of 1% lowest-quality translations (according to some multilingual scoring model), and if:

- **Top n-gram count (TNG)**. The count of the top repeated n-gram in the translation is greater than the count of the top repeated source n-gram by at least t (in their work, $n = 4$ and $t = 2$);
- **Repeated targets (RT)**. The translation is repeated for multiple unique source sentences.

Alternatively, Berard et al. (2019) noticed that attention patterns in which most attention mass is concentrated on the source `EOS` token are often associated with a model ignoring the source and generating a hallucinatory translation. As such, they proposed `Attn-ign-SRC` for detection of hallucinations:

- **Attn-ign-SRC**: this method (*attention ignores source*) consists of computing the proportion of source words with a total incoming attention mass lower than a threshold λ:

$$\frac{1}{N}\sum_{j=1}^{N} \mathbb{1}\left[(\boldsymbol{\Omega}^{\top}\mathbf{1})_j < \lambda\right]. \tag{2.14}$$

where $\boldsymbol{\Omega}$ is the cross-attention weight matrix respective to the average of the cross-attention heads of the decoder's last layer.[23] In their work, the authors set λ to 0.2.

23: We follow the notation from the previous section: $\boldsymbol{\Omega} \in [0,1]^{T \times N}$, where N and T are the source and target text length, respectively.

An exception from the general framework of natural hallucination detection is the work by Zhou et al. (2021) who *learn* to detect token-level hallucinations. Specifically, the authors create synthetic data where they randomly corrupt some tokens in a translation and reconstruct them with the BART model (Lewis et al., 2020). Then, the authors fine-tune a pretrained language model to identify the replaced tokens.

2.5.2 Understanding the phenomenon of hallucinations

Many works have provided important insights towards better understanding of hallucinations by analyzing the phenomenon under different lenses: construction of the training data (Raunak et al., 2021), decoding strategies (Wang and Sennrich, 2020; Müller and Sennrich, 2021), cross-attention (Berard et al., 2019; Raunak et al., 2021; Voita et al., 2021) and global source token influence patterns (Dale et al., 2022; Ferrando et al., 2022a; Xu et al., 2023a), memorization (Raunak et al., 2021), domain shift (Müller et al., 2020; Wang and Sennrich, 2020), and others. Below we will detail findings on the relevant directions for the work in this thesis.

- **Noise in the training data.** Raunak et al. (2021) examine the impact of different noise patterns in the training data on the prevalence of natural hallucinations. The authors create different types of noise patterns for the experiment: one where unique source sentences are paired with repeated unrelated target sentences, and another where repeated source sentences are paired with unique unrelated target sentences. The findings suggest a strong relationship between these patterns of noise and the types of hallucinations produced. Specifically, when unique source sentences are paired repeatedly with unrelated targets, the model tends to generate detached hallucinations that directly copy references from the training set.[24] Conversely, when the same source sentence is repeatedly paired with unique unrelated targets, the model tends to generate a higher proportion of oscillatory hallucinations.

24: This is the main motivation for constructing the RT detection method.

- **On memorization.** Apart from the repetition and memorization of reference translations induced by noise patterns in the training data, Raunak et al. (2021) also explore the hypothesis that samples memorized by a NMT model are more likely to produce hallucinations when perturbed. Through empirical analysis, the authors found a strong positive correlation between the level of memorization and the frequency of hallucinations under perturbation. In other words, the samples that the model memorized the most were also those most likely to produce hallucinations when perturbed.

- **On domain shift.** Previous work has indicated a substantial increase in hallucinatory rates when NMT models are evaluated under domain shift. For a specific model, hallucination rates may fluctuate

from 1 to 2% in in-domain data, while surpassing 30% in out-of-domain data (Müller et al., 2020; Wang and Sennrich, 2020).

▸ **On attention patterns and source sentence influence.** Research on hallucinations in NMT has consistently suggested that hallucinations are associated with anomalous cross-attention patterns (Lee et al., 2018a; Berard et al., 2019; Raunak et al., 2021). Cross-attention of hallucinated translations is usually characterized by more fixed, lower-entropy distributions, in which almost all the probability mass is concentrated on a few tokens or the source end-of-sentence (EOS) token.[25] These patterns suggests a potential decoupling between the encoder and decoder, whereby the decoder ignores context provided by the encoder. In work concurrent to this thesis, Ferrando et al. (2022b) introduced ALTI+, a method for calculating global token contributions in encoder-decoder Transformer-based models, and further validate that when a NMT model is hallucinating, the source tokens contributions are significantly smaller than for normal *healthy* translations. Lastly, Xu et al. (2023a), also concurrently to our work, show that the distribution of the source contributions is more *static*, i.e. the source contribution distribution shifts over generation steps varies little and is more fixed, on hallucinated translations than that on non-hallucinated translations.

25: This observation has inspired the construction of the Attn-ign-SRC detection method.

2.6 Conclusion

We have introduced the relevant background to understand the main concepts and problems studied in this thesis, laying a foundation that will be built upon in the subsequent chapters. As we present our novel research in the following chapters, we will revisit some of the notions herein presented and introduce additional ones that may be relevant for the chapter at hand.

Part II

SAFEGUARDS

Highlights and Structure

Hallucinations in Machine Translation Models

▸ **Highlights and Contributions** The following chapters present a comprehensive study on hallucinations in machine translation models, focusing on three main research directions: **detection**, **analysis**, and **mitigation**.

Our investigation begins by establishing a solid foundation for studying hallucinations in classical bilingual translation models. We conduct a rigorous comparison of different hallucination methods, create a human-annotated hallucination dataset, and develop a simple model-based method to mitigate hallucinations. Building on these foundational findings, we propose a novel hallucination detection method based on Optimal Transport theory, offering a simple yet effective approach to this challenge.

Moving beyond bilingual models, we extend our analysis to massively multilingual translation models and contemporary generative large language models. We examine hallucination properties across over 100 language pairs, including both English- and non-English-centric pairs. We study how hallucination properties and mitigation strategies vary across model scale, families, and architectures.

▸ **Publications** The work presented in this part is based on the following published publications:

- **Nuno M. Guerreiro**, Elena Voita, and André F.T. Martins. *Looking for a Needle in a Haystack: A Comprehensive Study of Hallucinations in Neural Machine Translation*. In Proc. EACL 2023.
- **Nuno M. Guerreiro**, Pierre Colombo, Pablo Piantanida, and André F.T. Martins. *Optimal Transport for Unsupervised Hallucination Detection in Neural Machine Translation*. In Proc. ACL 2023.
- **Nuno M. Guerreiro**, Duarte M. Alves, Jonas Waldendorf, Barry Haddow, Alexandra Birch, Pierre Colombo, and André F.T. Martins. *Hallucinations in Large Multilingual Translation Models*. In Transactions of the Association for Computational Linguistics, 2023.

Looking for a Needle in a Haystack: A Study of Hallucinations in NMT

3

In this chapter, we set foundations for the study of natural hallucinations in neural machine translations. We take a step back from previous work and, instead of considering perturbed settings for which hallucinations are more frequent, we consider a *natural scenario* and face the actual problem of identifying a small fraction of hallucinations (a "needle") in a large number of translated sentences (a "haystack"). Then, we provide a rigorous comparison among hallucination detection methods. Apart from analysing those proposed in previous work (e.g., heuristics based on anomalous encoder-decoder attention), we also propose to use simple model uncertainty measures as detectors. For each of these methods, we select examples marked as hallucinations, put them together, and gather human annotations. As a result, we introduce a corpus of 3415 structured annotations for different NMT pathologies and hallucinations. We use this corpus for analysis and show that, in preventive settings where high recall is desirable, previously proposed methods are mostly inadequate, and filtering according to standard sequence log-probability performs the best. In fact, it performs on par with the state-of-the-art COMET (Rei et al., 2020a) which uses reference translation and thus cannot be used in most real-world on-the-fly applications. Surprisingly, its reference-free version COMET-QE (Rei et al., 2020b), which was shown to generally perform on par with COMET (Freitag et al., 2021b; Kocmi et al., 2021b), substantially fails to penalise the severity of hallucinations. Overall, methods targeting the phenomena are largely unfit, quality estimation systems fail, and sequence log-probability, i.e. a byproduct of generating a translation, turns out to be the best.

Apart from our analysis of detection methods, we propose DeHallucinator, a method for mitigating hallucinations at test time. At a high level, we first apply a lightweight hallucination detector and then, if a translation is flagged, we try to overwrite it with a better version. For this, we generate several MC-dropout hypotheses (Gal and Ghahramani, 2016), score them with some measure, and pick the highest-scoring translation as the final candidate. With this approach, the proportion of correct translations among the ones flagged by the detector increases from 33% to 85%, and the hallucinatory rate decreases threefold.

Overall, we show that, in preventive settings:

- previously proposed hallucination detectors are mostly inadequate;
- quality estimation techniques fail to distinguish hallucinations from less severe errors;
- sequence log-probability is the best hallucination detector and performs on par with reference-based COMET;
- our DeHallucinator significantly alleviates hallucinations at test time.

Additionally, we release our annotated dataset along with the model, training data, and code.

3.1 Taxonomy of Translation Pathologies

Choosing a good taxonomy is a compromise between simplicity (which minimizes annotation effort) and comprehensiveness. Thus, generic quality assessment taxonomies, such as MQM (Lommel et al., 2014b), might be unfit or too complex when we focus only on critical errors and hallucinations. For hallucinations, in turn, previous work used multiple, often overlapping, definitions (Lee et al., 2018b; Raunak et al., 2021; Zhou et al., 2021; Raunak et al., 2022a). The taxonomy we build here is rather general: it covers categories considered previously (Lee et al., 2018b; Raunak et al., 2021) and others not reported before. For a broader discussion on the taxonomy of hallucinations in machine translation, and how it differs from other natural language generation tasks, refer to Ji et al. (2022).

3.1.1 Hallucinations

To distinguish hallucinations from other errors, we rely on the idea of detachment from the source sequence. From this perspective, other critical errors such as mistranslation of named entities are not considered as hallucinations. In Section 3.5 and Section 3.7.2, we show that properties of hallucinations differ a lot from these other errors and thus our taxonomy is very reasonable.

- **Oscillatory hallucinations.** These are inadequate translations that contain erroneous repetitions of words and phrases.

- **Largely fluent hallucinations.** These are largely fluent translations that are unrelated to the content of the source sequence. Previous work assumed they always bear *no relation at all* to the source content (Lee et al., 2018b; Raunak et al., 2021). However, we find that a

Table 3.1: Examples of natural (detached and oscillatory) hallucinations.

Strongly Detached hallucination	
Source	Tickets für Busse und die U-Bahn ist zu teuer, vor allem in Stockholm.
Reference	Tickets for buses and the subway is too expensive, especially in Stockholm.
Hallucination	The hotel is located in the centre of Stockholm, close to the train station.
Fully Detached hallucination	
Source	Handys, die bis auf Wäschewaschen und Staubsaugen scheinbar alles können.
Reference	Mobile phones that can practically do everything except clean the laundry and vacuum clean.
Hallucination	The staff were very friendly and helpful.
Oscillatory hallucination	
Source	In dieser Zeit stürzt sich Murnau bereits wieder in Theaterproben, allerdings widmet er sich nicht mehr der Schauspielerei, sondern der Regie.
Reference	During this period, Murnau once again dedicates his time to theatre rehearsals; however, this time not as an actor, but as a director.
Hallucination	Murnau was born in Murnau, Germany. He was born in Murnau, Germany.

large proportion of fluent hallucinations partially support the source. Therefore, we also consider severity of a hallucination and distinguish translations that are *fully* detached from those that are *strongly* (but not fully) detached.

Note that oscillatory hallucinations can also be either fully or only partially detached, but since these hallucinations are less frequent, in what follows we do not split them by severity. We show examples of these hallucination types in Table 3.1.

3.1.2 Translation errors

- **Undergeneration.** These are incomplete translations that do not cover part of the source content. This problem is often studied in isolation (Koehn and Knowles, 2017; Kumar and Sarawagi, 2019; Stahlberg and Byrne, 2019). Undergenerations are sometimes considered as hallucinations (Lee et al., 2018b) but we do not consider them so in our work.

- **Mistranslation of named entities.** Appropriately translating named entities (e.g. names, dates, etc.) is also a known difficulty of NMT systems (Ugawa et al., 2018; Li et al., 2021; Hu et al., 2022). Note that for production systems, this error is rather critical. However, we do not consider it as an hallucination as it does not show detachment from the source but rather an incorrect attempt to translate part of its content.

- **Other errors.** These are other incorrect translations that do not fit the categories above. They may include errors related to part of speech, word order, and others. For an extensive analysis on machine translation errors, refer to Vilar et al. (2006).

3.2 Hallucination Detection Methods

Approaches to hallucination detection generally aim to find low-quality translations that may also satisfy additional constraints. Previous work either relied only on quality filtering (Lee et al., 2018b; Müller and Sennrich, 2021; Raunak et al., 2021), or only on heuristics (Berard et al., 2019), or a combination of the two (Raunak et al., 2021). We stick to this general form and consider different quality filters and heuristics.

3.2.1 Quality filters

Quality filters come in two forms: reference-free and reference-based filters. The latter rely on reference translations, while the former do not.

- **Reference-free methods.** We use the state-of-the-art COMET-QE (Rei et al., 2020b) for its superior performance compared to other metrics (Mathur et al., 2020; Freitag et al., 2021b; Kocmi et al., 2021b).

▸ **Reference-based methods.** Previous work used adjusted BLEU or CHRF2 scores of less than 1% as a standalone criteria (Lee et al., 2018b; Müller and Sennrich, 2021; Raunak et al., 2021; Yan et al., 2022). In this work, we analyse CHRF2 because it is more suitable for sentence-level evaluation. In addition to this lexical metric, we also consider neural COMET (Rei et al., 2020a), a state-of-the-art reference-based metric (Kocmi et al., 2021b).

Note that in real-world on-the-fly applications, detecting hallucinations is needed when references are not available. Thus, we use reference-based methods to estimate an upper bound for performance of the other methods.

3.2.2 Hallucination detection heuristics

Previously Used Heuristics

▸ **Binary-score Heuristics.** These were used by Raunak et al. (2021) to detect oscillatory and fully detached hallucinations. Given a corpus of source-translation pairs, a translation is flagged as an hallucination if it is in the set of 1% lowest-quality translations, and if:

- **Top n-gram count (TNG)**. The count of the top repeated n-gram in the translation is greater than the count of the top repeated source n-gram by at least t (in their work, $n = 4$ and $t = 2$);
- **Repeated targets (RT)**. The translation is repeated for multiple unique source sentences.

1: These patterns are respective to the average of the cross-attention heads of the decoder's last layer.

▸ **Anomalous decoder-encoder attention.** Attention patterns[1] in which most attention mass is concentrated on the source EOS token are often associated with a model ignoring the source and generating a hallucination (Lee et al., 2018b; Berard et al., 2019; Raunak et al., 2021). We consider two different criteria targeted to find this pattern:

- **Attn-to-EOS**: the proportion of attention paid to the EOS source token;
- **Attn-ign-SRC**: the proportion of source words with a total incoming attention mass lower than 0.2. This was used as a data filtering criterion in Berard et al. (2019).

Uncertainty-Based Heuristics

Now we describe the uncertainty measures we propose to use as hallucination detectors. Previously, these were used to improve quality assessments (Fomicheva et al., 2020; Zerva et al., 2021).

▸ **Sequence log-probability (Seq-Logprob).** For a trained model $p_\theta(\boldsymbol{y}|\boldsymbol{x})$ and a generated translation $\boldsymbol{y}$, Seq-Logprob (i.e., model confidence) is the *length-normalised* sequence log-probability:

$$\frac{1}{L}\sum_{t=1}^{L}\log p_\theta(y_t \mid \boldsymbol{y}_{<t}, \boldsymbol{x}). \tag{3.1}$$

We hypothesise that when hallucinating, a model is not confident.

- **Dissimilarity of MC hypotheses (MC-DSim).** This method measures how the original hypothesis y disagrees with hypotheses $\{\boldsymbol{h}_1, \ldots, \boldsymbol{h}_N\}$ generated in stochastic passes. For the same source sentence, we generate these new hypotheses using Monte Carlo (MC) Dropout (Gal and Ghahramani, 2016). Then we evaluate the average similarity:

$$\frac{1}{N}\sum_{i=1}^{N} \text{SIM}(\boldsymbol{h}_i, \boldsymbol{y}). \tag{3.2}$$

Different similarity measures can be used in place of SIM (e.g. METEOR (Banerjee and Lavie, 2005), BERTScore (Zhang* et al., 2020), etc.). We follow previous work and use METEOR with $N = 10$ (Fomicheva et al., 2020; Zerva et al., 2021).

3.2.3 Trained hallucination detection model

An exception from the general framework of hallucination detection is the work by Zhou et al. (2021) who *learn* to detect token-level hallucinations. Specifically, the authors create synthetic data where they randomly corrupt some tokens in a translation and reconstruct them with the BART model Lewis et al., 2020. Then, the authors fine-tune a pretrained language model to identify the replaced tokens.

- **TokHal-Model.** We evaluate the proportion of tokens that are predicted to be hallucinated and use this as a detection score.

3.2.4 Binary vs continuous scores

The methods above fall into two categories: *binary-score* and *continuous-score* heuristics. The former (only TNG and RT) output a value in $\{0, 1\}$, whereas the latter output a value in $\mathbb{R}$ and the prediction is made depending on a chosen threshold.

3.3 Experimental Setting

- **Model.** We use Transformer base (Vaswani et al., 2017) (hidden size of 512, feedforward size of 2048, 6 encoder and 6 decoder layers, 8 attention heads). The model has approximately 77M parameters.

- **Data.** We use the WMT2018 DE-EN news translation data excluding Paracrawl (Bojar et al., 2018). We filter our data using language identification and simple length-heuristics described in Tran et al. (2021). We encode the data with byte-pair encoding (Sennrich et al., 2016b) using the SentencePiece framework (Kudo and Richardson, 2018). We set the vocabulary size to 32k and compute joint encodings and vocabulary. We randomly choose 2/3 of the final dataset (5.8M sentence pairs) for training and use the remaining 1/3 as a held-out set for analysis. For validation, we use the *newstest2017* dataset.

- **Held-out Data Filtering.** We are mainly interested in hallucinations produced for clean data. Since our held-out data comes from the WMT2018 training dataset and thus can be noisy, we filter it using Bicleaner, the filtering tool used in official releases of filtered ParaCrawl data (**10.1162/tacl_a_00447**; Sánchez-Cartagena et al., 2018; Ramírez-Sánchez et al., 2020). Following previous work, we exclude examples with a score below 0.5 and end up with about 1.3M examples.

- **Training and Inference.** Models are trained for 250K updates with a batch size of about 32K tokens. We set dropout to 0.3. Similarly to (Vashishth et al., 2019), we train our model using the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$ and use an inverse square root learning rate scheduler with initial value 5×10^{-4}, and a linear warm-up in the first 4000 steps. At inference time, we produce translations using beam search with a beam of 5. We validate our models during training using SacreBLEU (Post, 2018), and we choose the checkpoint based on best BLEU in validation. We train and perform inference on top of the Fairseq framework (Ott et al., 2019).

3.4 Hallucinations Dataset

To analyse the effectiveness of hallucination detection criteria, we pick a subset of examples that are likely to be hallucinations, and obtain fine-grained annotations from professional translators.

3.4.1 Data for annotation

Our data selection is motivated by two goals: (i) find as many hallucinatory translations as possible – to analyse hallucinations, (ii) pick some translations from a long tail of hallucination detection predictions – to analyse the behaviour of these detection methods. Thus, we first pick 250 worst-scored samples for each heuristic and quality filter (including binary assignments obtained through TNG and RT). Next, we turn to a broader set of samples and consider translations whose scores fall below a chosen percentile for a given method, i.e. we consider long tails of the scores.[2] For in-domain settings, previous work reported hallucinatory rates of 0.2 – 2%. However, these rates were either obtained on noisy and/or low-resource data or using weaker models. Therefore, in our cleaner in-domain setting with a stronger model, we expect the hallucination rate to lie in the lower end of the indicated range. Thus, we consider approximately 0.4% of the worst scores (which amounts to 5000 flagged translations for each criteria).[3] From these, we sample 250 examples and add them to the dataset. In total, we end up with 3415 examples for annotation.[4]

3.4.2 Guidelines and annotation

We perform a rigorous and comprehensive manual annotation procedure with professional translators, in order to make sure that we are reliably analysing hallucinated translations.[5] First, the translators were

2: From now on, we refer to examples contained in a long tail of a method as "flagged" or "detected" by this method.

3: The threshold for prediction is consistent with this percentile.

4: Note that a sample originally obtained from the worst scores or sampled from the long tail of a given method may belong to the long tail of another method.

5: Our translators were hired through Upwork and they were informed about the academic purposes of the data annotation process. All translators hired for this study reside in Europe. We paid a fair wage (25-30 USD per hour) – well above both the US federal minimum and the average EU minimum wage – and inspected their work for quality.

asked to be familiar with our task by reading our provided annotation examples, along with detailed annotation instructions. Then they had to pass a test to show they can recognize different pathologies in different translations. We selected the two best translators to gather the annotations. After being hired, we ran three one-hour tutorial sessions to explain the task thoroughly and to clarify any possible questions. During the annotation process, we made sure to be promptly available to answer any question from the translators.

- **GUIDELINES.** We make available the full guidelines used by the translators along with all other resources in the project repository. In short, the annotators were presented with a source sentence and a model-generated hypothesis and asked to deem that translation as correct (COR) or incorrect. If incorrect, they were prompted to answer a series of yes/no questions, regarding the presence of specific hallucinatory pathologies: oscillations (OSC), strong detachment (SD) and full detachment (FD). We also asked annotators to flag critical errors such as named-entity mistranslations (NE) and under-generated translations (UG).

- **INTER-ANNOTATOR AGREEMENT.** To determine the reliability of our annotations, we asked both our translators to annotate a set of 400 randomly sampled translations. For all hallucinatory categories but SD, the annotators achieved – according to Cohen's kappa coefficient (Cohen, 1960) – almost perfect agreement. For all other categories, moderate to substantial agreement was obtained. This confirms that our data conforms very well to our instructions. The agreement scores for each category are displayed in Table 3.2.

COR	UG	NE	OSC	SD	FD
0.62	0.73	0.42	0.86	0.45	0.89

Table 3.2: Fleiss's Kappa inter-annotator agreement scores (↑) for the different categories translators were prompted to identify.

3.5 High-level Overview of the Dataset

3.5.1 General statistics

Figure 3.1 gives a structured overview of dataset statistics. First, we see that while we picked translations that are likely to be pathological, 60% of the dataset consists of correct translations. This highlights that with the existing methods, finding poor translations reliably is still challenging. Next, note that most of the incorrect translations have translation errors that are not severe enough to be deemed hallucinations. This agrees with the view that hallucinations lie at the extreme end of the spectrum of MT pathologies (Raunak et al., 2021). Finally, the results of the annotation confirm that our data selection is very reasonable. Indeed, while previous work has reported hallucinatory rates of 0.2 – 2% in in-domain settings, we see that, for a reasonably numbered collection of examples, our hallucination rate is substantially higher (9%) – 294 hallucinations among the 3415 translations. In Figure 3.1, we also show the method-specific statistics of human annotation results for each heuristic and quality filter. Unsurprisingly, the long tails of each method display different characteristics. For example, almost all translations flagged by Attn-to-EOS are correct, whereas the proportion of good translations flagged by COMET-QE or Seq-Logprob is rather small. We will analyse this further in Section 3.6.

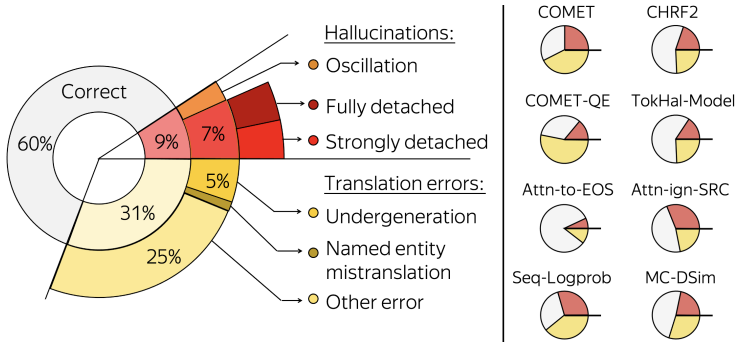
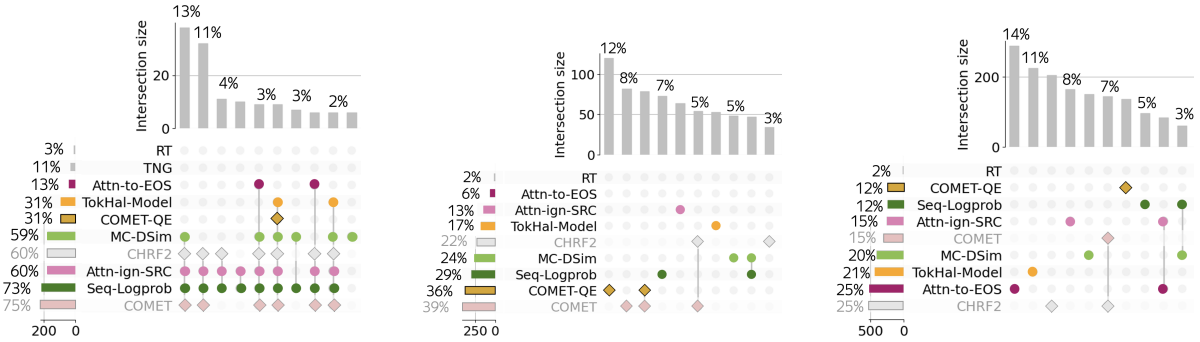


Figure 3.1: Overall (left) and method-specific (right) statistics of human annotation results. Method-specific statistics show the percentages of correct translations (grey), translation errors (yellow) and hallucinations (red) among the examples flagged by each method.



(a) Hallucinations **(b)** MT errors **(c)** Correct translations

Figure 3.2: Structure of the sets of translations flagged by the considered methods. Horizontal bars show the proportion of examples flagged by each method among all translations of the considered category (e.g., 75% of hallucinations in the dataset are flagged by COMET). Each vertical bar shows the size for the set of translations that are (i) flagged by all the methods marked in the corresponding column and (ii) not flagged by any of the rest (e.g., 13% of the hallucinations are detected simultaneously by MC-DSim, chrF2, Attn-ign-SRC, Seq-Logprob and COMET and not detected with the remaining methods); only the top intersections are shown. Quality filters are shown with diamond marks, and detection heuristics – with circles. Methods requiring reference translations are shaded.

3.5.2 MT Errors vs hallucinations

Now let us look separately at the sets of examples with hallucinations and other translation errors. Figure 3.2 shows the structures of these sets with respect to interactions of the different criteria.

We see that it is reasonable to consider hallucinations separately from other errors: patterns in Figures 3.2a and 3.2b are substantially different. For example, for translation errors, COMET-QE performs well, being on par with reference-aware COMET (Figure 3.2b). However, for hallucinations, it identifies less than half the amount of examples identified not only by COMET, but even by simple uncertainty-based heuristics (Figure 3.2a). Furthermore, Figure 3.2 reveals a significant difference between interactions of different criteria for hallucinations and for other translations: hallucinations in our data are most often flagged by multiple criteria simultaneously (e.g. Seq-Logprob flags the majority of hallucinations detected with Attn-ign-SRC and MC-DSim), whereas most MT errors and correct translations are flagged by a single method. This difference in patterns supports our choice of taxonomy: properties of hallucinations are very different from those of other less severe errors.

3.6 Analysing Detection Criteria

In this section, we provide a comprehensive analysis of the performance of the heuristics and quality filters introduced in Section 3.2.

3.6.1 Quality filters

Here we start by analysing reference-based methods, namely COMET and CHRF2, and then turn to the reference-free COMET-QE. Overall, our results show that reference information is helpful, while COMET-QE fails to penalise hallucinations.

▸ **Reference information helps detection.** Figure 3.2a shows that leveraging reference information helps detecting hallucinations: COMET detects more hallucinations than any of the other methods. As expected, lexical-based CHRF2 is significantly worse than neural-based COMET. In fact, it lags behind several heuristics (e.g., Seq-Logprob).

We also explore whether previously proposed methods for detecting hallucinations under domain shift are appropriate in our cleaner in-domain setting. Specifically, we follow Müller and Sennrich (2021) and consider translations with CHRF2 score lower than 1%. Strikingly, for our clean setting, this approach is inadequate: in the entire held-out set of 1.3M examples, it flagged only 2 translations. This suggests that methods suitable for noisy settings (e.g., domain shift) might not be applicable in settings where models are less likely to hallucinate.

▸ **COMET-QE fails to penalise hallucinations.** From Figure 3.1 (right) we see that, as expected, most of the translations flagged by COMET-QE are incorrect. However, the vast majority of them are not hallucinations. Indeed, Figure 3.2a shows that COMET-QE is one of the worst hallucination detectors, meaning that it fails to rank by the severity of a translation pathology. This supports the hypothesis made in previous work: since quality estimation models are mostly trained on data that lacks negative examples, they may be inadequate for evaluating poor translations (Sudoh et al., 2021; Takahashi et al., 2021).

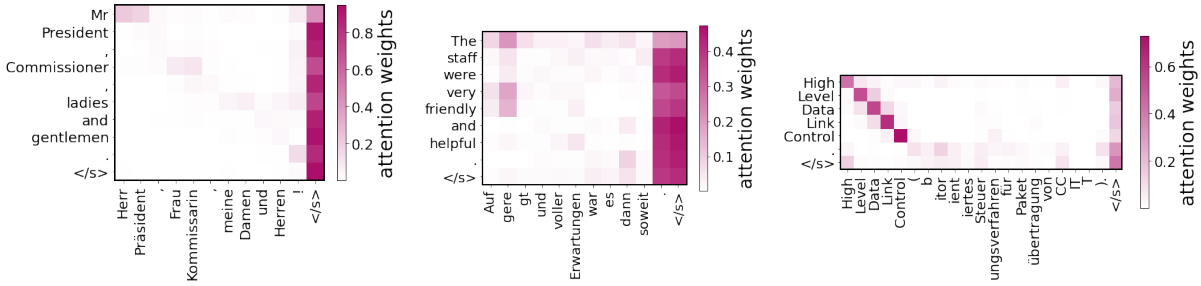
Overall, among the considered quality filters, only COMET may be used as a hallucination detector. However, it is important to keep in mind that, in on-the-fly applications, detecting hallucinations is needed when references are not available, rendering reference-based methods not applicable.

3.6.2 Detection heuristics

The results in Section 3.6.1 leave a relevant gap: where do we turn to when references are not available? Is there any information, besides quality, that may help detecting hallucinations? Our evidence suggests that in preventive settings, previously proposed heuristics are mostly inadequate, and uncertainty may be the answer for the questions we pose.

Table 3.3: Translations flagged by TNG and RT, as well as overall dataset statistics.

Heuristic	Correct	MT Errors	Hallucinations OSC	SD	FD
TNG	0	0	32	0	0
RT	18	19	2	1	7
All dataset	2048	1073	86	90	118



(a) A correct translation, flagged by Attn-to-EOS

(b) A hallucination, flagged by Attn-ign-SRC

(c) An undergeneration, flagged by Attn-ign-SRC

Figure 3.3: Examples of attention maps flagged by attention-based heuristics.

▸ **Binary-score heuristics perform the worst.** Table 3.3 shows the number of detected translations for Top n-gram count (TNG) and Repeated targets (RT). These heuristics are targeted to identify oscillatory and fully detached hallucinations, respectively. We see that while TNG obtains perfect precision, it fails to identify more than half of the oscillatory translations. RT, in turn, performs poorly across the board: only a few hallucinations are detected, and a significant proportion of flagged translations turn out to be correct. Moreover, Figure 3.2 shows that even if we join sets of translations detected by TNG and RT (as done in Raunak et al. (2021)), altogether we get *fewer hallucinations than almost any other considered method*. Thus, in preventive settings, these methods are highly inadequate.

▸ **Anomalous attention is mostly *not* hallucination.** Figures 3.1 and 3.2 show that behaviors of Attn-to-EOS and Attn-ign-SRC are significantly different. First, Attn-to-EOS *is not indicative of hallucinations*. Indeed, attention patterns in which most attention mass is concentrated on the EOS token largely correspond to correct translations. On the other hand, Attn-ign-SRC performs well and is second only to uncertainty-based Seq-Logprob. Such a difference in performance is surprising: both methods are motivated by a common belief that if almost all the attention mass is concentrated on the source EOS token, a translation is likely to be a hallucination (Berard et al., 2019; Raunak et al., 2021). In fact, both methods were designed to identify this specific pattern. However, patterns identified with Attn-ign-SRC span from attention mass coming to various uninformative tokens (e.g., punctuation) to examples where attention is mostly diagonal (typically, these correspond to undergenerations). We show examples of such attention maps in Figure 3.3. Overall, the results highlight a disparity between what is *believed to indicate* hallucinations and what *actually indicates* them.

Note that while Attn-ign-SRC performs relatively well, it should be used with caution: attention-based heuristics rely on the assumption

that attention patterns reflect model reasoning. This assumption is not entirely reliable: although there is evidence that attention can play recognizable roles (Voita et al., 2018, 2019), a lot of work questions attention explainability (Jain and Wallace, 2019; Serrano and Smith, 2019; Wiegreffe and Pinter, 2019; Bastings and Filippova, 2020; Pruthi et al., 2020). Since hallucinations identified by Attn-ign-SRC are overwhelmingly contained among the ones identified by Seq-Logprob (Figure 3.2a), we recommend using the latter instead.

- **TokHal-Model is unfit for natural hallucinations.** Let us recall that the model used for the TokHal-Model scores was trained to identify replaced tokens in corrupted translations that are fluent and do not differ much from the original ones (Zhou et al., 2021). This means that during training, the model was unlikely to observe highly pathological translations that reflect the types of hallucinations produced by actual NMT systems. This raises several concerns when using this model in our setting. For example, it might incorrectly flag adequate tokens such as synonyms or paraphrases. What is more, since severely flawed examples are mostly out of distribution for the model, labels predicted for such translations may be unreasonable.

  The results confirm our concerns: Figures 3.1 and 3.2a show that (i) the vast majority of translations flagged by TokHal-Model are correct and (ii) it is one of the worst hallucination detectors.

- **Model confidence may be all you need.** Figure 3.2 shows that Seq-Logprob is *the best heuristic* and performs on par with reference-based COMET. This means that the less confident the model is, the more likely it is to generate an inadequate translation. This agrees with some observations made in previous work on quality estimation (Fomicheva et al., 2020). Interestingly, such performance of the method contrasts with its simplicity: Seq-Logprob scores are easily obtained as a by-product of generating a translation. This distinguishes the method from all the rest that require additional computation (e.g., corpus-level search for RT or generating multiple hypotheses for MC-DSim).

- **MC-DSim: hallucinations are mostly unstable.** The intuition behind this heuristic is simple: when faced with a source sentence for which a good translation is not immediate, the set of hypotheses the model "keeps at hand" may be very diverse. Indeed, MC-DSim performs relatively well and identifies a good proportion of hallucinations (Figures 3.1, 3.2). Thus, most hallucinations are *unstable*: dissimilarity of MC hypotheses helps to identify them.

  Note that while MC-DSim is not the best choice for detection, later we will show that the intuition behind this method is very helpful for alleviating hallucinations at test time (Section 3.8).

3.6.3 Combining heuristics and quality filters

Intersecting a set of translations obtained via a heuristic with the bottom scored translations according to a quality filter was introduced in Raunak et al. (2021). The motivation for this was to avoid incorrectly flagging good translations as hallucinations (e.g., without this intersection, RT flags good-quality paraphrases). Intuitively, this idea is very

reasonable: hallucinations are indeed incorrect translations. However, implementing this in practice requires a good quality estimation model, specifically for ranking poor translations. Unfortunately, we showed in Section 3.6.1 that for this purpose, even the state-of-the-art COMET-QE is largely inadequate. This means that using such quality estimates may lead to filtering out a lot of hallucinations, which is not desirable in preventive settings. Our results show that this is exactly the case: e.g., filtering with COMET-QE leads to losing nearly 80% of hallucinations detected by Seq-Logprob. All in all, such an intersection generally does more harm than good.

3.7 Analysing Hallucination Pathologies

In this section, we look at translation pathologies in isolation and show that the behavior of detection methods varies depending on the type of pathology. For example, for a given pathology, some methods may be specialised, whereas others may fail.

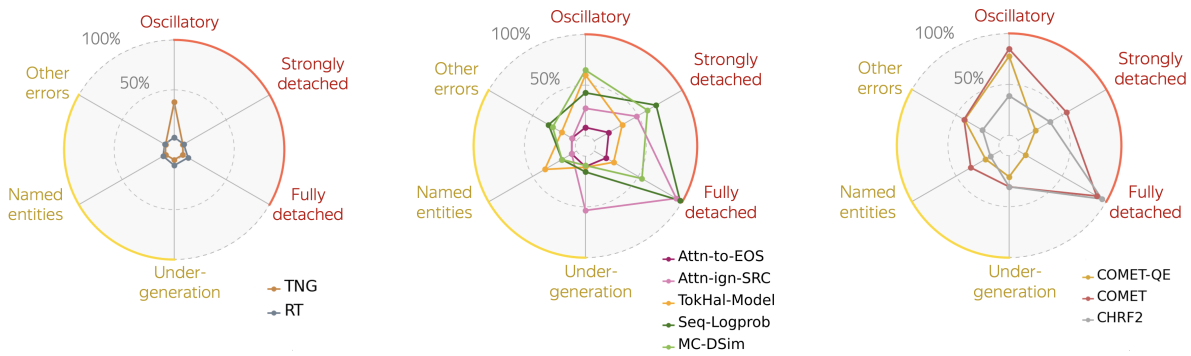
3.7.1 Hallucinations

- **Fully detached hallucinations.** Figure 3.4 shows that these hallucinations are easily detected by several methods, e.g. Seq-Lobprob, Attn-ign-SRC, COMET, CHRF2. This is not surprising: intuitively, the most severe pathology should be the easiest to detect. However, COMET-QE *fails to identify almost all these hallucinations*. While COMET-based metrics are known to not penalise enough certain types of errors (e.g., discrepancies in numbers and named entities; see Amrhein and Sennrich (2022a) and Raunak et al. (2022a)), such poor performance for completely inadequate translations is highly unexpected. This calls for further research on the behavior of quality estimation models.

  On a more general note, previous work suggested that fully detached hallucinations emerge as exact copies of references from training data (Raunak et al., 2021). We validate this hypothesis and find the contrary: out of the 44 unique translations marked as fully detached from the source, only 4 are exact copies of references in the training data. Nevertheless, when looking at these sentences more closely, we see that they do contain large substrings that are seen frequently during training. Therefore, fully detached hallucinations are indeed likely to be traced back to the training data, but they *emerge in non trivial ways and not necessarily as exact copies*. This can be seen as one more evidence that, when dealing with memorisation in language models, it is necessary to consider not just full copies in the training data but also near-duplicates (Lee et al., 2022).

- **Strongly detached hallucinations.** As expected, this pathology is harder to detect than fully detached hallucinations (Figure 3.4). However, the trends are largely similar: for example, COMET-QE fails again, and Seq-Logprob performs best and outperforms even reference-based COMET.

- **Oscillatory hallucinations.** The method specifically developed to detect this hallucination type (Top n-gram count, TNG) performs worse



(a) Binary-score heuristics **(b)** Continuous-score heuristics **(c)** Quality filters

Figure 3.4: Distribution of the translations flagged by each method conditioned on each pathology.

than most of the other methods. Among the rest, COMET performs best. Interestingly, in contrast to previous observations, COMET-QE performs well, being on par with COMET.

3.7.2 Less severe translation errors

We notice that some of the detection methods are specialised on specific pathologies. For example, Attn-ign-SRC is by far the best for detecting omissions. This is expected: an omission does not cover part of the source sentence, thus a significant proportion of source tokens receives little attention mass (see example in Figure 3.3c). For named entity errors, the best heuristic is TokHal-Model. This mirrors the discussion in Section 3.6.2: while severe errors (i.e., hallucinations) fall out of distribution for this model, mistranslations of short phrases are more in line with the model's training.

On a different note, MC-DSim performs much better for hallucinations than for less severe pathologies (Figure 3.4b). This again points to hallucinations being more unstable than other errors. We continue discussing this in the next section.

3.8 DeHallucinator: Mitigating Hallucinations at Test Time

In previous sections, we saw that hallucinations are more unstable than other translations: for them, generated MC-dropout hypotheses tend to vary greatly. This motivates us to look more closely into these hypotheses: are any of these translations *not* hallucinations? Answering this question not only gives insight into the inner workings of hallucinating NMT models, but also leads to an interesting practical application – overwriting hallucinations at inference time. This is of utmost importance for production systems where hallucinations have a deeply compromising effect on user trust.

▸ **Whenever flagged, overwrite with better.** Intuitively, our idea is similar to hybrid pipelines when a machine-generated translation is first passed to a quality estimation system and then, if needed, is

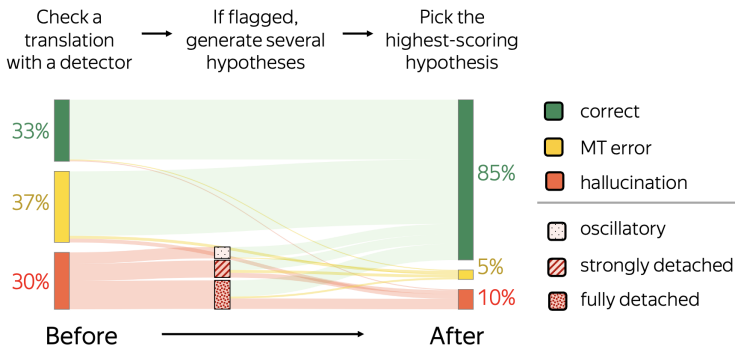


Figure 3.5: Our pipeline scheme along with results.

corrected by human translators. In our case, we first apply a hallucination detector and then, if a translation is flagged, we try to overwrite it with a better translation (Figure 3.5). For this, we generate several MC-dropout hypotheses, score them with some measure, and pick the highest-scoring translation as a final candidate (in the spirit of reranking approaches (Shen et al., 2004; Lee et al., 2021; Fernandes et al., 2022b; Freitag et al., 2022a)).

The general pipeline above relies on the choice of a hallucination detector and a scoring measure. For the detector, we use the best of the analysed detectors, i.e. Seq-Logprob.[6] For the scoring measure, a natural choice would be a quality estimation system: by construction, these systems are designed to score translations according to quality. However, as we saw earlier, even the state-of-the-art COMET-QE may fail (Section 3.6.1). Therefore, we compare two measures: COMET-QE and Seq-Logprob.

6: In this experiment, we take the translations from our dataset and consider the percentiles defined in Section 3.4.

In this experiment, we randomly choose 200 translations from our dataset flagged by Seq-Logprob. For each, we generate 10 hypotheses with MC-dropout. Then, for the overwritten translations we gather annotations according to our guidelines (Section 3.4). The results are summarized in Figure 3.5. Although we were concerned about COMET-QE because of its low performance when ranking poor translations, we find that for choosing the best hypothesis, it is indeed appropriate and performs better than Seq-Logprob.

▸ **Most hallucinations and errors become correct.** Figure 3.5 shows that most hallucinations are overwritten with correct translations. This is surprising: in most cases, the model is not stuck in a hallucinatory mode and can generate good translations in a small vicinity of model parameters. In this sense, most hallucinations result from "bad luck" during generation and not profound model defect. Note that fully detached hallucinations are the hardest to improve. This makes sense as these are likely to be traced back to (near-)duplicates in the training data and, therefore, they do highlight model anomalies.

Note that in this pipeline, we overwrite not only hallucinations but also other errors and correct translations that were flagged by the detector. This means that our method needs to appropriately handle such translations. From Figure 3.5 we see a pleasant side-effect: our approach overwrites most of the errors with correct translations. Just as importantly, almost all originally correct translations remain correct. Overall, the proportion of correct translations among the ones flagged by the detector increases from 33% to 85%, and the hallucinatory

Table 3.4: Examples of hallucinations of each type that have been overwritten with correct translations.

Correction of a **fully detached hallucination**	
Source	Handys, die bis auf Wäschewaschen und Staubsaugen scheinbar alles können.
Reference	Mobile phones that can practically do everything except clean the laundry and vacuum clean.
Original MT	The staff were very friendly and helpful.
Corrected MT	Mobile phones that seem to be able to do everything except on laundry and dustproofing.
Correction of a **strongly detached hallucination**	
Source	Tickets für Busse und die U-Bahn ist zu teuer, vor allem in Stockholm.
Reference	Tickets for buses and the subway is too expensive, especially in Stockholm.
Original MT	The hotel is located in the centre of Stockholm, close to the train station.
Corrected MT	Buses and metro tickets are too expensive, especially in Stockholm.
Correction of an **oscillatory hallucination**	
Source	In dieser Zeit stürzt sich Murnau bereits wieder in Theaterproben, allerdings widmet er sich nicht mehr der Schauspielerei, sondern der Regie.
Reference	During this period, Murnau once again dedicates his time to theatre rehearsals; however, this time not as an actor, but as a director.
Original MT	Murnau was born in Murnau, Germany. He was born in Murnau, Germany.
Corrected MT	During this time, Murnau began to appear in theater tests again, but he was no longer concerned with acting, but with directing.

rate decreases threefold. In Table 3.4, we show several examples of overwritten hallucinations.

3.9 Conclusions

Dealing with hallucinations is difficult. First, we had to take a step back from previous work and refuse procedures that amplify the problem, as these hinder the behavior of models in their standard settings. After that, we notice that work on detection often relies on assumptions that remained unquestioned (e.g., generic quality measures, targeted heuristics, anomalous attention being suitable detectors). Through extensive experiments, we establish order in detection methods. Surprisingly, despite introduction of several methods specifically targeted for hallucinations, what works best has always been at our disposal: standard sequence log-probability. This suggests that characteristics innate to a model can have a lot of value. In fact, such characteristics are the backbone of our DeHallucinator, a lightweight approach that significantly alleviates hallucinations at test time. This leaves space for future research on model uncertainty, hallucination prevention, understanding where hallucinations come from, among others. For this, we released our corpus with structured annotations along with the model and its training data. Altogether, this allows us to say that we provide solid ground for future study of hallucinations in NMT.

Our work has made several notable impacts on the literature. Concurrent research by Xu et al. (2023a) conducted a comprehensive exploration of hallucinations in NMT models, with their findings regarding attention maps and quality estimation systems aligning with our conclusions. Building directly on our research, Dale et al. (2022) investigated alternative detection methods, including ALTI (Ferrando et al., 2022b), which uses source contribution to generated translations for hallucination identification—an approach we also leverage in Chapter 5. Their work further expanded on our DeHallucinator method,

confirming through human evaluation that MC-dropout combined with QE reranking provides an effective strategy for hallucination mitigation. Following our research, Dale et al. (2023) introduced HalOmi, a multilingual annotated dataset examining hallucination and omission phenomena across 18 translation directions with varying resource levels. This dataset builds upon our taxonomy and directly uses the detectors we studied to construct it. Moreover, our findings regarding sequence log-probability inspired Sennrich et al. (2024) to develop contrastive decoding methods for hallucination mitigation. These methods either search for translations that are probable given the correct input but improbable given random input segments. Furthermore, our dataset has been used in several works for evaluating hallucination detectors in subsequent research, serving both the development of specialized detectors and quality estimation metrics (Guerreiro et al., 2023c; Himmi et al., 2024).

The next chapter builds upon the foundations established in this work to propose a new model-based hallucination detector. Our findings here have shown that model-based detectors can be effective in predicting hallucinations. We expand on this by exclusively focusing on the cross-attention mechanism, developing an approach that better leverages anomalous attention patterns compared to those presented in this chapter.

Optimal Transport for Hallucination Detection

4

In this chapter, we focus on leveraging the cross-attention mechanism to develop a novel hallucination detector. This mechanism is responsible for selecting and combining the information contained in the source sequence that is relevant to retain during translation. Therefore, as hallucinations are translations whose content is detached from the source sequence, it is no surprise that connections between *anomalous* attention patterns and hallucinations have been drawn before in the literature (Berard et al., 2019; Raunak et al., 2021; Ferrando et al., 2022b). These patterns usually exhibit scattered source attention mass across the different tokens in the translation (e.g. most source attention mass is concentrated on a few irrelevant tokens such as punctuation and the end-of-sequence token). Inspired by such observations, previous work has designed *ad-hoc* heuristics to detect hallucinations that specifically target the anomalous maps. While such heuristics can be used to detect hallucinations to a satisfactory extent (Guerreiro et al., 2023a), we argue that a more theoretically-founded way of using anomalous attention information for hallucination detection is lacking in the literature.

Rather than aiming to find particular patterns, we go back to the main definition of hallucinations and draw the following hypothesis: as hallucinations—contrary to good translations—are not supported by the source content, they may exhibit cross-attention patterns that are statistically different from those found in good quality translations. Based on this hypothesis, we approach the problem of hallucination detection as a problem of anomaly detection with an **optimal transport (OT) formulation** (Kantorovich, 2006; Peyré, Cuturi, et al., 2019). Namely, we aim to find translations with source attention mass distributions that are highly distant from those of good translations. Intuitively, the more distant a translation's attention patterns are from those of good translations, the more **anomalous** it is in light of that distribution.

Our key contributions are:

- We propose an OT-inspired fully unsupervised hallucination detector that can be plugged into any attention-based NMT model;
- We find that the idea that attention maps for hallucinations are anomalous in light of a reference data distribution makes for an effective hallucination detector;
- We show that our detector not only outperforms all previous model-based detectors, but is also competitive with external detectors that employ auxiliary models that have been trained on millions of samples.

We will reference Chapter 3, and its resulting publication—(Guerreiro et al., 2023a)—interchangeably throughout this chapter.

4.1 Optimal Transport Problem and Wasserstein Distance

The first-order Wasserstein distance between two arbitrary probability distributions $\boldsymbol{\mu} \in \triangle_n$ and $\boldsymbol{\nu} \in \triangle_m$ is defined as

$$W(\boldsymbol{\mu}, \boldsymbol{\nu}) = \inf_{\boldsymbol{\gamma} \in \Pi(\boldsymbol{\mu}, \boldsymbol{\nu})} \mathbb{E}_{(u,v)\sim\boldsymbol{\gamma}} \left[c(u, v)\right], \tag{4.1}$$

where $c : [n] \times [m] \rightarrow \mathbb{R}_0^+$ is a cost function,[1] and $\Pi(\boldsymbol{\mu}, \boldsymbol{\nu}) = \{\boldsymbol{\gamma} \in \triangle^{n\times m} : \boldsymbol{\gamma}\mathbf{1} = \boldsymbol{\mu}; \boldsymbol{\gamma}^\top\mathbf{1} = \boldsymbol{\nu}\}$[2] is the set of all joint probability distributions whose marginals are $\boldsymbol{\mu}$, $\boldsymbol{\nu}$. The Wasserstein distance arises from the method of optimal transport (OT) (Kantorovich, 2006; Peyré, Cuturi, et al., 2019): OT measures distances between distributions in a way that depends on the geometry of the sample space. Intuitively, this distance indicates how much probability mass must be transferred from $\boldsymbol{\mu}$ to $\boldsymbol{\nu}$ in order to transform $\boldsymbol{\mu}$ into $\boldsymbol{\nu}$ while minimizing the transportation cost defined by c.

A notable example is the Wasserstein-1 distance, W_1, also known as Earth Mover's Distance (EMD), obtained for $c(u, v) = \|u - v\|_1$. The name follows from the simple intuition: if the distributions are interpreted as "two piles of mass" that can be moved around, the EMD represents the minimum amount of "work" required to transform one pile into the other, where the work is defined as the amount of mass moved multiplied by the distance it is moved.

Although OT has been explored for robustness Paty and Cuturi, 2019; Staerman et al., 2021 and out-of-distribution detection Wang et al., 2021; Yan et al., 2021; Cheng et al., 2022 in computer vision, the use of OT for anomaly detection in NLP applications remains largely overlooked.

1: We denote the set of indices $\{1, \ldots, n\}$ by $[n]$.

2: We extend the simplex notation for matrices representing joint distributions, $\triangle^{n\times m} = \{\boldsymbol{P} \in \mathbb{R}^{n\times m} : \boldsymbol{P} \geq \mathbf{0}, \mathbf{1}^\top\boldsymbol{P}\mathbf{1} = 1\}$.

4.2 On-the-fly Detection of Hallucinations

On-the-fly hallucination detectors are systems that can detect hallucinations without access to reference translations. These detectors are particularly relevant as they can be deployed in online applications where references are not readily available.[3]

3: As such, in this chapter, we will not consider metrics that depend on a reference sentence (e.g. chrF (Popović, 2016), COMET (Rei et al., 2020a)). For an analysis on the performance of such metrics, please refer to Guerreiro et al. (2023a).

4.2.1 Categorization of hallucination detectors

Previous work on on-the-fly detection of hallucinations in NMT has mostly focused on two categories of detectors: *external* detectors and *model-based* detectors. External detectors employ auxiliary models trained for related tasks such as quality estimation (QE) and cross-lingual embedding similarity. On the other hand, model-based detectors only require access to the model that generates the translations, and work by leveraging relevant internal features such as model confidence and cross-attention. These detectors are attractive due to their flexibility and low memory footprint, as they can very easily be plugged in on different models without the need for additional training data or computing infrastructure. Moreover, Guerreiro et al. (2023a) show

that model-based detectors can be predictive of hallucinations, outperforming QE models and even performing on par with state-of-the-art reference-based metrics.

4.2.2 Problem statement

We will focus specifically on model-based detectors that require obtaining internal features from a model $\mathcal{M}$. Building a hallucination detector generally consists of finding a scoring function $s_{\mathcal{M}} : \mathcal{X} \to \mathbb{R}$ and a threshold $\tau \in \mathbb{R}$ to build a binary rule $g_{\mathcal{M}} : \mathcal{X} \to \{0, 1\}$. For a given test sample $\boldsymbol{x} \in \mathcal{X}$,

$$g_{\mathcal{M}}(\boldsymbol{x}) = \mathbb{1}\{s_{\mathcal{M}}(\boldsymbol{x}) > \tau\}. \tag{4.2}$$

If $s_{\mathcal{M}}$ is an anomaly score, $g_{\mathcal{M}}(\boldsymbol{x}) = 0$ implies that the model $\mathcal{M}$ generates a 'normal' translation for the source sequence $\boldsymbol{x}$, and $g_{\mathcal{M}}(\boldsymbol{x}) = 1$ implies that $\mathcal{M}$ generates a 'hallucination' instead.[4]

4: From now on, we will omit the subscript $\mathcal{M}$ from all model-based scoring functions to ease notation effort.

4.3 Unsupervised Hallucination Detection with Optimal Transport

Anomalous cross-attention maps have been connected to the hallucinatory mode in several works (Lee et al., 2018b; Berard et al., 2019; Raunak et al., 2021). Our method builds on this idea and uses the Wasserstein distance to estimate the cost of transforming a translation source mass distribution into a reference distribution. Intuitively, the higher the cost of such transformation, the more distant—and hence the more anomalous— the attention of the translation is with respect to that of the reference translation.

4.3.1 **Wass-to-Unif**: A data independent scenario

In this scenario, we only rely on the generated translation and its source mass distribution to decide whether the translation is a hallucination or not. Concretely, for a given test sample $\boldsymbol{x} \in \mathcal{X}$:

1. We first obtain the source mass attention distribution $\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}) \in \triangle_{|\boldsymbol{x}|}$, such that $\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}) = \frac{1}{m}[(\boldsymbol{\Omega}(\boldsymbol{x})]^\top \mathbf{1}$;[5]
2. We then compute an anomaly score, $s_{\mathsf{wtu}}(\boldsymbol{x})$, by measuring the Wasserstein distance between $\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x})$ and a reference distribution $\boldsymbol{u}$:

$$s_{\mathsf{wtu}}(\boldsymbol{x}) = W(\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}), \boldsymbol{u}). \tag{4.3}$$

5: We follow the same notation as presented in the Background chapter (see Chapter 2): for a source sequence of arbitrary length n and a target sequence of arbitrary length m, we will designate as $\boldsymbol{\Omega} \in [0, 1]^{m \times n}$ the matrix of attention weights that is obtained by averaging across all the cross-attention heads of the last layer of the decoder module.

▸ **Choice of reference translation.** A natural choice for $\boldsymbol{u}$ is the uniform distribution, $\boldsymbol{u} = \frac{1}{n} \cdot \mathbf{1}$, where $\mathbf{1}$ is a vector of ones of size n. In the context of our problem, a uniform source mass distribution means that all source tokens are equally attended.

▸ **Choice of cost function.** We consider the 0/1 cost function, $c(i, j) = \mathbf{1}[i \neq j]$, as it guarantees that the cost of transporting a unit mass from any token i to any token $j \neq i$ is constant. For this distance function, the

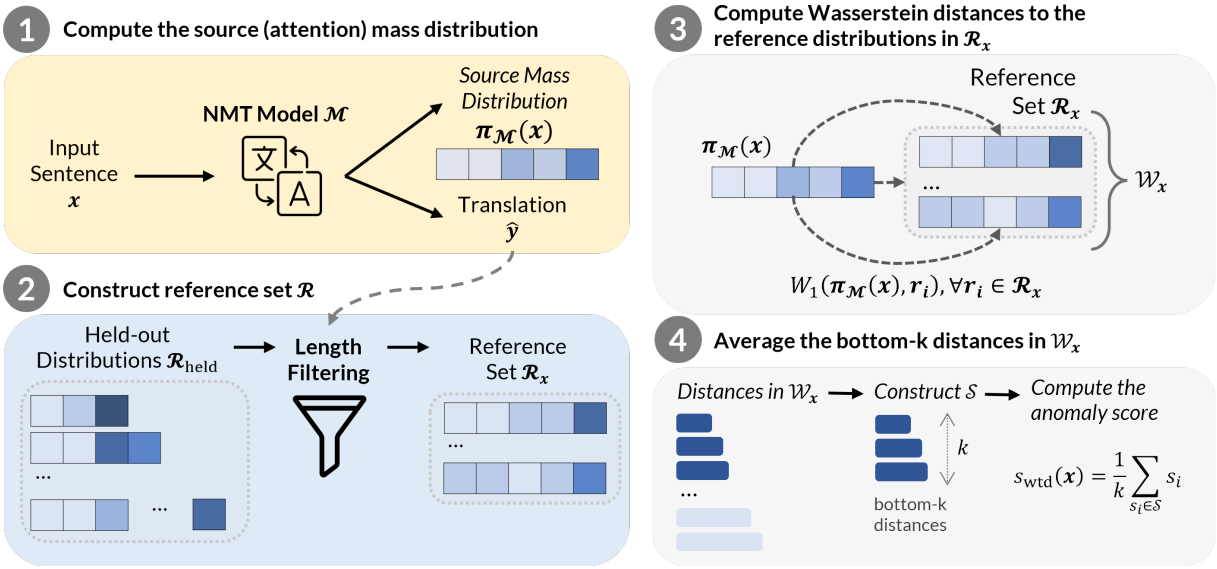


Figure 4.1: Procedure diagram for computation of the detection scores for the data-driven method Wass-to-Data.

problem in Equation 4.1 has the following closed-form solution (Villani, 2009):

$$W(\pi_{\mathcal{M}}(x), u) = \frac{1}{2}\|\pi_{\mathcal{M}}(x) - u\|_1. \tag{4.4}$$

This is a well-known result in optimal transport: the Wasserstein distance under the 0/1 cost function is equivalent to the total variation distance between the two distributions. On this metric space, the Wasserstein distance depends solely on the probability mass that is transported to transform $\pi_{\mathcal{M}}(x)$ to u. Importantly, *this formulation ignores the starting locations and destinations of that probability mass* as the cost of transporting a unit mass from any token i to any token $j \neq i$ is constant.

▸ **Interpretation of Wass-to-Unif.** Attention maps for which the source attention mass is highly concentrated on a very sparse set of tokens (regardless of their location in the source sentence) can be very predictive of hallucinations (Berard et al., 2019; Guerreiro et al., 2023a). Thus, the bigger the distance between the source mass distribution of a test sample and the uniform distribution, the more peaked the former is, and hence the closer it is to such predictive patterns.

4.3.2 Wass-to-Data: A data-driven scenario

In this scenario, instead of using a single reference distribution, we use a set of reference source mass distributions, $\mathcal{R}_x$, obtained with the same model. By doing so, we can evaluate how anomalous a given translation is compared to a model data-driven distribution, rather than relying on an arbitrary choice of reference distribution.

First, we use a held-out dataset $\mathcal{D}_{\text{held}}$ that contains samples for which the model $\mathcal{M}$ generates good quality translations according to an automatic evaluation metric (here, we use COMET (Rei et al., 2020a)). We use this dataset to construct (offline) a set of held-out source attention

distributions $\mathcal{R}_{\text{held}} = \{\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}) \in \triangle_{|\boldsymbol{x}|} : \boldsymbol{x} \in \mathcal{D}_{\text{held}}\}$. Then, for a given test sample $\boldsymbol{x} \in \mathcal{X}$, we apply the procedure illustrated in Figure 4.1:

1. We generate a translation $\hat{\boldsymbol{y}} = (y_1, \ldots, y_m)$ and obtain the source mass attention distribution $\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}) \in \triangle_{|\boldsymbol{x}|}$;
2. We apply a length filter to construct the sample reference set $\mathcal{R}_x$, by restricting $\mathcal{R}_x$ to contain source mass distributions of $\mathcal{R}_{\text{held}}$ correspondent to translations of size $[(1-\delta)m, (1+\delta)m]$ for a predefined $\delta \in]0, 1[$;[6]
3. We compute pairwise Wasserstein-1 distances between $\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x})$ and each element $\boldsymbol{r}_i$ of $\mathcal{R}_x$:

$$\mathcal{W}_{\boldsymbol{x}} = \left(W_1(\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}), \boldsymbol{r}_1), \ldots, W_1(\boldsymbol{\pi}_{\mathcal{M}}(\boldsymbol{x}), \boldsymbol{r}_{|\mathcal{R}_x|})\right) \tag{4.5}$$

4. We obtain the anomaly score $s_{\text{wtd}}(\boldsymbol{x})$ by averaging the bottom-k distances in $\mathcal{W}_{\boldsymbol{x}}$:

$$s_{\text{wtd}}(\boldsymbol{x}) = \frac{1}{k} \sum_{s_i \in \mathcal{S}} s_i, \tag{4.6}$$

   where $\mathcal{S}$ is the set containing the k smallest elements of $\mathcal{W}_{\boldsymbol{x}}$.

6: For efficiency reasons, we set the maximum cardinality of $\mathcal{R}_x$ to $|\mathcal{R}|_{\max}$. If length-filtering yields a set with more than $|\mathcal{R}|_{\max}$ examples, we randomly sample $|\mathcal{R}|_{\max}$ examples from that set to construct $\mathcal{R}_x$.

▸ **Interpretation of Wass-to-Data.** Hallucinations, unlike good translations, are not fully supported by the source content. Wass-to-Data evaluates how anomalous a translation is by comparing the source attention mass distribution of that translation to those of good translations. The higher the Wass-to-Data score, the more anomalous the source attention mass distribution of that translation is in comparison to those of good translations, and the more likely it is to be an hallucination.

▸ **Relation to Wass-to-Unif.** The Wasserstein-1 distance (see Section 4.1) between two distributions is equivalent to the ℓ_1-norm of the difference between their *cumulative distribution functions* (Peyré and Cuturi, 2018). Note that this is different from the result in Equation 4.4, as the Wasserstein distance under $c(i, j) = \mathbf{1}[i \neq j]$ as the cost function is proportional to the norm of the difference between their *probability mass functions*. Thus, Wass-to-Unif will be more sensitive to the overall structure of the distributions (e.g. sharp probability peaks around some points), whereas Wass-to-Data will be more sensitive to the specific values of the points in the two distributions.

4.3.3 Wass-Combo: The best of both worlds

With this scoring function, we aim at combining Wass-to-Unif and Wass-to-Data into a single detector. To do so, we propose using a two-stage process that exploits the computational benefits of Wass-to-Unif over Wass-to-Data. Put simply, (i) we start by assessing whether a test sample is deemed a hallucination according to Wass-to-Unif, and if not (ii) we compute the Wass-to-Data score. Formally,

$$s_{\text{wc}}(\boldsymbol{x}) = \mathbb{1}\left[s_{\text{wtu}}(\boldsymbol{x}) > \tau_{\text{wtu}}\right] \times \tilde{s}_{\text{wtu}}(\boldsymbol{x}) + \mathbb{1}\left[s_{\text{wtu}}(\boldsymbol{x}) \leq \tau_{\text{wtu}}\right] \times s_{\text{wtd}}(\boldsymbol{x}) \tag{4.7}$$

for a predefined scalar threshold τ_{wtu}. To set that threshold, we compute $\mathcal{W}_{\text{wtu}} = \{s_{\text{wtu}}(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{D}_{\text{held}}\}$ and set $\tau_{\text{wtu}} = P_K$, i.e τ_{wtu} is the K^{th} percentile of $\mathcal{W}_{\text{wtu}}$ with $K \in]98, 100[$ (in line with hallucinatory rates

reported in (Müller et al., 2020; Wang and Sennrich, 2020; Raunak et al., 2022a)).[7]

7: In order to make the scales of s_{wtu} and s_{wtd} compatible, we use a scaled $\tilde{s}_{\text{wtu}}$ value instead of s_{wtu} in Equation 4.7. We obtain $\tilde{s}_{\text{wtu}}$ by min-max scaling s_{wtu} such that $\tilde{s}_{\text{wtu}}$ is within the range of s_{wtd} values obtained for a held-out set.

4.4 Experimental Setup

4.4.1 Model and data

We follow the setup in Guerreiro et al. (2023a). In that work, the authors released a dataset of 3415 translations for WMT18 DE-EN news translation data (Bojar et al., 2018) with annotations on critical errors and hallucinations. Our analysis in the main text focuses on this dataset as it is the only available dataset that contains human annotations on hallucinations produced naturally by an NMT model.[8] Nevertheless, in order to access the broader validity of our methods for other low and mid-resource language pairs and models, we follow a similar setup to that of Tang et al. (2022) in which quality assessments are converted to hallucination annotations. For those experiments, we use the RO-EN (mid-resource) and NE-EN (low-resource) translations from the MLQE-PE dataset (Fomicheva et al., 2022). In Appendix A.6, we present full details on the setup and report the results of these experiments. Importantly, our empirical observations are similar to those of the main text. For all our experiments, we obtain all model-based information required to build the detectors using the same models that generated the translations in consideration.

8: Full details about the dataset and the NMT model can be found in the previous chapter (Chapter 3).

4.4.2 Baseline detectors

Model-based detectors

We compare our methods to the two best performing model-based methods in Guerreiro et al. (2023a).

- **ATTN-IGN-SRC.** As we have seen in Chapter 2, this method (*attention ignores source*) consists of computing the proportion of source words with a total incoming attention mass lower than a threshold λ:

$$s_{\text{ais}}(\boldsymbol{x}) = \frac{1}{n}\sum_{j=1}^{n} \mathbb{1}\left[(\boldsymbol{\Omega}^{\top}(\boldsymbol{x})\mathbf{1})_j < \lambda\right]. \tag{4.8}$$

This approach was initially proposed in Berard et al. (2019). We follow their work and use $\lambda = 0.2$.

- **SEQ-LOGPROB.** We compute the length-normalised sequence log-probability of the translation:

$$s_{\text{slp}}(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^{m} \log p_{\boldsymbol{\theta}}(y_k \mid \boldsymbol{y}_{<k}, \boldsymbol{x}). \tag{4.9}$$

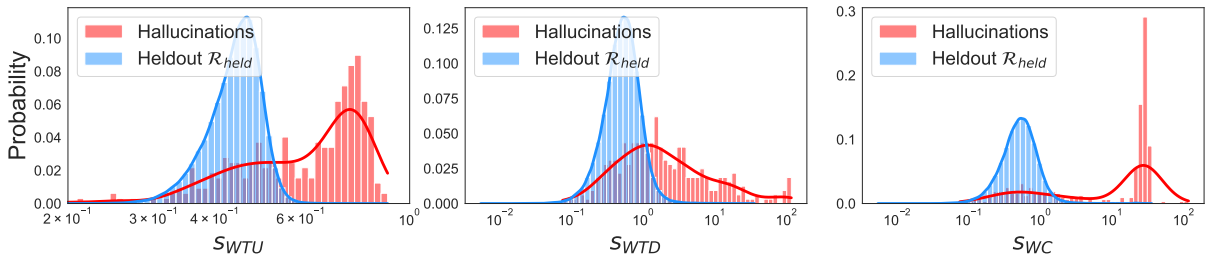


Figure 4.2: Histogram scores for our methods – Wass-to-Unif (left), Wass-to-Data (center) and Wass-Combo (right). We display Wass-to-Data and Wass-Combo scores on log-scale.

External detectors

We provide a comparison to detectors that exploit state-of-the-art models in related tasks, as it helps monitor the development of model-based detectors.

- **CometKiwi.** We compute sentence-level quality scores with CometKiwi (Rei et al., 2022b), the winning reference-free model of the WMT22 QE shared task (Zerva et al., 2022b). It has more than 565M parameters and it was trained on more than 1M human quality annotations. Importantly, this training data includes human annotations for several low-quality translations and hallucinations.

- **LaBSE.** We leverage LaBSE (Feng et al., 2020) to compute cross-lingual sentence representations for the source sequence and translation. We use the cosine similarity of these representations as the detection score. The model is based on the BERT (Devlin et al., 2019) architecture and was trained on more than 20 billion sentences. LaBSE makes for a good baseline, as it was optimized in a self-supervised way with a translate matching objective that is very much aligned with the task of hallucination detection: during training, LaBSE is given a source sequence and a set of translations including the true translation and multiple negative alternatives, and the model is optimized to specifically discriminate the true translation from the other negative alternatives by assigning a higher similarity score to the former.

4.4.3 Evaluation metrics

We report the Area Under the Receiver Operating Characteristic curve (AUROC) and the False Positive Rate at 90% True Positive Rate (FPR@90TPR) to evaluate the performance of different detectors.

4.4.4 Implementation details

We use WMT18 DE-EN data samples from the held-out set used in Guerreiro et al. (2023a), and construct $\mathcal{D}_{\text{held}}$ to contain the 250k samples with highest COMET score. To obtain Wass-to-Data scores, we set $\delta = 0.1$, $|\mathcal{R}|_{\max} = 1000$ and $k = 4$. To obtain Wass-to-Combo scores, we set $\tau_{\text{wtu}} = P_{99.9}$. We perform extensive ablations on the construction of $\mathcal{R}_{\text{held}}$ and on all other hyperparameters in Appendix A.3. We also report the computational runtime of our methods in Appendix A.1.

Table 4.1: Performance of all hallucination detectors. For Wass-to-Data and Wass-Combo we present the mean and standard deviation scores across five random seeds.

DETECTOR	AUROC ↑	FPR@90TPR ↓
External Detectors		
CometKiwi	86.96	53.61
LaBSE	**91.72**	**26.91**
Model-based Detectors		
Attn-ign-SRC	79.36	72.83
Seq-Logprob	83.40	59.02
OURS		
Wass-to-Unif	80.37	72.22
Wass-to-Data	84.20 ± 0.15	**48.15** ± 0.54
Wass-Combo	**87.17** ± 0.07	**47.56** ± 1.30

4.5 Results

4.5.1 Performance on on-the-fly detection

We start by analyzing the performance of our proposed detectors on a real world on-the-fly detection scenario. In this scenario, the detector must be able to flag hallucinations *regardless of their specific type* as those are unknown at the time of detection.

- **WASS-COMBO IS THE BEST MODEL-BASED DETECTOR.** Table 4.1 shows that Wass-Combo outperforms most other methods both in terms of AUROC and FPR. When compared to the previous best-performing model-based method (Seq-Logprob), Wass-Combo obtains boosts of approximately 4 and 10 points in AUROC and FPR, respectively. These performance boosts are further evidence that model-based features can be leveraged, in an unsupervised manner, to build effective detectors. Nevertheless, the high values of FPR suggest that there is still a significant performance margin to reduce in future research.

- **THE NOTION OF DATA PROXIMITY IS HELPFUL TO DETECT HALLUCINATIONS.** Table 4.1 shows that Wass-to-Data outperforms the previous best-performing model-based method (Seq-Logprob) in both AUROC and FPR (by more than 10%). This supports the idea that cross-attention patterns for hallucinations are anomalous with respect to those of good model-generated translations, and that our method can effectively measure this level of anomalousness. On the other hand, compared to Wass-to-Uni, Wass-to-Data shows a significant improvement of 30 FPR points. This highlights the effectiveness of leveraging the data-driven distribution of good translations instead of the ad-hoc uniform distribution. Nevertheless, Table 4.1 and Figure 4.2 show that combining both methods brings further performance improvements. This suggests that these methods may specialize in different types of hallucinations, and that combining them allows for detecting a broader range of anomalies. We will analyze this further in Section 4.5.2.

- **OUR MODEL-BASED METHOD ACHIEVES COMPARABLE PERFORMANCE TO EXTERNAL MODELS.** Table 4.1 shows that Wass-Combo outperforms CometKiwi, with significant improvements on FPR. However, there still exists a gap to LaBSE, the best overall detector. This performance gap indicates that more powerful detectors can be built, paving the way for future work in model-based hallucination detection. Nevertheless, while

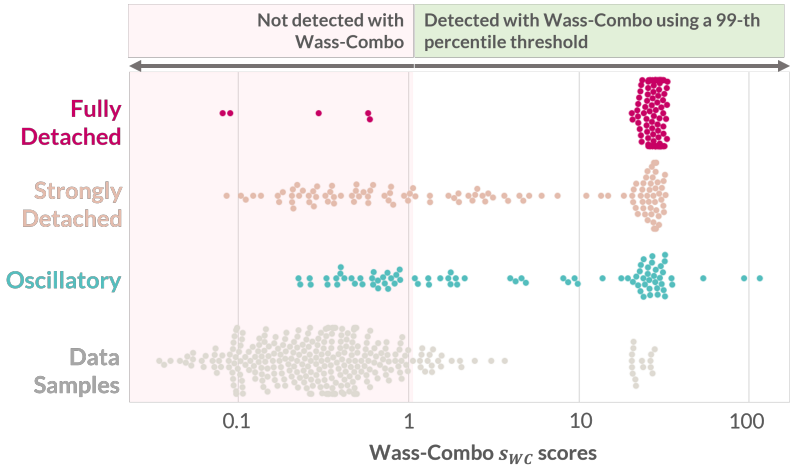


Figure 4.3: Distribution of Wass-Combo scores (on log-scale) for hallucinations and data samples, and performance on a fixed-threshold scenario.

relying on external models seems appealing, deploying and serving them in practice usually comes with additional infrastructure costs, while our detector relies on information that can be obtained when generating the translation.

▸ **Translation quality assessments are less predictive than similarity of cross-lingual sentence representations.** Table 4.1 shows that LaBSE outperforms the state-of-the-art quality estimation system CometKiwi, with vast improvements in terms of FPR. This shows that for hallucination detection, quality assessments obtained with a QE model are less predictive than the similarity between cross-lingual sentence representations. This may be explained through their training objectives (see Section 4.4.2): while CometKiwi employs a more general regression objective in which the model is trained to match human quality assessments, LaBSE is trained with a translate matching training objective that is very closely related to the task of hallucination detection.

4.5.2 Do detectors specialize in different types of hallucinations?

In this section, we present an analysis on the performance of different detectors for different types of hallucinations. We report both a quantitative analysis to understand whether a detector can distinguish a specific hallucination type from other translations (Table 4.2), and a qualitative analysis on a fixed-threshold scenario[9] (Figure 4.3). This analysis is particularly relevant to better understand how different detectors specialize in different types of hallucinations. In Appendix A.6, we show that the trends presented in this section hold for other mid- and low-resource language pairs.

9: We set the threshold by finding the 99th percentile of Wass-Combo scores obtained for 100k samples from the clean WMT18 DE-EN held-out set (see Section 4.4.4).

▸ **Fully detached hallucinations.** Detecting fully detached hallucinations is remarkably easy for most detectors. Interestingly, Wass-to-Unif significantly outperforms Wass-to-Data on this type of hallucination. This highlights how combining both methods can be helpful. In fact, Wass-Combo performs similarly to Wass-to-Unif, and can very easily separate most fully detached hallucinations from other translations on a fixed-threshold scenario (Figure 4.3). Note that the performance

Table 4.2: Performance of all hallucination detectors for each hallucination type. For Wass-to-Data and Wass-Combo, we present the mean and standard deviation across five random seeds.

Detector	Fully Detached	Oscillatory	Strongly Detached
External Detectors			
CometKiwi	87.75	**93.04**	81.78
LaBSE	**98.91**	84.62	**89.72**
Model-based Detectors			
Attn-ign-SRC	95.76	59.53	77.42
Seq-Logprob	95.64	71.10	**80.15**
Ours			
Wass-to-Unif	96.35	69.75	72.19
Wass-to-Data	88.24 ± 0.29	**87.80** ± 0.10	77.60 ± 0.18
Wass-Combo	**96.57** ± 0.10	85.74 ± 0.10	78.89 ± 0.15

(a) AUROC – the higher the better.

Detector	Fully Detached	Oscillatory	Strongly Detached
External Detectors			
CometKiwi	33.70	**23.80**	42.98
LaBSE	**0.52**	50.26	**28.88**
Model-based Detectors			
Attn-ign-SRC	8.51	81.24	76.68
Seq-Logprob	4.62	72.99	**65.39**
Ours			
Wass-to-Unif	**3.27**	78.78	88.32
Wass-to-Data	36.60 ± 1.92	**40.04** ± 1.57	**63.96** ± 2.04
Wass-Combo	3.56 ± 0.00	**41.38** ± 1.59	**64.55** ± 1.93

(b) FPR@90TPR (%) – the lower the better.

of Wass-to-Unif for fully detached hallucinations closely mirrors that of Attn-ign-SRC. This is not surprising, since both methods, at their core, try to capture similar patterns: translations for which the source attention mass distribution is highly concentrated on a small set of source tokens.

- **Strongly detached hallucinations.** These are the hardest hallucinations to detect with our methods. Nevertheless, Wass-Combo performs competitively with the previous best-performing model-based method for this type of hallucinations (Seq-Logprob). We hypothesize that the difficulty in detecting these hallucinations may be due to the varying level of detachment from the source sequence. Indeed, Figure 4.3 shows that Wass-Combo scores span from a cluster of strongly detached hallucinations with scores similar to other data samples to those similar to the scores of most fully detached hallucinations.

- **Oscillatory hallucinations.** Wass-to-Data and Wass-Combo significantly outperform all previous model-based detectors on detecting oscillatory hallucinations. This is relevance in the context of model-based detectors, as previous detectors notably struggle with detecting these hallucinations. Moreover, Wass-Combo also manages to outperform LaBSE with significant improvements in FPR. This hints that the repetition of words or phrases may not be enough to create sentence-level representations that are highly dissimilar from the non-oscillatory source sequence. In contrast, we find that CometKiwi appropriately penalizes oscillatory hallucinations, which aligns with observations made in Guerreiro et al. (2023a).

  Additionally, Figure 4.3 shows that the scores for oscillatory hallucinations are scattered along a broad range. After close evaluation, we observed that this is highly related to the severity of the oscillation: almost all non-detected hallucinations are not severe oscillations (see Appendix A.5).

4.6 Conclusions

We propose a novel plug-in model-based detector for hallucinations in NMT. Unlike previous attempts to build an attention-based detec-

tor, we do not rely on *ad-hoc* heuristics to detect hallucinations, and instead pose hallucination detection as an optimal transport problem: our detector aims to find translations whose source attention mass distribution is highly distant from those of good quality translations. Our empirical analysis shows that our detector outperforms all previous model-based detectors. Importantly, in contrast to these prior approaches, it is suitable for identifying oscillatory hallucinations, thus addressing an important gap in the field. We also show that our detector is competitive with external detectors that use state-of-the-art quality estimation or cross-lingual similarity models. Notably, this performance is achieved without the need for large models, or any data with quality annotations or parallel training data. Finally, thanks to its flexibility, our detector can be easily deployed in real-world scenarios, making it a valuable tool for practical applications.

Our work has informed several subsequent works studying hallucinations in translation models. Dale et al. (2023) examined our proposed detectors in the context of massively multilingual translation models, revealing that attention proves less effective at predicting anomalous patterns in such scenarios. Himmi et al. (2024) advanced this line of research by proposing a simple yet effective ensembling strategy that combines various hallucination detectors, including those we proposed. Building directly on our work, OTTAWA (Huang et al., 2024) introduced a novel Optimal Transport (OT)-based word aligner specifically designed to enhance the detection of hallucinations and omissions in MT systems. This approach demonstrated improved performance over our methods across the board, even in multilingual scenarios.

Throughout the last two chapters, we have focused on developing and analysing methods for detecting and mitigating hallucinations in bilingual neural machine translation models. We began by establishing a solid foundation for studying hallucinations, introducing a new human-labelled dataset, identifying effective detection methods, and proposing a lightweight mitigation approach. Then, in this chapter, we advanced to proposing a novel model-based detector that leverages the cross-attention mechanism using an optimal transport formulation. Now, we will turn to investigating hallucinations in massively multilingual encoder-decoder translation models and decoder-only large language models, broadening our scope to encompass a wider range of translation scenarios, including low-resource language pairs, non-English-centric directions, and domain-specific translations. This will allow us to examine how the insights gained from our previous work apply to more complex and diverse translation scenarios, and to uncover new challenges in addressing hallucinations across different model architectures and data conditions.

Hallucinations in Large Multilingual Translation Models

5

Studies such as the ones presented in previous chapters have contributed towards better understanding, detection and mitigation of hallucinations in machine translation. However, this research has been exclusively conducted on *small bilingual* models (<100M parameters) trained on a *single English-centric high-resource* language pair (Raunak et al., 2021; Dale et al., 2022; Ferrando et al., 2022b; Guerreiro et al., 2023a,b; Xu et al., 2023a). This leaves a knowledge gap regarding the prevalence and properties of hallucinations in large-scale translation models across different translation directions, domains and data conditions.

In this chapter, we aim to fill this gap by investigating hallucinations on two different classes of models. The first and main class in our analysis is the *de facto* standard approach of encoder-decoder massively multilingual supervised models: we use the M2M-100 family of multilingual NMT models (Fan et al., 2020), which includes the largest open-source multilingual NMT model with 12B parameters. The second class is the novel and promising approach of leveraging generative decoder-only LLMs for translation. Contrary to conventional NMT models, these models are trained on massive amounts of monolingual data in many languages, with a strong bias towards English, and do not require parallel data. In our analysis, we use ChatGPT and GPT-4, as they have been shown to achieve high translation quality over a wide range of language pairs (Hendy et al., 2023a; Peng et al., 2023).

We organize our study by analyzing the two prevalent types of hallucinations in NMT considered in the literature: hallucinations under perturbation and natural hallucinations (Lee et al., 2018a; Raunak et al., 2021; Guerreiro et al., 2023a). Firstly, we study hallucinations under perturbation and evaluate whether these translation systems are robust to source-side artificial perturbations. While previous studies have found that these perturbations (e.g., spelling errors and capitalization mistakes) can reliably induce hallucinations (Lee et al., 2018a; Raunak et al., 2021), it is not clear whether those conclusions hold for large multilingual models. Secondly, we comprehensively investigate natural hallucinations, and evaluate their properties in the outputs of the massively multilingual M2M models on a vast range of conditions, spanning from English-centric to non-English-centric language pairs, translation directions with little supervision, and specialized medical domain data where hallucinations can have devastating impact. Finally, we study a hybrid setup where other models can be requested as fallback systems when an original system hallucinates, with the aim of mitigating hallucinations and improving translation quality on-the-fly.

We provide several key insights on properties of hallucinations, including:

- models predominantly struggle with hallucinations in low-resource language pairs and translating out of English;

- toxic patterns found in hallucinations can be traced back to the training data;
- smaller distilled models can, surprisingly, hallucinate less than large-scale models; we hypothesize that this is due to modeling choices that discourage hallucinations (e.g., leveraging less potent shallow decoders that rely more on the encoder representations);
- LLMs produce hallucinations that are qualitatively different from those of conventional translation models, mostly consisting of off-target translations, overgeneration, and even failed attempts to translate;
- hallucinations are *sticky* and hard to reverse with models that share the same training data, whereas employing more diverse fallback systems can substantially improve overall translation quality and eliminate pathologies such as oscillatory hallucinations.

5.1 Experimental Suite

5.1.1 Models

For supervised multilingual NMT models, we use the transformer-based (Vaswani et al., 2017) M2M-100 family of models (Fan et al., 2020): `M2M (S)` with 418M parameters, `M2M (M)` with 1.2B parameters, and `M2M (L)` with 12B parameters. These models were trained on a many-to-many parallel dataset of 7.5B sentences, and support 100 languages and thousands of language pairs (LP). We also evaluate `SMaLL100` (Mohammadshahi et al., 2022), a model with 330M parameters obtained via distillation of `M2M (L)`. `SMaLL100` was trained on a smaller training set obtained via uniform sampling across all language pairs to reduce the bias towards high-resource languages: only 100k parallel sentences from the M2M training data were used for each LP, for a total of 456M parallel sentences. For decoding, we run beam search with a beam size of 4.

As for the alternative strategy using LLMs, we use two GPT models: `ChatGPT` (`gpt-3.5-turbo`) and `GPT-4`.[1] These models are upgraded versions of GPT-3.5 — a 175B parameter GPT (Radford and Narasimhan, 2018; Radford et al., 2019; Brown et al., 2020b) large language model — that has been fine-tuned with human feedback in the style of InstructGPT (Ouyang et al., 2022). In particular, `ChatGPT` has been shown to achieve impressive results for multiple multilingual tasks, including translation (Fu et al., 2023; Hendy et al., 2023a; Kocmi and Federmann, 2023c). To generate translations, we use the zero-shot prompt template used in Hendy et al. (2023a) and the default API parameters.[2]

1: https://platform.openai.com/docs/models/; we used the API in March and April, 2023.

2: We encountered several API/server errors when prompting GPT models for translation, particularly for low-resource language pairs and languages with lower coverage scripts.

5.1.2 Datasets

We chose datasets familiar to researchers and practitioners, ensuring no train/test overlap for the M2M models.[3] To this end, we selected premier translation benchmarks: FLORES-101 (Goyal et al., 2022),

3: The training data of GPT models is not publicly available; thus, we cannot guarantee that they have not been exposed to the data we use in our analysis.

TICO (Anastasopoulos et al., 2020), and WMT. FLORES-101 is a multi-parallel dataset that consists of Wikipedia text in 101 languages, allowing evaluation across numerous translation directions; we join the `dev` and `devtest` sets. TICO is a specialized medical-domain multilingual benchmark with COVID-19 related data, such as medical papers and news articles; we join the `dev` and `test` sets. Additionally, we use WMT benchmarks from the M2M paper evaluation suite, as they were removed from the training data.[4] Further details about the datasets can be found in Appendix B.2.

4: WMT data serves as our validation set for tuning thresholds as needed.

5.1.3 Evaluation metrics

Our main lexical metric is spBLEU (Goyal et al., 2022), as it has been widely employed in works on multilingual translation (Fan et al., 2020; Wenzek et al., 2021; NLLB Team et al., 2022) and offers fairer evaluation for low-resource languages compared to BLEU (Papineni et al., 2002; Post, 2018).[5] Moreover, we also adopt the latest neural reference-based and reference-free COMET variants: COMET-22 and CometKiwi (Rei et al., 2022a,b). Lastly, we use the cross-lingual encoder LaBSE (Feng et al., 2022) to obtain sentence similarity scores, as these have been successfully used in prior research on detection of hallucinations (Dale et al., 2022; Guerreiro et al., 2023b).

5: Signature: `nrefs:1|case:mixed|eff:yes|tok:flores101|smooth:exp|version:2.3.1`.

5.2 Hallucinations under Perturbation

We start our analysis by focusing on artificial hallucinations. We first provide an in-depth overview of our evaluation setting, focusing on the construction of the perturbed data and detection approach. Then, we present our results and analyze the properties of these hallucinations.

5.2.1 Evaluation setting

- **PERTURBATIONS.** We apply the same minimal perturbations used in Xu et al. (2023a) to construct perturbed source sequences: misspeling errors, insertion of frequent tokens in the beginning of the sequence, and capitalization errors. We randomly misspell words by changing the characters with a probability of 0.01;[6] insert a token chosen randomly from the 50 most frequent tokens or punctuation in the test set; and, randomly title-case words with a probability of 0.1, guaranteeing that at least one word's capitalization is perturbed.

- **TRANSLATION DIRECTIONS.** We use the FLORES dataset for these experiments, and focus specifically on translation out of English. We select all M2M bridge languages, as well as additional low-resource languages that were underrepresented among bridge languages.[7] Overall, we generate translations for 31 different language pairs.[8]

- **DETECTION.** Our detection approach is inspired by previous works on hallucinations under perturbation (Lee et al., 2018a; Raunak et al., 2021; Ferrando et al., 2022b; Xu et al., 2023a). We presented the algorithm back in Algorithm 1 in Chapter 2.

6: For misspelling words, we use the `ButterFingersPerturbation` from the NL-Augmenter framework (Dhole et al., 2022).

7: Fan et al. (2020) divide the 100 languages into 14 related groups (e.g. Romance, Slavic) and mine languages within a group against each other. Then they define 1-3 "bridge" languages per group (often those with most resources) and mine them against each other to connect the groups. We present these languages and all others used in this work in Appendix B.1.

8: We pair the following languages with English: `af`, `ar`, `ast`, `bn`, `cs`, `cy`, `de`, `el`, `es`, `fa`, `fi`, `fr`, `he`, `hi`, `hr`, `hu`, `id`, `ja`, `ko`, `lt`, `nl`, `oc`, `pl`, `pt`, `ru`, `sv`, `sw`, `tl`, `tr`, `vi`, `zh`. We selected all bridge languages (see Table B.1) and additional low-resource languages, as they are underrepresented among the bridge languages. Although Tamil is a bridge language, we did not include it in our analysis, as we could not find enough candidates with reasonable quality for hallucinations under perturbation after applying rule (i) of the detection approach that is described below.

Table 5.1: Fraction of languages for which models produces at least one hallucination under perturbation, and average hallucination rate (and median, in subscript) across all languages at each resource level.

Model	Low Resource		Mid Resource		High Resource	
	LP Fraction	Rate (%)	LP Fraction	Rate (%)	LP Fraction	Rate (%)
SMaLL100	2/7	$0.225_{0.00}$	4/19	$0.017_{0.00}$	1/5	$0.017_{0.00}$
M2M (S)	6/7	$0.332_{0.17}$	13/19	$0.166_{0.08}$	0/5	$0.000_{0.00}$
M2M (M)	4/7	$0.107_{0.08}$	7/19	$0.039_{0.00}$	0/5	$0.000_{0.00}$
M2M (L)	4/7	$0.308_{0.08}$	4/19	$0.022_{0.00}$	0/5	$0.000_{0.00}$
ChatGPT	4/7	$0.059_{0.08}$	12/19	$0.183_{0.08}$	0/5	$0.000_{0.00}$
GPT-4	0/7	$0.000_{0.00}$	3/19	$0.035_{0.00}$	0/5	$0.000_{0.00}$

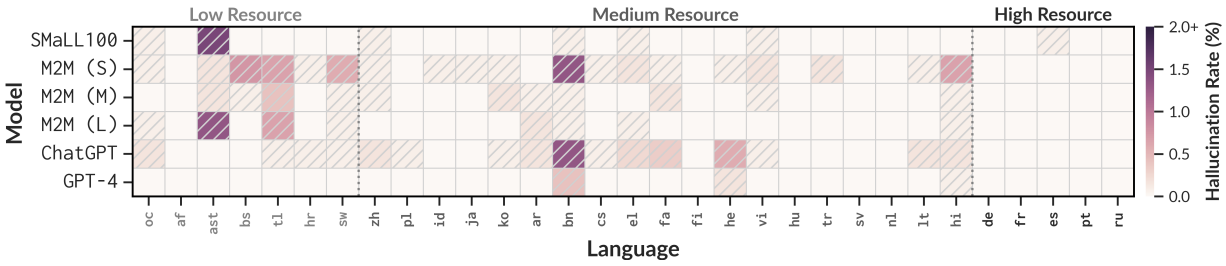


Figure 5.1: Heatmap of hallucination rates for each model in the languages considered. Pattern-filled cells indicate at least one hallucination under perturbation for a given model-language pair.

We adapt the algorithm to ensure consistency across multiple models and LPs. We first obtain source sentences for which all models produce translations that meet a minimum quality threshold (spBLEU > 9), sort them by average quality across models, and select the top 20% as candidates. Finally, we apply rule (ii) and set the threshold to spBLEU < 3. We selected these thresholds based on those used in previous works (Raunak et al., 2021; Ferrando et al., 2022b; Xu et al., 2023a).[9]

9: While hallucinations under perturbation do not explicitly target detachment from the source, we found that over 85% of these hallucinations would be detected with our detection approach for natural hallucinations (Section 5.3.1).

5.2.2 Results

Overall, we find that perturbations have the potential to trigger hallucinations, even in larger models. In what follows, we present several key insights.

- **Average hallucination rates for NMT models generally decrease with increasing resource levels.** Table 5.1 shows that all NMT models exhibit lower hallucination rates as resource levels increase. This is expected and suggests that these models are better equipped to handle source-side perturbations for language pairs with more parallel data during training. In fact, hallucinations under perturbation for high-resource languages are almost non-existent. However, Figure 5.1 reveals variability across languages, and even within the models in the same family that share the same training data. For instance, when translating to Asturian (`ast`), M2M (L) and the distilled SMaLL100 show significantly higher hallucination rates than M2M (S). This suggests that hallucinations emerge in non-trivial ways unrelated to the training data.

- **SMaLL100 exhibits lower hallucination rates than its teacher model M2M (L).** Recall that SMaLL100 was trained using uniform

sampling across all language pairs to reduce bias towards high-resource languages. The results in Table 5.1 may reflect a positive outcome from this approach: despite being much smaller than M2M (L), SMaLL100 hallucinates less and for fewer directions on low- and mid-resource language pairs.

▸ **Hallucinations under perturbation are not correlated with the quality of original translations.** The approach for detection of hallucinations under perturbation raises an interesting question: *are the original source sentences for which models produce higher quality translations less likely to lead to hallucinations when perturbed?* Our analysis found a weak Pearson correlation[10] between hallucinations under perturbation and spBLEU scores for the original unperturbed sources across all models. This indicates that even minimal perturbations in the source can cause models to undergo significant shifts in translation quality.

10: The point-biserial Pearson correlation between the hallucination assignments and the original sentence spBLEU scores is in the range $[-0.05, -0.02]$ for all models.

▸ **LLMs exhibit different hallucination patterns from conventional NMT models.** Contrary to NMT models, the GPT models generate more hallucinations for mid-resource languages than for low-resource languages (Table 5.1). Notably, GPT-4 produces no hallucinations in both low- and high-resource directions, and hallucinates for fewer languages compared to all other models. Additionally, hallucinations from LLMs are qualitatively different from those of other models: they often consist of off-target translations,[11] overgeneration, or even failed attempts to translate (e.g., *"This is an English sentence, so there is no way to translate it to Vietnamese"*; we provide further examples in Appendix B.3). Moreover, unlike NMT models that frequently produce oscillatory character hallucinations, GPT models do not generate any such hallucinations under perturbation. This further demonstrates that translation errors, even critical ones, obtained with LLMs are different from those produced by NMT models (Vilar et al., 2022; Hendy et al., 2023a).

11: We perform automatic language identification using the `fasttext` (Joulin et al., 2016) LID model `lid.176.bin`.

Interestingly, we also found that almost all hallucinations can be reversed with additional sampling from the model. This aligns with findings in Guerreiro et al. (2023a) and Manakul et al. (2023): as with traditional NMT models, LLM hallucinations may not necessarily imply model defect or inability to generate adequate translations, but could just stem from "bad luck" during generation.

5.3 Natural Hallucinations

We now turn to investigating natural hallucinations.[12] We first provide an overview of our evaluation setting, focusing on the scenarios and detection approach. Then, we present a thorough analysis exploring various properties of hallucinations.

12: From now on, we will use the terms natural hallucinations – both detached and oscillatory hallucinations – and hallucinations interchangeably.

5.3.1 Evaluation setting

▸ **Evaluation scenarios.** Analyzing multilingual translation models opens up several research scenarios that have not been explored in previous studies conducted on bilingual models. We will investigate

Table 5.2: Fraction of LPs on the English-centric setup for which models produce at least one hallucination, and average hallucination rate (and median, in subscript) across all LPs at each resource level.

Model	Low Resource		Mid Resource		High Resource	
	LP Fraction	Rate (%)	LP Fraction	Rate (%)	LP Fraction	Rate (%)
SMaLL100	14/16	$2.352_{0.57}$	19/38	$0.055_{0.02}$	1/10	$0.005_{0.00}$
M2M (S)	15/16	$15.20_{2.86}$	22/38	$0.254_{0.05}$	3/10	$0.025_{0.00}$
M2M (M)	14/16	$12.53_{1.42}$	17/38	$0.110_{0.00}$	2/10	$0.010_{0.00}$
M2M (L)	14/16	$11.22_{2.19}$	11/38	$0.034_{0.00}$	0/10	$0.000_{0.00}$

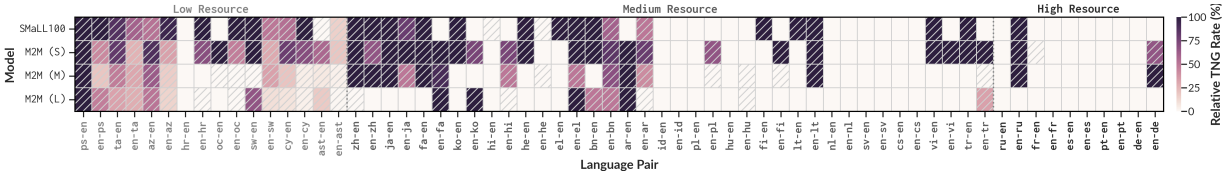


Figure 5.2: Heatmap of the rate of hallucinations detected with TNG (oscillatory hallucinations) among all hallucinations. Patterned cells indicate at least one natural hallucination for a given model-LP pair.

three different multilingual scenarios, comprising over 100 translation directions.

We begin with an English-centric scenario, pairing 32 languages[13] with English for a total of 64 translation directions. Then, we study a non-English-centric scenario inspired by Fan et al. (2020), exploring 25 language pairs corresponding to real-world use cases of translation not involving English.[14] Finally, we examine hallucinations in medical data, where they can have severely compromising effects. We pair 9 languages[15] with English for a total of 18 directions. We report results for the first two setups using the Flores dataset. For the final setup, we use the TICO dataset.

▸ **Detection.** We integrate key findings from research on detection of hallucinations and employ two main detectors: ALTI+ (Ferrando et al., 2022b) for detached hallucinations, and top n-gram (TNG) (Raunak et al., 2021, 2022a; Guerreiro et al., 2023a) for oscillatory hallucinations.

ALTI+ assesses the relative contributions of source and target prefixes to model predictions. As hallucinations are translations detached from the source sequence, ALTI+ can be leveraged to detect them by identifying sentences with minimal source contribution. Notably, it faithfully reflects model behavior and explicitly signals model detachment from the source in any translation direction (Ferrando et al., 2022b). This method has been successfully employed to detect hallucinated toxicity in a multilingual context in NLLB Team et al. (2022), and validated on human-annotated hallucinations in Dale et al. (2022), where it was shown that ALTI+ scores easily separate detached hallucinations from other translations.[16]

TNG, on the other hand, is a simple, lightweight black-box heuristic targeting oscillatory hallucinations. It compares the count of the top repeated translation n-gram to the count of the top repeated source n-gram, ensuring a minimum difference of t. This method has been validated on human-annotated hallucinations and found to identify oscillatory hallucinations with perfect precision (Guerreiro et al., 2023a).

13: `ar`, `ast`, `az`, `bn`, `cs`, `cy`, `de`, `el`, `es`, `fa`, `fi`, `fr`, `he`, `hi`, `hr`, `hu`, `id`, `ja`, `ko`, `lt`, `nl`, `oc`, `pl`, `ps`, `pt`, `ru`, `sv`, `sw`, `ta`, `tr`, `vi`, `zh`; similarly to the choice for the experiments on hallucinations under perturbation, we selected all the bridge languages and additional low-resource languages.

14: `hi-bn`, `it-fr`, `de-hu`, `it-de`, `cs-sk`, `nl-fr`, `fr-sw`, `ro-ru`, `ro-uk`, `de-hr`, `hr-sr`, `be-ru`, `hr-hu`, `hr-cs`, `el-tr`, `hr-sk`, `nl-de`, `af-zu`, `ro-hu`, `hi-mr`, `ro-tr`, `uk-ru`, `ro-hy`, `ar-fr`, `ro-de`; we selected language pairs that were used in the analysis from the original M2M paper on real-world settings for many-to-many translation (Fan et al., 2020). They represent translation directions that are commonly used in countries that have official and regional languages that do not include English.

15: `ar`, `fr`, `hi`, `id`, `ps`, `pt`, `ru`, `sw`, `zh`; we selected languages from the TICO benchmark that are supported by the M2M models.

Concurrently to the work in Chapter 4, Dale et al. (2022) investigated the use of ALTI+ for detection of natural hallucinations. ALTI+ shows superior performance in detecting detached hallucinations. Our proposed method in Chapter 4, while considerably stronger for oscillatory hallucinations, was impractical in this massively multilingual scenario, as it would possibly require creating over 100 datastores (one per language pair). Hence, we adopted TNG for oscillatory detection, given its proven precision (see Chapter 3). We provide the comparison between our OT approach and ALTI+ for the dataset released in Chapter 3 in Appendix A.4

Following previous work, we use $n = 4$ and $t = 2$ (Raunak et al., 2021; Guerreiro et al., 2023a) and exclude translations that meet the minimum quality threshold from Section 5.2.1.[17]

▸ **Remark on Model Selection.** We use ALTI+, a model-based detector, for reliable detection of detached hallucinations. Since we lack access to internal features from the GPT models, we exclude it from our model selection to ensure consistency in our analysis. Importantly, using alternative detectors could lead to misleading results and create discrepancies between the evaluation of LLMs and other models. Nonetheless, we will further examine LLMs in Section 5.4, exploring various aspects such as the generation of oscillatory hallucinations and performance as fallback systems.

5.3.2 English-centric translation

We start by studing hallucinations on English-centric language pairs. We reveal key insights on how properties of hallucinations change across resource levels, models and translation directions.

▸ **Hallucinations in low-resource language pairs are not only more frequent, but also distinct.** Table 5.2 shows that hallucinations occur frequently for low-resource directions, with all M2M models exhibiting average hallucination rates over 10%. Moreover, all models generate hallucinations for almost all low-resource language pairs. Regarding the type of hallucinations, Figure 5.2 shows that in low-resource directions, in contrast to mid- and high-resource ones, oscillatory hallucinations are less prevalent, while detached hallucinations occur more frequently. These findings suggest that, although massive multilingual models have significantly improved translation quality for low-resource languages, there is considerable room for improvement, and also highlight potential safety issues arising from translations in these directions.

▸ **SMaLL100 consistently relies more on the source than other models.** Despite being the smallest model, SMaLL100 shows remarkable hallucination rates across low- and mid-resource directions, hallucinating significantly less than its larger counterparts in low-resource ones. We hypothesize that these improved rates may be attributed not only to the uniform sampling of language pairs during training, but also to its architecture. While SMaLL100 shares a 12-layer encoder with the other models to obtain source representations, it diverges by employing a shallow 3-layer decoder – instead of a 12-layer decoder – and placing the target language code on the encoder side. This design may encourage greater reliance on the more complex encoder representations, reducing detachment from the source. In fact, distinct patterns in ALTI+ scores support this hypothesis: SMaLL100 has higher scores and similar patterns across all resource levels (see Figure 5.3). In contrast, the M2M models tend to rely less on the source, especially in low-resource `en-xx` LPs. Importantly, however, SMaLL100's lower hallucination rates do not necessarily imply superior translation quality compared to the M2M models: we found a strong correlation between M2M models' COMET-22 scores and their respective hallucination rates for low-resource LPs, whereas, contrastingly, the correlation is weak

16: We followed the recommendations in Guerreiro et al. (2023a) and set model-based ALTI+ thresholds based on validation data where the models are expected to perform well. Specifically, we obtained the lowest 0.02% — in line with natural hallucination rates reported in the literature (Raunak et al., 2022a) — of the ALTI+ score distributions for high-resource WMT benchmarks. Additionally, to ensure further trustworthy, high-precision measurements, we excluded detected candidates with LaBSE or CometKiwi scores — as these have been also been validated for detection of human-annotated detached hallucinations (Dale et al., 2022; Guerreiro et al., 2023b) — exceeding the top 10% of scores on translations from the same WMT benchmarks.

17: Note that oscillatory hallucinations can be simultaneously detected with ALTI+ and TNG.

for SMaLL100 (Table 5.3). This suggests that, despite relying more on the source, SMaLL100's translations may not necessarily be of higher quality than those of the M2M models.

▸ **Scaling up models within the same family reduces hallucination rates.** Table 5.2 shows that increasing the size of the M2M models results in consistent reductions in hallucination rates. Relative improvements are more pronounced for mid- and high-resource language pairs, with M2M (L) exhibiting fewer hallucinations and hallucinating for fewer languages than all other models.

▸ **Hallucinations are more frequent when translating out of English.** Table 5.4 demonstrates that models are significantly more prone to hallucinate when translating out of English. In fact, we found that models tend to detach more from the source text when translating out of English. This is evidenced by ALTI+ source contributions (see Figure 5.3) being lower across all language pairs in this direction compared to translating into English, which aligns with observations in Ferrando et al. (2022b). Interestingly, we discovered that the translation direction can also influence the properties of hallucinations: (i) over 90% of off-target hallucinations occur when translating out of English, and (ii) nearly all hallucinations into English for mid- and high-resource language pairs are oscillatory.

Model	Resource Level		
	Low	Medium	High
SMaLL100	−0.39	−0.14	−0.27
M2M (S)	−0.82	−0.67	−0.49
M2M (M)	−0.80	−0.68	−0.15
M2M (L)	−0.85	−0.38	—

Table 5.3: Model-based average Pearson correlation scores between COMET-22 and hallucination rates across all language pairs at each resource level.

▸ **Toxic hallucinations can be traced back to the training data.** To assess the prevalence of toxic text in detected hallucinations, we utilized the toxicity wordlists provided by NLLB Team et al. (2022). We found that toxic text predominantly appears in translations out of English and almost exclusively for low-resource directions. For instance, over 1 in 8 hallucinations in Tamil contain toxic text. Interestingly, these hallucinations exhibit high lexical overlap among them and are repeated across models for multiple unique source sentences. Moreover, they are not necessarily reduced by scaling up the model size. These observations suggest that these hallucinations are likely to be traced back to the training data, aligning with observations in Raunak et al. (2021) and Guerreiro et al. (2023a). In fact, upon inspecting the corpora that were used to create the training data, we found reference translations that exactly match the toxic hallucinations. Additionally, we found that these hallucinations can propagate through model distillation, as evidenced by SMaLL100 generating copies of its teacher model's toxic hallucinations. This highlights the necessity of rigorously filtering training data to ensure safe use of these models.

5.3.3 Beyond English-centric translation

We shift our focus to translation directions that do not involve English, typically corresponding to directions with less supervision during training.

▸ **Trends are largely similar to English-centric directions.** Table 5.5 reveals trends that largely mirror those observed in the English-centric setup:[18] (i) hallucinations are more frequent in low-resource directions;[19] (ii) SMaLL100 significantly outperforms the M2M models

18: Comparing absolute hallucination rates between the two setups is not advised, as they involve different translation directions, which may render such comparisons unreliable.

19: We considered the resource level of the language pair to be the smallest resource level between the two languages.

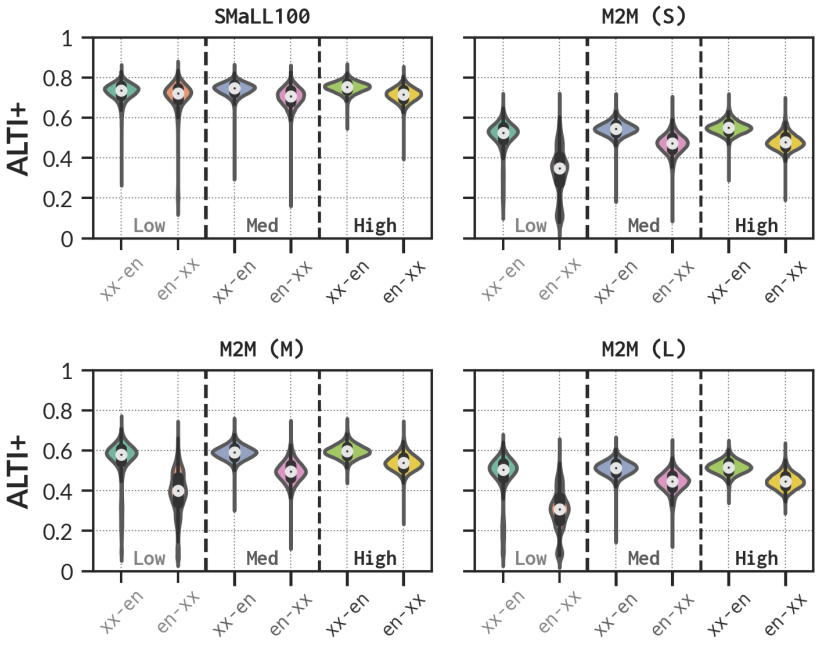


Figure 5.3: Model-based distribution of ALTI+ scores for `en-xx` and `xx-en` directions. Language pair specific distributions for each model can be found in Appendix B.4.

Table 5.4: Average hallucination rates (%) across all LPs at each direction (into or out of English).

LPs	SMaLL100	M2M (S)	M2M (M)	M2M (L)
`xx-en`	0.221	1.756	2.290	2.483
`en-xx`	1.022	6.152	4.110	3.169

in low-resource language pairs; and (iii) scaling up to M2M (L) consistently yields substantial improvements over the smaller M2M models in low- and mid-resource directions. Additionally, the trends related to hallucination types are also similar: detached hallucinations are more prevalent in low-resource directions, while oscillatory hallucinations overwhelmingly dominate in mid- and high-resource ones.

▸ **Less supervised language pairs exhibit extremely high hallucination rates.** As expected, models struggle more with hallucinations for directions with less or even no supervision during training, such as `ro-hy` and `af-zu`.[20] For instance, M2M (M) hallucinates for nearly half of the translations in these directions.

20: These translation directions are completely unsupervised: the languages are not in the same language family and neither of them is a M2M bridge language.

5.3.4 Translation on specialized domains

We now study hallucinations in medical data using the TICO dataset. We compare hallucination rates with the Flores dataset in 18 directions.

▸ **Hallucinations are not exacerbated under medical domain data.** Table 5.6 reveals that hallucination rates for the TICO data are not consistently higher than those observed for the Flores data. This finding diverges from previous works that investigated hallucinations in specialized domain data (Müller et al., 2020; Wang and Sennrich, 2020). We hypothesize that, unlike the smaller models typically trained on limited datasets from a single domain used in those works, the concept of "domain shift" is not as pronounced for M2M models. These models are not only much larger but, crucially, they are trained on a massive dataset containing over 7B sentences from various domains.

Table 5.5: Fraction of LPs on the non-English-centric setup for which models produce at least one hallucination, and average hallucination rate (and median, in subscript) across all LPs at each resource level.

Model	Low Resource		Mid Resource		High Resource	
	LP Fraction	Rate (%)	LP Fraction	Rate (%)	LP Fraction	Rate (%)
SMaLL100	5/10	$2.160_{0.02}$	6/13	$0.054_{0.00}$	1/2	$0.025_{0.02}$
M2M (S)	10/10	$12.61_{1.79}$	12/13	$0.467_{0.05}$	1/2	$0.075_{0.07}$
M2M (M)	7/10	$12.22_{2.41}$	7/13	$0.172_{0.05}$	0/2	$0.000_{0.00}$
M2M (L)	6/10	$6.580_{2.02}$	4/13	$0.077_{0.00}$	0/2	$0.000_{0.00}$

Table 5.6: Delta average hallucination rate at each resource level for FLORES and TICO medical data. Positive values indicate higher rates for TICO.

Resource Level	SMaLL100	M2M (S)	M2M (M)	M2M (L)
Low	−0.019	−1.516	−1.317	−0.412
Mid	0.021	0.080	0.095	−0.013
High	−0.008	0.007	0.000	0.000

This vast training set potentially mitigates the impact of domain shift and, consequently, reduces its influence on hallucinations.

5.4 Mitigation of Hallucinations through Fallback Systems

We now explore the potential of mitigating hallucinations and improving overall translation quality by employing a simple hybrid setup that can take advantage of multiple systems with possible complementary strengths. Put simply, we leverage an alternative system as a fallback when the original model produces hallucinations. Our analysis is focused on the more extensive English-centric setup.

5.4.1 Employing models of the same family as fallback systems

We begin by analyzing the performance of same-family models when employed as fallback systems for one another (e.g., using SMaLL100, M2M (M), and M2M (L) as fallbacks for M2M (S)).[21]

21: For simplicity, we consider the distilled SMaLL100 as a model from the M2M family.

▸ **Detached hallucinations are particularly *sticky* across M2M models.** Figure 5.4 reveals that when employing M2M models as fallback systems, reversal rates—percentage of hallucinations from the original system that are reversed by the fallback system—are consistently higher for oscillatory hallucinations than for detached hallucinations. These findings not only align with those in Guerreiro et al. (2023a), where oscillatory hallucinations were found to be less related to model defects, but also further highlight the close connection between detached hallucinations and the training data. This connection can help explain their *stickiness*: since the M2M models share the same training data, reversing these hallucinations using other M2M variants as fallbacks is more challenging. Interestingly, we also observe that M2M (L) particularly struggles to reverse the detached hallucinations generated by its distilled counterpart SMaLL100, suggesting that model defects can persist and be shared during distillation.

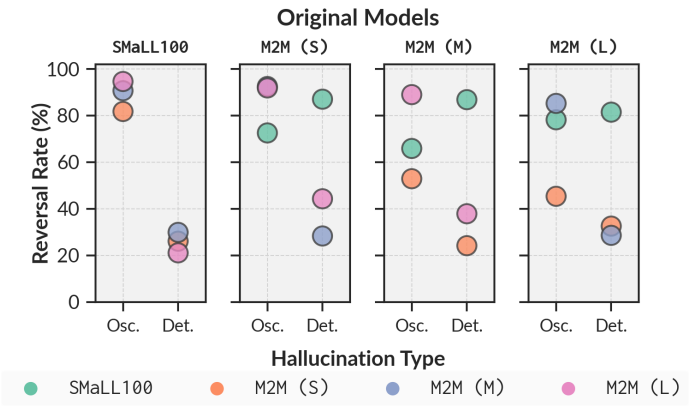


Figure 5.4: Reversal rates for oscillatory (`Osc.`) and detached (`Det.`) hallucinations when using models of the same family as fallback systems.

n sesema

- **Scaling up within the model family is not an effective strategy for mitigating hallucinations.** In line with our findings in Section 5.3.2, Figure 5.4 shows that reversal rates using SMaLL100 as a fallback system are higher for detached hallucinations than for oscillatory hallucinations. Although SMaLL100 is a distilled M2M model, its training data, training procedure, and architecture differ from those of the M2M models. This distinction may make it a more complementary fallback system to other M2M models than simply scaling up within the same model family. This suggests that merely increasing the scale of models within the same family is not an effective strategy for mitigating hallucinations, and exploring alternative models with different training procedure and data may yield further improvements. We will analyze this alternative strategy in the next section.

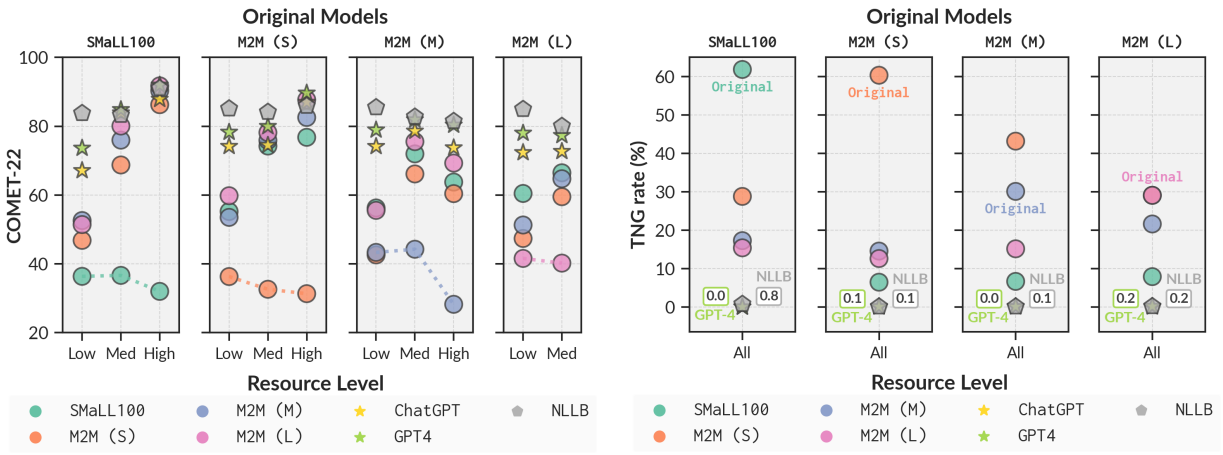
5.4.2 Employing diverse fallback systems

Building on the findings from the previous section, we will investigate how using models outside the M2M family can be employed to mitigate hallucinations and improve translation quality. We will test this approach with two different models: (i) we will prompt the GPT models as detailed in Section 5.1, and (ii) we will use a high-quality 3.3B parameter model from the NLLB family of NMT models (NLLB) (NLLB Team et al., 2022).

- **Diverse fallback systems can significantly improve translation quality.** Figure 5.5a shows that external fallback systems can significantly enhance translation quality of originally hallucinated translations compared to same-family models.[22] This improvement is most pronounced for low-resource directions, where both the LLMs and NLLB consistently boost translation quality. Moreover, NLLB generally outperforms the GPT models for low- and mid-resource languages, aligning with the findings in Hendy et al. (2023a), which found that these models lag behind supervised models in these directions. Nonetheless, even for these language pairs, GPT models still surpass dedicated M2M translation systems when used as a fallback system, underscoring the limitations of relying on same-family models as fallbacks.

- **Oscillatory hallucinations are practically non-existent when LLMs are the fallback system.** Figure 5.5b demonstrates another

22: For reference, in the English-centric setup, the averaged COMET-22 corpus-level scores for low-, mid-, and high-resource LPs obtained with `M2M(L)` are 73.63, 86.54, and 87.19, respectively.



(a) Translation quality.

(b) Prevalence of oscillatory hallucinations.

Figure 5.5: We analyse overall translation quality improvements on the original model hallucinated translations (represented with dashed lines) across different resource levels via COMET-22 scores in (a), and overall prevalence of oscillatory hallucinations among the fallback translations in (b).

benefit of using diverse fallback systems: oscillatory hallucinations are almost completely eliminated. Consistent with our findings in Section 5.2, we observe that LLMs produce very few, if any, oscillations, slightly improving the rates obtained with NLLB. This provides further evidence that, although hallucinations obtained via prompting LLMs may still occur, they exhibit different properties and surface forms. Investigating and understanding these differences presents a relevant research direction for future work.

5.5 Conclusion

We have comprehensively investigated hallucinations in massively multilingual translation models, exploring a wide range of translation scenarios that remained overlooked in previous work.

Our analysis provided several key insights on the prevalence and properties of hallucinations across models of different scale, translation directions, and data conditions, including: the prevalence of hallucinations across multiple translation directions across different resource levels and beyond English-centric directions; the emergence of toxicity in hallucinations; how model distillation may bring reduced hallucination rates; and, the effect of scaling up within the same model family on the prevalence of hallucinations. We also examined how fallback systems can mitigate hallucinations and improve translation quality. We found that hallucinations can be *sticky* and difficult to reverse when using models that share the same training data and training regime. However, leveraging diverse fallbacks can significantly improve translation quality and virtually eliminate pathologies such as oscillatory hallucinations.

Our work has had impact across several areas of the literature. In the domain of hallucination mitigation, we have directly influenced the studies of Waldendorf et al. (2024) and Sennrich et al. (2024). Waldendorf et al. (2024) made use of our findings in their development of

a contrastive decoding strategy that identifies translations by maximizing the log-likelihood difference between a model and its variant with reduced encoder output contribution.[23] As one of the first studies examining hallucinations generated by large language models in a multilingual context, our work has become a key reference point for subsequent work investigating hallucinations in LLMs more broadly. Furthermore, our identification of critical issues in translation with general-purpose large language models (such as ChatGPT and GPT-4), particularly output verbosity, has sparked important follow-up research. For instance, Briakou et al. (2024) investigate how verbose LLM translations — arising from various factors such as safety concerns, copyright considerations, insufficient context, or ambiguity — impact automatic evaluation methods.

23: In practice, the authors try different strategies to reduce the contribution from the encoder (which can be read as reducing the contribution from the source text representations), such as using uniform cross-attention scores, or using a much smaller encoder of the same model family.

In the preceding three chapters, we have explored methods for detecting and mitigating hallucinations in machine translation, as well as analyzed the internal mechanisms of translation models that lead to these problematic outputs. A recurring theme throughout this work, particularly in detection efforts, is the challenge of automatically evaluating translation quality. Notably, we have observed that even state-of-the-art quality estimation systems, such as COMET-QE (Rei et al., 2020b) and CometKiwi (Rei et al., 2022b), often struggle to adequately penalize hallucinations. Moreover, these systems typically provide only a single numerical score, which can be difficult to interpret meaningfully. In light of these limitations, we now turn our attention to bridging this gap by developing more interpretable and robust metrics for assessing translation quality.

Part III

Interpretable Evaluation

Highlights and Structure

TOWARDS INTERPRETABLE PREDICTIONS AND EVALUATION

▸ **HIGHLIGHTS AND CONTRIBUTIONS** We start with a brief detour from machine translation to explore ideas on selective rationalizers— models trained to extract sparse spans of input text that directly inform downstream predictions. This exploration introduces SPECTRA, a novel framework for the deterministic extraction of structured rationales in text classification tasks. SPECTRA offers a unified approach for extracting interpretable highlights and matchings with diverse constraints. Through comparison between deterministic and stochastic rationalizers, we demonstrate improved performance and stability in rationalization for sentiment classification and natural language inference tasks.

Building upon these interpretability concepts, we return to the domain of machine translation with the introduction of a new metric for automatic evaluation of translation quality, XCOMET. This metric, unlike regression-based metrics that exclusively provide numerical scores as estimates of translation quality, provides quality reports that include spans of the translation text corresponding to errors along with their severities (e.g., whether errors are minor, major, or critical). We achieve this by combining sentence-level evaluation with error span detection via a curriculum, attaining high performance across sentence-level, system-level, and error span prediction tasks. Most importantly, it achieves this while providing greater interpretability of machine translation quality assessment through fine-grained error analysis and demonstrating more robustness than other widely used metrics to critical errors such as hallucinations.

▸ **PUBLICATIONS** The ideas and work presented in the following chapters were first published in:[24]

24: *: joint first-authorship.

- **Nuno M. Guerreiro** and André F.T. Martins. *SPECTRA: Sparse Structured Text Rationalization*. In Proc. EMNLP 2021.
- **Nuno M. Guerreiro***, Ricardo Rei*, Daan van Stigt, Luísa Coheur, Pierre Colombo, André F.T. Martins. *xCOMET: Transparent Machine Translation Evaluation through Fine-grained Error Detection*. In Transactions of the Association for Computational Linguistics, 2024.

6 Sparse Structured Text Rationalization

One, if not the main, drawback of rationalizers is that it is difficult to train the generator and the predictor jointly under instance-level supervision (Jain et al., 2020). Hard attention mechanisms that stochastically sample rationales employ regularization to encourage sparsity and contiguity, and make it necessary to estimate gradients using the score function estimator (SFE), also known as REINFORCE (Williams, 1992), or reparameterized gradients (Kingma and Welling, 2014; Jang et al., 2017). Both of these factors substantially complicate training by requiring sophisticated hyperparameter tuning and lead to brittle and fragile models that exhibit high variance over multiple runs. Other works use strategies such as top-k to map token-level scores to rationales, but also require gradient estimations to train both modules jointly (Chang et al., 2020; Paranjape et al., 2020). In turn, sparse attention mechanisms (Treviso and Martins, 2020) are deterministic and have exact gradients, but lack a direct way to control sparsity and contiguity in the rationale extraction. This raises the question: *how can we build an easy-to-train fully differentiable rationalizer that allows for flexible constrained rationale extraction?*

To answer this question, we propose **SPECTRA** (**sparse structured text rationalization**), which employs LP-SparseMAP (Niculae and Martins, 2020), a constrained structured prediction algorithm, to provide a deterministic, flexible and modular rationale extraction process. We exploit our method's inherent flexibility to extract highlights and interpretable text matchings with a diverse set of constraints.

Our contributions are:

- We present a unified framework for deterministic extraction of structured rationales (§6.2) such as constrained highlights and matchings;
- We show how to add constraints on the rationale extraction, and experiment with several structured and hard constraint factors, exhibiting the modularity of our strategy;
- We conduct a rigorous comparison between deterministic and stochastic rationalizers (§6.3) for both highlights and matchings extraction.

Experiments on selective rationalization for sentiment classification and natural language inference (NLI) tasks show that our proposed approach achieves better or competitive performance and similarity with human rationales, while exhibiting less variability and easing rationale regularization when compared to previous approaches.

Table 6.1: Positioning of our approach in the literature of rationalization for highlights extraction. Our method is an easy-to-train fully differentiable deterministic rationalizer that allows for flexible rationale regularization.

Method	Deterministic Training	Exact Gradients	Constrained Extraction	Encourages Contiguity
Lei et al. (2016)	✗	✗	✗	✓
Bastings et al. (2019)	✗	✗	✓	✓
Treviso and Martins (2020)	✓	✓	✗	✗
SPECTRA (ours)	✓	✓	✓	✓

6.1 Structured Prediction on Factor Graphs

Finding a rationale under constraints such as sparsity and contiguity is a structured prediction problem, which involves searching over a very large and combinatorial space. We assume that a rationale $\boldsymbol{z}$ can be represented as an L-dimensional binary vector. For example, in highlights extraction, L is the number of words in the document and $\boldsymbol{z}$ is a binary mask selecting the relevant words; and in the extraction of matchings, $L = L_P \times L_H$ and $\boldsymbol{z}$ is a flattened binary vector whose entries indicate if a premise word is aligned to a word in the hypothesis. We let $\mathcal{Z} \subseteq \{0,1\}^L$ be the set of rationales that satisfy the given constraints, and let $\boldsymbol{s} = \text{gen}(\boldsymbol{x}) \in \mathbb{R}^L$ be a vector of scores.

▸ **Factor Graph.** We consider problems that consist of multiple interacting subproblems. Niculae and Martins (2020) present structured differentiable layers, which decompose a given problem into simpler subproblems, instantiated as local factors that must agree when overlapped. Formally, we assume a factor graph $\mathcal{F}$, where each factor $f \in \mathcal{F}$ corresponds to a subset of variables. We denote by $\boldsymbol{z}_f = (\boldsymbol{z}_i)_{i \in f}$ the vector of variables corresponding to factor f. Each factor has a local score function $h_f(\boldsymbol{z}_f)$. Examples are **hard constraint factors**, which take the form

$$h_f(\boldsymbol{z}_f) = \begin{cases} 0 & \text{if } \boldsymbol{z}_f \in \mathcal{Z}_f \\ -\infty & \text{otherwise,} \end{cases} \tag{6.1}$$

where $\mathcal{Z}_f$ is a polyhedral set imposing hard constraints (see Table 6.2 for examples); and **structured factors**, which define more complex functions with structural dependencies on $\boldsymbol{z}_f$, such as

$$h_f(\boldsymbol{z}_f) = \sum_{i=1}^{L-1} r_{i,i+1} z_{i,i+1}, \tag{6.2}$$

where $r_{i,i+1} \in \mathbb{R}$ are edge scores, which together define a **sequential factor**. We require that for any factor the following local subproblem is tractable:

$$\hat{\boldsymbol{z}}_f = \arg \max_{\boldsymbol{z}_f \in \{0,1\}^{|f|}} \boldsymbol{s}_f^\top \boldsymbol{z}_f + h_f(\boldsymbol{z}_f). \tag{6.3}$$

▸ **MAP inference.** The problem of identifying the highest-scoring global structure, known as **maximum *a posteriori*** (MAP) **inference**, is written as:

$$\hat{\boldsymbol{z}} = \arg \max_{\boldsymbol{z} \in \{0,1\}^L} \left(\boldsymbol{s}^\top \boldsymbol{z} + \sum_{f \in \mathcal{F}} h_f(\boldsymbol{z}_f)\right) := \arg \max_{\boldsymbol{z} \in \{0,1\}^L} \text{score}(\boldsymbol{z}; \boldsymbol{s}). \tag{6.4}$$

The objective being maximized is the global score function $\text{score}(\boldsymbol{z}; \boldsymbol{s})$,

which combines information coming from all factors. The solution of the MAP problem is a vector $\hat{\boldsymbol{z}}$ whose entries are zeros and ones. However, it is often difficult to obtain an exact maximization algorithm for complex structured problems that involve interacting subproblems that impose global agreement constraints.

▸ **Gibbs distribution and sampling.** The global score function can be used to define a Gibbs distribution $p(\boldsymbol{z}; \boldsymbol{s}) \propto \exp(\text{score}(\boldsymbol{z}; \boldsymbol{s}))$. The MAP in (6.4) is the mode of this distribution. Sometimes (e.g. in stochastic rationalizers) we want to sample from this distribution, $\hat{\boldsymbol{z}} \sim p(\boldsymbol{z}; \boldsymbol{s})$. Exact, unbiased samples are often intractable to obtain, and approximate sampling strategies have to be used, such as perturb-and-MAP (Papandreou and Yuille, 2011; Corro and Titov, 2019a,b). These strategies necessitate gradient estimators for end-to-end training, which are often obtained via REINFORCE (Williams, 1992) or reparametrized gradients (Kingma and Welling, 2014; Jang et al., 2017).

▸ **LP-MAP inference.** In many cases, the MAP problem (6.4) is intractable due to the overlapping interaction of the factors $f \in \mathcal{F}$. A commonly used relaxation is to replace the integer constraints $\boldsymbol{z} \in \{0, 1\}^L$ by continuous constraints, leading to:

$$\hat{\boldsymbol{z}} = \arg \max_{\boldsymbol{z} \in [0,1]^L} \text{score}(\boldsymbol{z}; \boldsymbol{s}). \tag{6.5}$$

The problem above is known as LP-MAP inference (Wainwright and Jordan, 2008). In some cases (for example, when the factor graph $\mathcal{F}$ does not have cycles), LP-MAP inference is *exact*, i.e., it gives the same results as MAP inference. In general, this does not happen, but for many problems in NLP, LP-MAP relaxations are often nearly optimal (Koo et al., 2010; Martins et al., 2015). Importantly, computation in the hidden layer of these problems may render the network unsuitable for gradient-based training, as with MAP inference.

▸ **LP-SparseMAP inference.** The optimization problem respective to LP-SparseMAP is the ℓ_2 regularized LP-MAP (Niculae and Martins, 2020):

$$\hat{\boldsymbol{z}} = \arg \max_{\boldsymbol{z} \in [0,1]^L} \left(\text{score}(\boldsymbol{z}; \boldsymbol{s}) - \frac{1}{2} \|\boldsymbol{z}\|^2 \right). \tag{6.6}$$

Unlike MAP and LP-MAP, the LP-SparseMAP relaxation is suitable to train with gradient backpropagation. Moreover, it favors sparse vectors $\hat{\boldsymbol{z}}$, i.e., vectors that have only a few non-zero entries. One of the most appealing features of this method is that it is modular: an arbitrary complex factor graph can be instantiated as long as a MAP oracle for each of the constituting factors is provided. This approach generalizes SparseMAP (Niculae et al., 2018), which requires an exact MAP oracle for the factor graph in its entirety. In fact, LP-SparseMAP recovers SparseMAP when there is a single factor $\mathcal{F} = \{f\}$. By only requiring a MAP oracle for each $f \in \mathcal{F}$, LP-SparseMAP makes it possible to instantiate more expressive factor graphs for which MAP is typically intractable. Table 6.2 lists several logic constraint factors which can be used in this framework.

Table 6.2: Collection of logic factors and its imposed constraints. Each of these factors defines a constraint set $\mathcal{Z}_f$ as described in §6.1.

Factor	Constraint
XOR	$\sum z_k = 1$
AtMostOne	$\sum z_k \leq 1$
BUDGET	$\sum z_k \leq B$

6.2 Deterministic Structured Rationalizers

The idea behind our approach for selective rationalization is very simple: leverage the inherent flexibility and modularity of LP-SparseMAP for constrained, deterministic and fully differentiable rationale extraction.

6.2.1 Highlights extraction

▸ **Model Architecture.** We use the model setting described in Chapter 2. First, a generator model produces token-level scores $s_i, i \in \{1, \dots, L\}$. We propose replacing the current rationale extraction mechanisms (e.g. sampling from a Bernoulli distribution, or using sparse attention mechanisms) with an LP-SparseMAP extraction layer that computes token-level values $\hat{\boldsymbol{z}} \in [0, 1]^L$, which are then used to mask the original sequence for prediction. Due to LP-SparseMAP's propensity for sparsity, many entries in $\hat{\boldsymbol{z}}$ will be zero, which approaches what is expected from a binary mask.

▸ **Factor Graphs.** The definition of the factor graph $\mathcal{F}$ is central to the rationale extraction, as each of the local factors $f \in \mathcal{F}$ will impose constraints on the highlight. We start by instantiating a factor graph with L binary variables (one for each token) and a pairwise factor for every pair of contiguous tokens:

$$\mathcal{F} = \{\mathsf{PAIR}(z_i, z_{i+1}; r_{i,i+1}) : 1 \le i < L\}, \tag{6.7}$$

which yields the binary pairwise MRF (§6.1)

$$\text{score}(\boldsymbol{z}; \boldsymbol{s}) = \sum_{i=1}^{L} s_i z_i + \sum_{i=1}^{L-1} r_{i,i+1} z_i z_{i+1}. \tag{6.8}$$

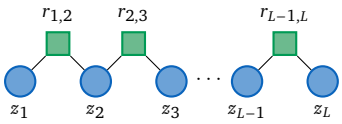


Figure 6.1: The factor graph $\mathcal{F}$ in Equation (6.7).

Instantiating this factor with non-negative edge scores, $r_{i,i+1} \ge 0$, encourages contiguity on the rationale extraction. Making use of the modularity of the method, we impose sparsity by further adding a BUDGET factor (see Table 6.2):

$$\begin{aligned}\mathcal{F} = &\{\mathsf{PAIR}(z_i, z_{i+1}; r_{i,i+1}) : 1 \le i < L\} \\ &\cup \{\mathsf{BUDGET}(z_1, \dots, z_L; B)\}.\end{aligned} \tag{6.9}$$

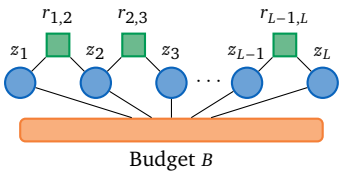


Figure 6.2: The factor graph $\mathcal{F}$ in Equation (6.9).

The size of the rationale is constrained to be, at most, B% of the input document size. Intuitively, the lower the B, the shorter the extracted rationales will be. Notice that this graph is composed of L local factors. Thus, LP-SparseMAP would have to enforce agreement between all these factors in order to compute $\boldsymbol{z}$. Interestingly, factor graph representations are usually not unique. In our work, we instantiate an equivalent formulation of the factor graph in Eq. 6.7 that consists of a single factor, **H:SeqBudget**. This factor can be seen as an extension of that of the LP-Sequence model in Niculae and Martins (2020): a linear-chain Markov factor with MAP provided by the Viterbi algorithm (Viterbi, 1967; Rabiner, 1989). The difference resides in the additional budget constraints that are incorporated in the MAP decoding. These

constraints can be handled by augmenting the number of states in the dynamic program to incorporate how many words in the budget have already been consumed at each time step, leading to time complexity $\mathcal{O}(LB)$.

6.2.2 Matchings extraction

▸ **Model Architecture.** Our architecture is inspired by ESIM (Chen et al., 2017). First, a generator model encodes two documents $\boldsymbol{x}_P$, $\boldsymbol{x}_H$ separately to obtain the encodings $(\tilde{\boldsymbol{h}}^P_1, \ldots, \tilde{\boldsymbol{h}}^P_{L_P})$ and $(\tilde{\boldsymbol{h}}^H_1, \ldots, \tilde{\boldsymbol{h}}^H_{L_H})$, respectively. Then, we compute alignment dot-product pairwise scores between the encoded representations to produce a score matrix $\boldsymbol{S} \in \mathbb{R}^{L_P \times L_H}$ such that $s_{ij} = \langle \tilde{\boldsymbol{h}}^P_i, \tilde{\boldsymbol{h}}^H_j \rangle$. We use LP-SparseMAP to obtain $\boldsymbol{Z}$, a constrained structured symmetrical alignment $\boldsymbol{Z}$ in which $z_{ij} \in [0, 1]$, as described later. Then, we "augment" each word in the premise and hypothesis with the corresponding aligned weighted average by computing $\bar{\boldsymbol{h}}^P_i = \left[\tilde{\boldsymbol{h}}^P_i, \sum_j z_{ij} \tilde{\boldsymbol{h}}^H_j\right]$ and $\bar{\boldsymbol{h}}^H_j = \left[\tilde{\boldsymbol{h}}^H_j, \sum_i z_{ji} \tilde{\boldsymbol{h}}^P_i\right]$, and separately feed these vectors to another encoder and pool to find representations $\boldsymbol{r}^P$ and $\boldsymbol{r}^H$. Finally, the feature vector $\boldsymbol{r} = [\boldsymbol{r}^P, \boldsymbol{r}^H, \boldsymbol{r}^P - \boldsymbol{r}^H, \boldsymbol{r}^P \odot \boldsymbol{r}^H]$ is fed to a classification head for the final prediction. We also experiment with a strategy in which we assume that the hypothesis is known and the premise is masked for *faithful* prediction. We consider $\bar{\boldsymbol{h}}^P_i = \left[\sum_j z_{ij} \tilde{\boldsymbol{h}}^H_j\right]$, such that the only information about the premise that the model has to make a prediction comes from the alignment and its masking of the encoded representation.

▸ **Factor Graphs.** We instantiate three different factor graphs for matchings extraction. The first – **M:XorAtMostOne** – is the same as the LP-Matching factor used in Niculae and Martins (2020) with one XOR factor per row and one AtMostOne factor per column:

$$\begin{aligned}\mathcal{F} = &\{\mathsf{XOR}(z_{i1}, \ldots, z_{iL_H}) : 1 \le i \le L_P\} \\ &\cup \{\mathsf{AtMostOne}(z_{1j}, \ldots, z_{L_P j}) : 1 \le j \le L_H\}\end{aligned} \tag{6.10}$$

which requires at least one active alignment for each word of the premise, since the i^{th} word in the premise **must** be connected to the hypothesis. The j^{th} word in the hypothesis, however, is not constrained to be aligned to any word in the premise. In the second factor graph – **M:AtMostOne2** – we alleviate the XOR restriction on the premise words to an AtMostOne restriction. The expected output is a sparser matching for there is no requirement of an active alignment for each word of the premise. The third factor graph – **M:Budget** – allows us to have more refined control on the sparsity of the resulting matching, by adding an extra global BUDGET factor (with budget B) to the factor graph of M:AtMostOne2 so that the resulting matching will have at most B active alignments.

▸ **Stochastic Matchings Extraction.** Prior work for selective rationalization of text matching uses constrained variants of optimal transport to obtain the rationale (Swanson et al., 2020). Their model is end-to-end differentiable using the Sinkhorn algorithm (Cuturi, 2013). Thus, in order to provide a comparative study of stochastic and deterministic methods for rationalization of text matchings, we implement a

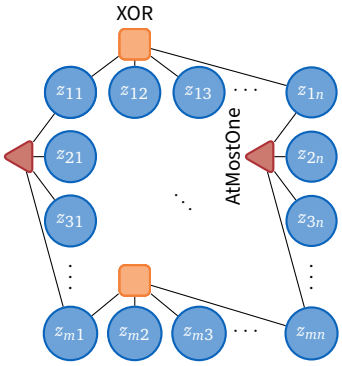


Figure 6.3: The factor graph $\mathcal{F}$ in Equation (6.10).

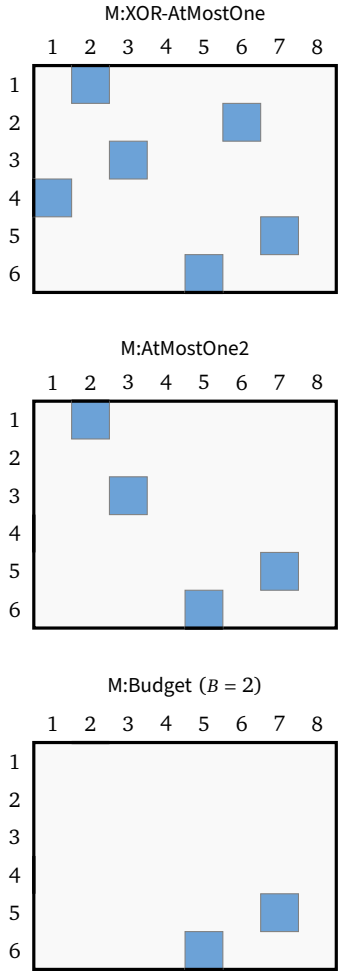


Figure 6.4: Toy example of matchings extracted with the different SPECTRA strategies for a synthetic 6×8 matrix.

perturb-and-MAP rationalizer (§6.1). We perturb the scores s_{ij} by computing $\tilde{\boldsymbol{S}} = \boldsymbol{S} + \boldsymbol{P}$, in which each element of $\boldsymbol{P}$ contains random samples from the Gumbel distribution, $p_{ij} \sim \mathcal{G}(0, 1)$. We utilize these perturbed scores to compute non-symmetrical alignments from the premise to the hypothesis and vice-versa, such that their entries are in $[0, 1]$. At test time, we obtain the most probable matchings, such that their entries are in $\{0, 1\}$. These matchings are such that every word in the premise **must** be connected to a single word in the hypothesis and vice-versa.

6.3 Experimental Setup

6.3.1 Highlights for sentiment classification

- DATA AND EVALUATION. We used the SST, AgNews, IMDB, and Hotels datasets for text classification and the BeerAdvocate dataset for regression. The statistics and details of all datasets can be found in §C.1. The rationale specified lengths, as percentage of each document, for the strategies that impose fixed sparsity are 20% for the SST, AgNews and IMDB datasets, 15% for the Hotels dataset, and 10% for the BeerAdvocate dataset. We evaluate end task performance (Macro F_1 for classification tasks and MSE for regression), and matching with human annotations through token-level F_1 score (DeYoung et al., 2019) for the datasets that contain human annotations.

- BASELINES. We compare our results with three versions of the stochastic (🛟) rationalizer of Lei et al. (2016): the original one – **SFE** – which uses the score function estimator to estimate the gradients; a second one – **SFE w/ Baseline** – which uses SFE with a moving average baseline variance reduction technique; a third – **Gumbel** – in which we employ the Gumbel-Softmax reparameterization (Jang et al., 2017) to reparameterize the Bernoulli variables; and, a fourth – **HardKuma** – in which we employ HardKuma variables (Bastings et al., 2019) instead of Bernoulli variables and use reparameterized gradients for training end-to-end. Moreover, the latter rationalizer employs a Lagrangian relaxation to solve the constrained optimization problem of targeting specific sparsity rates. We also experimented with two deterministic (🎯) strategies that use sparse attention mechanisms: a first that utilizes **sparsemax** (Martins and Astudillo, 2016), and a second that utilizes **fusedmax** (Niculae and Blondel, 2019) which encourages the network to pay attention to contiguous segments of text, by adding an additional total variation regularizer, inspired by the fused lasso. It is a natural deterministic counterpart of the constrained rationalizer proposed by Lei et al. (2016), since the regularization encourages both sparsity and contiguity. The use of fusedmax for this task is new to the best of our knowledge. Similarly to Jain et al. (2020), we found that the stochastic rationalizers of Lei et al. (2016) and its variants (SFE, SFE w/ Baseline and Gumbel) require cumbersome hyperparameter search and tend to degenerate in such a way that the generated rationales are either the whole input text or empty text. Thus, at inference time, we follow the strategy proposed by Jain et al. (2020) and restrict the generated rationale to a specified length ℓ via two mappings: **contiguous**, in which the span of length ℓ, out of all the spans of this length, whose token-level scores cumulative sum is the highest is selected; and **top-k**, in which the

ℓ tokens with highest token-level scores are selected. Contrary to (Jain et al., 2020), for the rationalizer of Bastings et al. (2019) (HardKuma), we carefully tuned both the model hyperparameters and the Lagrangian relaxation algorithm hyperparameters, so as to use the deterministic policy in testing time that they propose.[1] All implementation details can be found in §C.3. We also report the full-text baselines for each dataset in Table 6.3.

1: We have found that using the deterministic policy at testing time proposed by Bastings et al. (2019) instead of the top-k or contiguous strategies is critical to achieve good performance with the HardKuma rationalizer.

Table 6.3: Model predictive performances across datasets using full-text. We report mean and min/max F_1 scores across five random seeds on test sets for all datasets but Beer where we report MSE.

Dataset	Performance Score
SST	0.83 (.82/.83)
AgNews	0.93 (.93/.93)
IMDB	0.90 (.90/.90)
Beer	0.019 (.18/.21)
Hotels	0.87 (.86/.88)

6.3.2 Matchings for natural language inference

▸ **Data and Evaluation.** We used the English language SNLI and MNLI datasets (Bowman et al., 2015; Chen et al., 2017). We evaluate end task performance for both datasets. For the experiments with the M:Budget, we used a fixed budget of $B = 4$ for SNLI and $B = 6$ for MNLI. We also conduct further experiments with the HANS dataset (McCoy et al., 2019) which aims to analyse the use of linguistic heuristics (lexical overlap, constituent and subsequence heuristics) of NLI systems. The statistics and details of each dataset can be found in §C.2.

▸ **Baselines.** We compare our results with variants of constrained optimal transport for selective rationalization employed by Swanson et al. (2020): relaxed 1:1, which is similar in nature to our proposed M:AtMostOne2 factor; and exact $k = 4$ similar to our proposed M:Budget with budget $B = 4$. We also replicate the LP-matching implementation of Niculae and Martins (2020) which consists of the original ESIM model described in §6.2.2 with $\boldsymbol{Z}$ as the output of the LP-SparseMAP problem with a M:XorAtMostOne factor. Importantly, both these models aggregate the encoded premise representation with the information that comes from the alignment. All implementation details can be found in §C.3. We also report the ESIM baselines in Table 6.4.

Table 6.4: ESIM predictive performances for SNLI and MNLI. We report mean and min/max F_1 scores across three random seeds on test sets for both datasets.

Dataset	Performance Score
SNLI	0.86 (.86/.86)
MNLI	0.74 (.73/.74)

6.4 Results and Analysis

6.4.1 Extraction of text highlights

▸ **Predictive Performance.** We report the predictive performances of all models in Table 6.5. We observe that the deterministic rationalizers that use sparse attention mechanisms generally outperform the stochastic rationalizers while exhibiting lower variability across different random seeds and different datasets. In general and as expected, for the stochastic models, the top-k strategy for rationale extraction outperforms the contiguous strategy. As reported in Jain et al. (2020), strategies that impose a contiguous mapping trade coherence for performance on the end-task. Our experiments also show that HardKuma is the stochastic rationalizer least prone to variability across different seeds, faring competitively with the deterministic methods. The strategy proposed in this paper, H:SeqBudget, fares competitively with the deterministic methods and generally outperforms the stochastic methods. Moreover, similarly to the other deterministic rationalizers, our method exhibits lower variability across different runs. We show examples of highlights extracted by SPECTRA in Figure 6.5.

Table 6.5: Model predictive performances across datasets, for stochastic (🛟) and deterministic (💠) methods. We report mean and min/max F_1 scores across five random seeds on test sets for all datasets but Beer where we report MSE. We bold the best-performing rationalized model(s) for each corpus.

Method	Rationale	SST ↑	AgNews ↑	IMDB ↑	Beer ↓	Hotels ↑
🛟 SFE	top-k	.76 (.71/.80)	.92 (.92/.92)	.84 (.72/.88)	.018 (.016/.020)	.66 (.62/.69)
	contiguous	.71 (.68/.75)	.86 (.85/.86)	.65 (.57/.73)	.020 (.019/.024)	.62 (.34/.72)
🛟 SFE w/ Baseline	top-k	.78 (.76/.80)	.92 (.92/.93)	.82 (.72/.88)	.019 (.017/.020)	.56 (.34/.64)
	contiguous	.70 (.64/.75)	.86 (.84/.86)	.76 (.73/.80)	.021 (.019/.025)	.55 (.34/.69)
🛟 Gumbel	top-k	.70 (.67/.72)	.78 (.73/.84)	.74 (.71/.78)	.026 (.018/.041)	.83 (.73/.92)
	contiguous	.67 (.67/.68)	.77 (.74/.81)	.72 (.72/.73)	.043 (.040/.048)	.74 (.65/.84)
🛟 HardKuma	–	.80 (.80/.81)	.90 (.87/.88)	.87 (.90/.91)	.019 (.016/.020)	.90 (.88/.92)
💠 Sparse Attention	sparsemax	**.82** (.81/.83)	**.93** (.93/.93)	.89 (.89/.90)	.019 (.016/.021)	.89 (.87/.92)
	fusedmax	.81 (.81/.82)	.92 (.91/.92)	.88 (.87/.89)	.018 (.017/.019)	.85 (.77/.90)
💠 SPECTRA (ours)	H:SeqBudget	.80 (.79/.81)	.92 (.92/.93)	**.90** (.89/.90)	**.017** (.016/.019)	**.91** (.90/.92)

Table 6.6: Average size of the extracted rationales using the HardKuma stochastic rationalizer (🛟) and deterministic (💠) sparse attention mechanisms. We report mean and min/max average size across five random seeds.

Method	Rationale	SST	AgNews	IMDB	Beer	Hotels
🛟 HardKuma	–	.15 (.12/.19)	.19 (.18/.19)	.03 (.02/.03)	.08 (.00/.17)	.09 (.07/.12)
💠 Sparse Attention	sparsemax	.17 (.13/.23)	.13 (.11/.15)	.02 (.02/.03)	.11 (.09/.13)	.03 (.02/.04)
	fusedmax	.60 (.14/1.0)	.32 (.10/.66)	.02 (.01/.02)	.26 (.03/.98)	.04 (.01/.08)

6.4.2 Quality of the rationales

▸ **Rationale Regularization.** We report in Table 6.6 the average size of the extracted rationales (proportion of words not zeroed out) across datasets for the stochastic HardKuma rationalizer and for each rationalizer that uses sparse attention mechanisms. The latter strategies do not have any mechanism to regularize the sparsity of the extracted rationales, which leads to variability on the rationale extraction. This is especially the case for the fusedmax strategy, as it pushes adjacent tokens to be given the same attention probability. This might lead to rationale degeneration when the attention weights are similar across all tokens. On the other hand, HardKuma employs a Lagrangian relaxation algorithm to target a predefined sparsity level. We have found that careful hyperparameter tuning is required across different datasets. While, generally, the average size of the extracted rationales does not exhibit considerable variability, some random seeds led to degeneration (the model extracts empty rationales). Remarkably, our proposed strategy utilizes the BUDGET factor to set a predefined desired rationale length, regularizing the rationale extraction while still applying a deterministic policy that exhibits low variability across different runs and datasets (Table 6.5).

▸ **Matching with Human Annotations.** We report token-level F_1 scores in Table 6.7 to evaluate the quality of the rationales for the datasets for which we had human annotations for the test set. We observe that our proposed strategy and HardKuma outperform all the other methods on what concerns matching the human annotations. This was to be expected considering the results shown in Table 6.5 and Table 6.6: the stochastic models other than HardKuma do not fare competitively with the deterministic models and their variability across runs is also reflected on the token-level F_1 scores; and although the

An amber pour with hints of pink and yellow. Fluffy head, good lacing. Smells of high citrus (gf, lemon) and some leafy flower plants. Hops are in there somewhere. Taste has the hops ; nice crispness and [...]

Hazy bright orange in color with a fluffy white head that quickly dissipates, leaving delicate lace. Way too orange looking. Aroma is very mild wheat and subtle spice completely dominated by artificial orange. Smells like tang. [...]

Figure 6.5: Examples of highlights extracted with H:SeqBudget for the BeerAdvocate dataset.

Method	Rationale	Beer	Hotels
SFE	top-k	.19 (.13/.30)	.16 (.12/.30)
	contiguous	.35 (.18/.42)	.14 (.12/.15)
SFE w/ Baseline	top-k	.17 (.14/.19)	.14 (.13/.18)
	contiguous	.41 (.37/.42)	.15 (.14/.15)
Gumbel	top-k	.27 (.14/.39)	.36 (.27/.48)
	contiguous	.42 (.41/.42)	.36 (.29/.48)
HardKuma	–	.37 (.00/.90)	**.52** (.37/.57)
Sparse Attention	sparsemax	.48 (.41/.55)	.17 (.07/.31)
	fusedmax	.39 (.29/.53)	.25 (.09/.31)
SPECTRA (ours)	H:SeqBudget	**.61** (.56/.68)	.37 (.34/.40)

Table 6.7: Evaluation of the rationales through matching with human annotations, for stochastic (🛟) and deterministic (🔷) methods. We report mean token-level F_1 scores and min/max across five random seeds.

rationalizers that use sparse attention mechanisms are competitive with our proposed strategy, the lack of regularization on what comes to the rationale extraction leads to variable sized rationales which is also reflected on poorer matchings. We also observe that, when degeneration does not occur, HardKuma generally extracts high quality rationales on what comes to matching the human annotations. It is also worth remarking that the sparsemax and top-k strategies are not expected to fare well on this metric because human annotations for these datasets are at the *sentence-level*. Our strategy, however, not only pushes for sparser rationales but also encourages contiguity on the extraction.

6.4.3 Extraction of text matchings

- **Predictive Performance.** We report the predictive performances of all models in Table 6.8. Both the strategies that use the LP-SparseMAP extraction layer and our proposed stochastic matchings extractor outperform the OT variants for matchings extraction. We observe that, contrary to the text highlights experiments, the stochastic matchings extraction model does not exhibit noticeably higher variability compared to the deterministic models. In general, the faithful models are competitive with the non-faithful models. Since the latter ones are constrained to only utilize information from the premise that comes from alignments, these results demonstrate the effectiveness of the alignment extraction. As expected, there is a slight trade-off between how constrained the alignment is and the model's predictive performance. This is more noticeable with the M:Budget strategy, the most constrained version of our proposed strategies, in the faithful scenario. We show examples of matchings extracted by SPECTRA in Figure 6.6.

- **Heuristics Analysis with HANS.** We used two different M:AtMostOne2 models for our analysis: a first one trained on MNLI (**Vanilla**), and a second one trained on MNLI augmented (**Augmented**) with 30,000 HANS-like examples (≈ 8% of MNLI original size), replicating the data augmentation scenario in McCoy et al., 2019. We evaluated both models on the HANS evaluation set, which has six subcomponents, each defined by its correct label and the heuristic it addresses. We report the results in Table 6.9. Our models behave similarly to those in McCoy et al., 2019: when we augment the training set with HANS-like examples, the model no longer associates the heuristics to entailment. By observation of the extracted matchings, we noticed that these were similar between the two models. Thus, the effect of the augmented

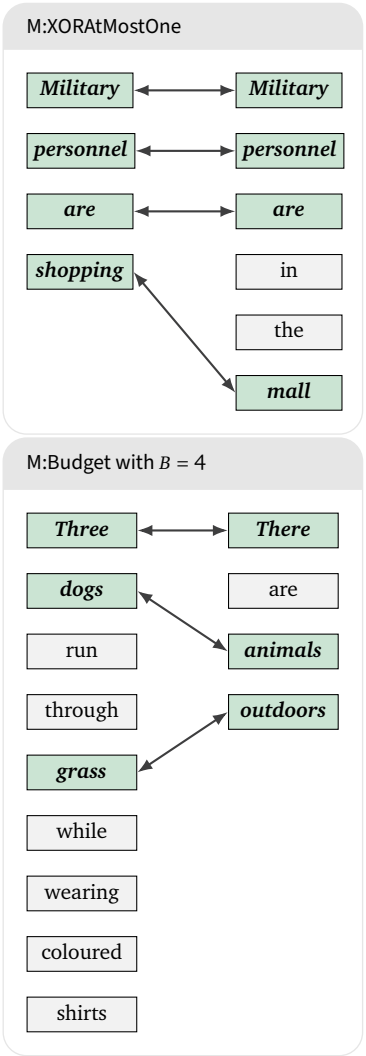


Figure 6.6: Examples of structured matchings extracted with SPECTRA.

Table 6.8: Model performance for stochastic (🛟) and deterministic (✣) methods. We report mean and min/max F_1 scores across three random seeds. We bold the best-performing models for each corpus. † means results come from Swanson et al. (2020).

Matching Structure	SNLI	MNLI
Not Faithful		
✣ OT relaxed 1:1†	.82	–
✣ OT exact $k = 4$†	.81	–
🛟 Gumbel Matching	.85 (.84/.85)	.73 (.72/.73)
✣ M:XorAtMostOne	**.86** (.86/.87)	**.76** (.75/.76)
✣ M:AtMostOne2	**.86** (.86/.87)	**.76** (.75/.76)
✣ M:Budget	.85 (.85/.86)	.75 (.75/.76)
Faithful		
🛟 Gumbel Matching	.85 (.84/.85)	.73 (.72/.73)
✣ M:XorAtMostOne	.85 (.85/.85)	.73 (.72/.73)
✣ M:AtMostOne2	.85 (.85/.85)	.73 (.73/.73)
✣ M:Budget	.82 (.81/.82)	.68 (.67/.68)

Table 6.9: Model performances (accuracy) for the vanilla and augmented models evaluated on all subcomponents of the HANS evaluation set.

HANS Subcomponent	Vanilla	Augmented
Entailment		
Lexical Overlap	.994	.996
Subsequence	.996	1.0
Constituent	.999	1.0
Non-entailment		
Lexical Overlap	.005	.999
Subsequence	.002	1.0
Constituent	.012	1.0

data resides on how the information from the matchings is used after the extraction layer. We show examples of matchings in Figure 6.7.

6.5 Conclusion

In this chapter, we presented SPECTRA, an easy-to-train fully differentiable rationalizer that allows for flexible constrained rationale extraction. We have provided a comparative study with stochastic and deterministic approaches for rationalization, showing that SPECTRA generally outperforms previous rationalizers in text classification and natural language inference tasks. Moreover, it does so while exhibiting less variability than stochastic methods and easing regularization of the rationale extraction when compared to previous deterministic approaches. Our framework constitutes a unified framework for deterministic extraction of different structured rationales.

Our work has been referenced to in several follow-up studies in rationalization (Yu et al., 2021; Chen et al., 2022; Chrysostomou and Aletras, 2022; Fernandes et al., 2022a; Zhao et al., 2022; Liu et al., 2023). One particular interesting work is that of Chen et al. (2022), who have investigated SPECTRA through the lens of its robustness to adversarial attacks, demonstrating that SPECTRA exhibits greater robustness to such attacks compared to full context models.[2] Beyond the literature of rationalization, SPECTRA has also proven valuable in the domain of counterfactual generation, particularly in CREST (Treviso et al., 2023). In this work, counterfactual generation is guided by SPECTRA-extracted rationales, building on the principle that these

2: The main premise is that since these models need to first generate rationales ("rationalizer") before making predictions ("predictor"), they have the potential to ignore noise or adversarially added text by simply masking it out of the generated rationale.

rationales are both *sufficient* and *predictive* of the gold label, thereby enabling the production of high-quality counterfactuals through targeted rationale editing. More broadly, our work has also influenced that of Santos et al. (2024), who adapt the SPECTRA architecture but form rationales according to the pattern associations (word tokens) extracted by a Hopfield pooling layer.

Beyond the applications of rationalization and those from other works that subsequently employed SPECTRA, we ourselves aimed at developing a quality estimator based on selective rationalization in Treviso et al., 2021. However, that attempt proved unsuccessful due to training difficulties. Nevertheless, the core ideas of rationalization—particularly that of extracting sparse spans of input text as explanations to inform predictions—ultimately influenced the development of XCOMET (Guerreiro et al., 2023c), the subject of the next chapter.

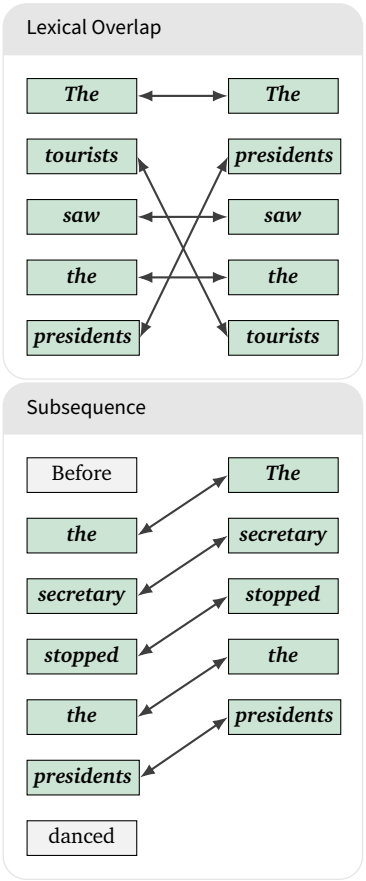


Figure 6.7: Extracted matchings from SPECTRA for the HANS dataset for each of its subcomponents.

xCOMET: MT Evaluation through Fine-grained Error Detection

7

Most metrics for machine translation evaluation are trained to regress on scores obtained through human annotations, by predicting a single sentence-level score representing the quality of the translation hypothesis. However, these single scores do not offer a detailed view into translation errors (e.g., it is not immediate which words or spans of words are wrongly translated). Moreover, as they are obtained by making use of highly complex pre-trained models, they can be difficult to interpret (Leiter et al., 2023; Rei et al., 2023c). One appealing strategy to bring a more detailed view into translation errors is to obtain finer-grained information on error spans through highlighting them and indicating their severity (Fonseca et al., 2019; Perrella et al., 2022; Bao et al., 2023). In fact, this is the strategy adopted in recent works that have employed generative large language models (LLMs) for machine translation evaluation: (i) identify errors within a given translation, subsequently (ii) categorize these errors according to their severity, and finally (iii) infer a sentence-level score from the predicted errors (Fernandes et al., 2023a; Xu et al., 2023c). However, these methods still lag behind dedicated learned metrics when using open LLMs, such as the LLaMA models (Touvron et al., 2023b; Xu et al., 2023c). As it stands, competitive performance with generative strategies remains contingent on utilizing large *proprietary*, *closed* LLMs such as PaLM-2 and GPT-4 (Fernandes et al., 2023a; Kocmi and Federmann, 2023a).

In this chapter, we bridge the gap between these two approaches to machine translation evaluation by introducing **xCOMET**: a *learned* metric that simultaneously performs sentence-level evaluation and error span detection. Through extensive experiments, we show that our metrics leverage the strengths of both paradigms: they achieve state-of-the-art performance in all relevant vectors of evaluation (sentence-level, system-level, and error span prediction), while offering, via the predicted error spans, a lens through which we can analyze translation errors and better interpret the sentence-level scores. We achieve this by employing a curriculum during training that is focused on leveraging high-quality *publicly available* data at both the sentence- and error span level, complemented by synthetic data constructed to enhance the metric's robustness. Moreover, xCOMET is a unified metric (Wan et al., 2022b), supporting all modes of evaluation within a single model. This enables the metric to be used for quality estimation (when no reference is available), or for reference-only evaluation, similarly to BLEURT (when a source is not provided). Crucially, xCOMET also provides sentence-level scores that are directly inferred from the predicted error spans, in the style of AUTOMQM (Fernandes et al., 2023a) and INSTRUCTSCORE (Xu et al., 2023c).

Our contributions can be summarized as follows:

1. We introduce **xCOMET**, a novel evaluation metric that leverages the advantages of regression-based metrics and error span detection to offer a more detailed view of translation errors.

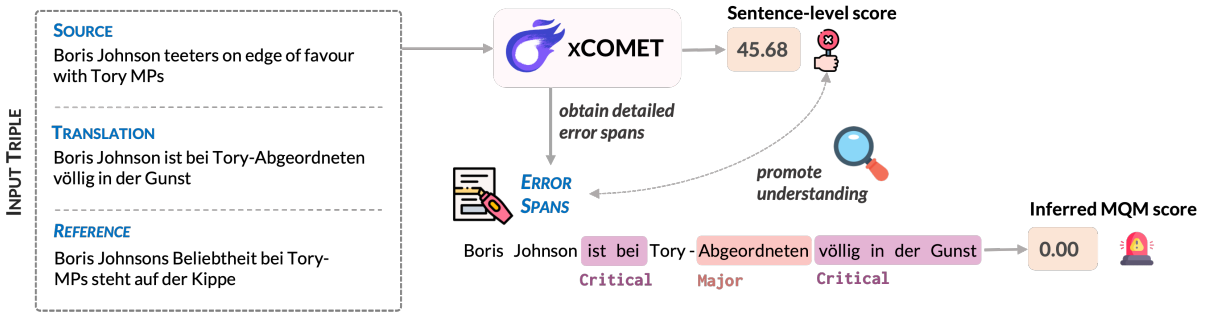


Figure 7.1: The **xCOMET** framework illustrated through a **real** example from the WMT22 News test set: the metric not only provides a sentence-level score, but also predicts translation error spans along with their respective severity. From these spans, we can infer MQM score (following the MQM typology) that informs and highly correlates with the sentence-level score (see Section 8.5). These spans complement the sentence-level score by providing a detailed view into the translation errors.

2. We show that **xCOMET** is a state-of-the-art metric at all relevant vectors of evaluation — sentence-level, system-level, and error span prediction — generally outperforming widely-used neural metrics and generative LLM-based machine translation evaluation.
3. We provide a comprehensive robustness analysis of **xCOMET**, showing that this new suite of metrics identifies the vast majority of localized critical errors and hallucinations.
4. We release two evaluation models: **xCOMET-XL**, with 3.5B parameters, and **xCOMET-XXL**, featuring 10.7B parameters.[1]

1: https://github.com/Unbabel/COMET/

7.1 Design and Methodology of xCOMET

In this section, we describe the methodology behind xCOMET, outlining its model architecture, training settings and corpora, and learning curriculum. We detail how the model is designed to perform both regression and error span detection while adopting a unified input approach for enhanced flexibility and performance.

7.1.1 Model architecture

xCOMET is built upon insights from contributions to the WMT22 Metrics and QE shared tasks (Rei et al., 2022a,b). It is designed to concurrently handle two tasks: sentence-level regression and error span detection. Figure 7.2 illustrates its architecture. We follow the same architecture of the scaled-up version of CometKiwi detailed in Rei et al. (2023a),[2] which uses a large pre-trained encoder model as its backbone encoder model. Importantly, following from our multi-task setup, the model has two prediction heads: (i) a sentence-level *regression* head, which employs a feed-forward network to generate a sentence score, and (ii) a word-level *sequence tagger*, which applies a linear layer to assign labels to each translation token.

We train two **xCOMET** versions — **xCOMET-XL** and **xCOMET-XXL** — using the XL (3.5B parameters) and XXL (10.7B parameters) versions of XLM-R (Goyal et al., 2021).[3]

2: Although we do not write a dedicated chapter on CometKiwi, the model was developed in the context of this thesis work.

3: To the best of our knowledge, these represent the two largest open-source encoder-only models. Curiously, Philipp Koehn, who was the session chair of the oral session on Machine Translation at ACL 2024 where this paper was presented, humorously asked, "*Did you say 11B parameters?*" regarding xCOMET-XXL. Indeed, we did! While an evaluation model with 11 billion parameters may initially seem excessive, such large-scale models can be leveraged beyond their conventional use to benefit the development of smaller, more efficient models through techniques like pruning and quantization. One such example is xCOMET-lite (Larionov et al., 2024), a compressed xCOMET-XXL that surpasses surpasses strong small scale metrics like COMET-22 and BLEURT-20 on the WMT22 metrics despite using 50% fewer parameters.

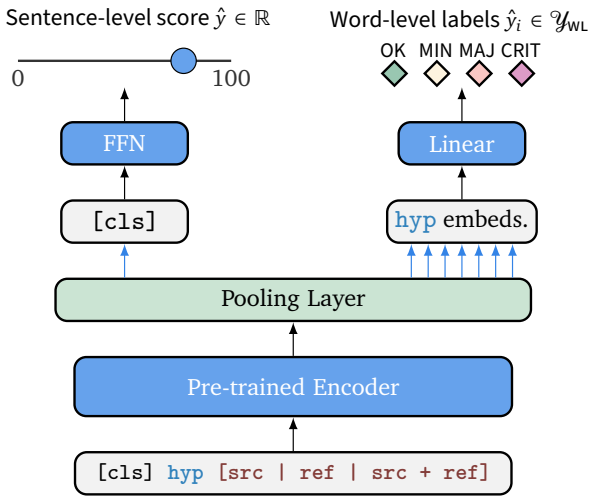


Figure 7.2: Architecture of **XCOMET**. The input to the model starts with a `[cls]` token followed by a `translation` and an `additional input` that will have the source, reference or both. After the pooling layer the `[cls]` token is passed to a feed-forward network to produce a quality score while all subword pieces corresponding to the `translation` are passed to a linear layer that will classify them according to their severity levels, $\mathcal{Y}_{WL} = \{\text{OK, MIN, MAJ, CRIT}\}$.

7.1.2 Fully unified evaluation

XCOMET adopts a unified input approach (Wan et al., 2022b), allowing for all the evaluation scenarios—REF, SRC+REF, and SRC evaluation—under a single model.[4] Thus, the input sequence consists of two parts: (i) the translated sentence $\boldsymbol{t} = [t_1, ..., t_n]$ of length n, and (ii) an additional input containing information from the source, reference, or both. To do so, when a reference is available, we run three distinct forward passes (one for each evaluation scenario), each yielding sentence-level and word-level predictions.

4: Refer back to Section 2.4.3 of Chapter 2.

Training time

For each step, we collect the sentence-level predictions and the word-level logits for each evaluation input format (REF, SRC+REF, and SRC): $\{\hat{y}_{SL}^{SRC}, \hat{y}_{SL}^{REF}, \hat{y}_{SL}^{SRC+REF}\}$, and $\{\hat{\boldsymbol{y}}_{WL}^{SRC}, \hat{\boldsymbol{y}}_{WL}^{REF}, \hat{\boldsymbol{y}}_{WL}^{SRC+REF}\}$.[5]

As we have mentioned before, XCOMET models are trained with supervision from both sentence-level quality assessments, y_{SL}, and word-level severity tags, $\boldsymbol{y}_{WL} = [y_1, \ldots, y_n]$, with $y_i \in \mathcal{Y}_{WL} = \{\text{OK, MIN, MAJ, CRIT}\}$. In the multi-task setting, we use the following loss $\mathcal{L}$ for each input type ($\text{INPUT} \in \{\text{SRC, REF, SRC+REF}\}$):

$$\mathcal{L}_{SL}^{INPUT} = \left(y_{SL} - \hat{y}_{SL}^{INPUT}\right)^2 \tag{7.1}$$

$$\mathcal{L}_{WL}^{INPUT} = -\frac{1}{n}\sum_{i=1}^{n} \alpha_{y_i} \log p(y_i^{INPUT}) \tag{7.2}$$

$$\mathcal{L}^{INPUT} = (1-\lambda)\mathcal{L}_{SL}^{INPUT} + \lambda\mathcal{L}_{WL}^{INPUT}, \tag{7.3}$$

$\boldsymbol{\alpha} \in \mathbb{R}^{|\mathcal{Y}_{WL}|}$ represents the class weights given for each severity label and λ is used to weigh the combination of the sentence and word-level losses.

The final learning objective is the summation of the losses for each input type:

$$\mathcal{L} = \mathcal{L}^{SRC} + \mathcal{L}^{REF} + \mathcal{L}^{SRC+REF} \tag{7.4}$$

5: Here, for each $\text{INPUT} \in \{\text{SRC, REF, SRC+REF}\}$, we define $\hat{\boldsymbol{y}}_{WL}^{INPUT} = [\hat{y}_1^{INPUT}, \ldots, \hat{y}_n^{INPUT}]$

Furthermore, in line with preceding metrics constructed upon the COMET framework, our models use features such as gradual unfreezing, and discriminative learning rates.

Inference time

▸ **ERROR SPAN PREDICTION.** For each subword in the translation, we average the output distribution of the word-level linear layer obtained for each forward pass. Using this distribution, we predict a set of word-level tags $\hat{\boldsymbol{y}}_{\text{WL}} = [\hat{y}_1, \ldots, \hat{y}_n]$ by taking the most likely class for each token. From these tags, we construct a list of *error spans*, S, by grouping adjacent subwords identified as errors. The severity of each span in S is defined according to the most severe error tag found within the span.

▸ **SENTENCE-LEVEL PREDICTION.** For each forward pass, we obtain the corresponding sentence-level scores: $\hat{y}_{\text{SRC}}$, $\hat{y}_{\text{REF}}$, and $\hat{y}_{\text{SRC+REF}}$.[6] Additionally, we leverage the information coming from the predicted list of error spans, S, to infer an automated MQM score. To do so, we follow the MQM framework: we obtain the error counts for each severity level— $c_{\text{MIN}}, c_{\text{MAJ}}, c_{\text{CRIT}}$ —and apply the predetermined severity penalty multipliers to define the error type penalty total, $e(S)$. Formally:

$$e(S) = c_{\text{MIN}} + 5 \times c_{\text{MAJ}} + 10 \times c_{\text{CRIT}}. \tag{7.5}$$

Finally, we obtain $\hat{y}_{\text{MQM}}$ by capping and flipping the sign of $e(S)$:

$$\hat{y}_{\text{MQM}} = \begin{cases} \frac{25 - e(S)}{25}, & \text{if } e(S) < 25. \\ 0, & \text{otherwise.} \end{cases} \tag{7.6}$$

Note that the predicted score $\hat{y}_{\text{MQM}}$ is bounded between 0 and 1, with a score of 1 corresponding to a perfect translation.

We aggregate the scores to compute the final sentence-level score, $\hat{y}_{\text{SL}}$, through a weighted sum of the different sentence-level scores. Importantly, we also include the inferred MQM score $\hat{y}_{\text{MQM}}$ to directly inform the final sentence-level prediction. Formally:

$$\hat{y}_{\text{SL}} = \boldsymbol{w}^\top \hat{\boldsymbol{y}} \tag{7.7}$$

where $\hat{\boldsymbol{y}} = [\hat{y}_{\text{SRC}}, \hat{y}_{\text{REF}}, \hat{y}_{\text{SRC+REF}}, \hat{y}_{\text{MQM}}]$, and $\boldsymbol{w}$ is set to $[1/9, 1/3, 1/3, 2/9]$.[7]

6: Here, for ease of notation, we use $\hat{y}_{\text{SRC}}$, $\hat{y}_{\text{REF}}$, and $\hat{y}_{\text{SRC+REF}}$ to represent the sentence-level score for each input type.

7: Under this paradigm, there is a connection to the work presented in Chapter 6 on SPECTRA: just as with rationalizers, we are directly using error span predictions (which can be used as explanations) to inform the final sentence-level prediction.

7.1.3 Corpora

Our models are exclusively trained on publicly available DA and MQM annotations, most of which have been collected by WMT over the recent years.

▸ **DA DATA.** We use DA annotations collected by WMT from 2017 to 2020, and the MLQE-PE dataset (Fomicheva et al., 2022). As the MLQE-PE dataset does not contain reference translations, we used the post-edit translations as reference translations. Overall, the corpus consists of around 1 million samples, spanning 36 language pairs.

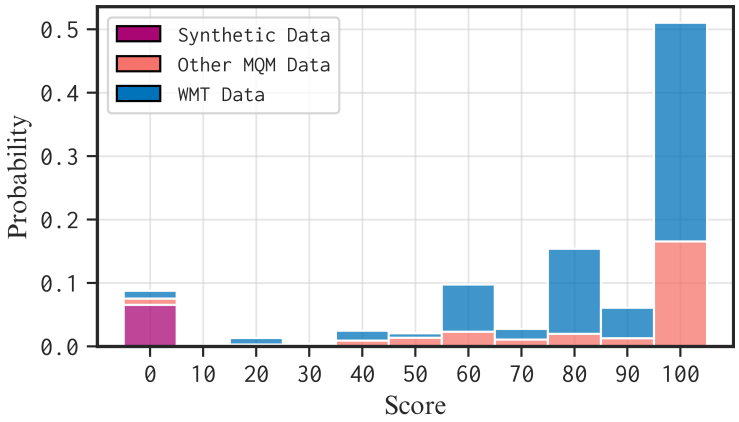


Figure 7.3: Histogram of the sentence-level scores from each partition of the MQM data. We aggregate the data from IndicMT, DEMETR under "Other MQM Data".

DATASET	No. Samples	Incorrect MT (%)	Error type dist.
WMT Data	147K (76%)	63%	[57, 42, 1]
IndicMT	7K (4%)	80%	[19, 52, 29]
DEMETR	22K (11%)	47%	[38, 19, 43]
MLQE-PE Hall.	1.7K (1%)	100%	All set to `CRIT`
Synthetic Hall.	16K (8%)	100%	All set to `CRIT`

Table 7.1: Number of samples, as well as error statistics (overall percentage of incorrect translations; rates of error type [`MIN`, `MAJ`, `CRIT`]), of each MQM data source used for training XCOMET.

▸ **MQM DATA.** We collected the MQM annotations from WMT from 2020 to 2022.[8] We also used annotations sourced from other MQM-annotated datasets: (i) IndicMT (Sai B et al., 2023), which contains MQM annotations spanning 5 Indian languages, and (ii) DEMETR (Karpinska et al., 2022), a diagnostic dataset with perturbations spanning semantic, syntactic, and morphological errors.

Corpora with MQM annotations are usually extremely unbalanced with critical errors being underrepresented (see stats for WMT in Figure 7.3 and Table 7.1). As a result, metrics might struggle to effectively detect translations with critical errors and hallucinations. (Amrhein and Sennrich, 2022b; Raunak et al., 2022b; Guerreiro et al., 2023a). As such, we augment the MQM corpus with hallucinations from the MLQE-PE corpus and *synthetic critical* errors. We create detached and oscillatory hallucinations (Raunak et al., 2021; Guerreiro et al., 2023a): (i) detached hallucinations, replacing the translation with a random sentence or an unrelated one semantically similar to the source sentence[9]; and (ii) oscillatory hallucinations, where we randomly sample a n-gram from the translation (with n in $\{2, 3, 4\}$) and repeat it between 1 and 10 times. We set the sentence-level scores of these hallucinations to 0. Overall, our MQM corpus consists of 194K samples across 14 language pairs.

▸ **SCALING OF SENTENCE-LEVEL SCORES.** While the sentence-level scores inferred from MQM annotations (through the procedure in Equation 7.6) are bounded between 0 and 1, DA annotations usually require z-normalization in order to mitigate variations in scoring strategies by different annotators (Bojar et al., 2017).[10] Thus, as z-scores are inherently centered at 0 and unbounded, there is a scaling mismatch between the data samples.

Consequently, to circumvent this limitation, we employ min-max scaling on our DA corpus to set its range of scores to $[0, 1]$. To do so, we set a practical minimum and maximum z-score value. We obtain the

8: Here, we exclude the 2022 News domain annotations, which we reserved for testing.

9: We measure cross-lingual similarity using sentence embeddings obtained with the LaBSE encoder (Feng et al., 2022).

10: This is particularly relevant for DA annotations, since these judgements typically come from non-expert annotators.

minimum score by averaging the z-scores for translations with over 1 annotation, wherein all annotators unanimously scored them with an unnormalized 0 DA score, i.e., they deemed the translation as "random". For determining a maximum value, we applied the same process for perfect translations, i.e., unnormalized 100 DA score.[11]

11: This was initially introduced in BLEURT-20 (Pu et al., 2021).

7.1.4 Training curriculum

xCOMET models undergo a 3-phase curriculum training. Throughout these phases, the training emphasis alternates between sentence-level prediction and error span prediction by tweaking the parameter λ in Equation 7.3. The curriculum phases can be described as follows:

Phase I: The model is trained exclusively using the DA data. In this phase, the focus is exclusively set on sentence-level regression.
Phase II: In this stage, we introduce word-level supervision. To achieve this, the model is fine-tuned on our diverse MQM corpus, with most emphasis placed on the word-level task.
Phase III: The last training phase is aimed at unifying both tasks. The model is further fine-tuned using high-quality MQM data from (Freitag et al., 2021a), with a bigger emphasis set to sentence-level prediction.[12]

12: The achieved λ weights for Phases II and III were $\lambda = 0.983$ and $\lambda = 0.055$, respectively.

▶ **INTERPRETATION OF THE CURRICULUM.** We start by training a sentence-level metric — similar to UNITE (Wan et al., 2022a) — on the vastly available DA annotations. Phase I acts as a warm-up for subsequent stages. In fact, prior research has shown that models trained on DA annotations leverage token-level information that aligns with MQM error annotations (Rei et al., 2023c). Moving to Phase II, we assume we have a metric that can perform sentence-level regression. Thus, the aim here shifts to integrating word-level supervision without compromising the previously acquired sentence-level prediction skills. To do so, we use the highly diverse corpora of MQM annotations and set most emphasis on the word-level task. Finally, we exclusively leverage a small corpus (around 25k samples) of very high-quality MQM annotations from (Freitag et al., 2021a) — each sample has three annotations from separate annotators — with additional synthetic hallucinations. Our focus here is to mitigate any potential decline in sentence-level regression capabilities during Phase II.

7.2 Experimental Setting

7.2.1 Evaluation

We evaluate the performance of our metrics using two datasets: (i) the MQM annotations from the News domain of the WMT 2022 Metrics shared task, and (ii) the WMT 2023 Metrics shared task evaluation suite. The WMT22 annotations encompass three language pairs: Chinese→English (zh→en), English→German (en→de), and English→Russian (en→ru). On the other hand, the WMT23 annotations cover Chinese→English (zh→en), English→German (en→de), and Hebrew→English (he→en). We evaluate the metrics in terms of sentence-level, system-level, and error span prediction performance.

Table 7.2: Segment-level Kendall-Tau (↑) in (a), and System-level Pairwise Accuracy (↑) in (b) using the Perm-Both hypothesis test (Deutsch et al., 2021) on the WMT22 Shared Task News domain test set. Numbers in bold belong to the top-performing cluster according to statistical significance ($p < 0.05$).

METRIC	zh→en	en→de	en→ru	Avg.
Baselines				
BLEURT-20	0.336	0.380	0.379	0.365
COMET-22	0.335	0.369	0.391	0.361
METRICX	**0.415**	0.405	0.444	0.421
GEMBA-GPT4-DA	0.292	0.387	0.354	0.354
XCOMET				
XCOMET-XL	0.399	0.414	0.448	0.421
XCOMET-XXL	0.390	**0.435**	**0.470**	**0.432**
MQM scores from the error spans ($\hat{y} = \hat{y}_{\text{MQM}}$)				
XCOMET-XL (MQM)	0.374	0.389	0.445	0.402
XCOMET-XXL (MQM)	0.332	0.415	0.439	0.395

(a) Sentence-level evaluation.

METRIC	zh→en	en→de	en→ru	Avg.
Baselines				
BLEURT-20	0.762	0.771	0.743	0.759
COMET-22	0.705	0.800	0.733	0.746
METRICX	0.762	0.781	0.724	0.756
GEMBA-GPT4-DA	0.752	**0.848**	**0.876**	**0.825**
XCOMET				
XCOMET-XL	**0.800**	0.743	0.790	0.778
XCOMET-XXL	**0.800**	**0.829**	**0.829**	**0.819**
MQM scores from the error spans ($\hat{y} = \hat{y}_{\text{MQM}}$)				
XCOMET-XL (MQM)	0.781	0.762	0.762	0.768
XCOMET-XXL (MQM)	0.781	**0.838**	0.810	0.810

(b) System-level Evaluation.

At the sentence-level, we report Kendall's Tau (τ) using the Perm-Both hypothesis test (Deutsch et al., 2021). We also evaluate the metrics on System-level Pairwise Accuracy (Kocmi et al., 2021a). We base these evaluations on 200 re-sampling runs, with a significance level (p) set to 0.05. For error span prediction, we adopt the WMT23 Quality Estimation shared task evaluation methodology and compute F1 scores calculated at the character level, taking into account partial matches for both minor and major errors.[13] For WMT23, we follow the evaluation setup from the shared task (Freitag et al., 2023c) and report the aggregated System-level Pairwise Accuracy pooled across all language pairs, and the primary metric Average Correlation, which encompasses ten tasks, spanning system- and sentence-level metrics.[14]

13: We convert all critical errors into major errors, in order to match the guidelines described in (Freitag et al., 2021a), that were used for annotating the zh→en and de→en test sets.

14: The ten tasks consist of the System-level Pairwise Accuracy pooled across the language pairs, as well as segment-level pairwise ranking accuracy with tie calibration (Deutsch et al., 2023a), and system- and segment-level Pearson correlation for each of the individual language pairs.

7.2.2 Baselines

- **SENTENCE AND SYSTEM-LEVEL.** We test our metrics against widely used *open* neural metrics: COMET-22 (Rei et al., 2022a) and BLEURT-20 (Pu et al., 2021). Additionally, we include METRICX, the best performing metric from the WMT22 Metrics shared task (Freitag et al., 2022c)[15], and GEMBA (Kocmi and Federmann, 2023b), which employs GPT4 (OpenAI, 2023) to evaluate translations following DA guidelines. For WMT23, we report the same metrics but update them, when needed (for METRICX-23 (Juraska et al., 2023) and GEMBA-MQM (Kocmi and Federmann, 2023a)), with the versions submitted to the official competition.[16]

- **ERROR SPAN PREDICTION.** We report results using GPT3.5 and GPT4 models, by prompting it in the style of AUTOMQM (Fernandes et al., 2023a).[17] We carefully select 5 shots that are held constant for all samples. This way, we can directly compare our results with state-of-the-art LLMs, which have been shown to be able to perform the task of error detection (Fernandes et al., 2023a; Xu et al., 2023c).

15: Specifically, we employ the `metricx_xxl_MQM_2020` submission scores from the `mt-metrics-eval` package. Although the metric has not been released publicly, it is public that it is built upon the mT5-XXL (Xue et al., 2021) and has 13B parameters (Deutsch et al., 2023b).

16: For all baselines, we report the official numbers from the WMT23 Metrics Shared Task (Freitag et al., 2023c).

17: We use the models from the OpenAI API (`gpt-3.5-turbo` and `gpt-4`) in October, 2023.

Table 7.3: System-level Pairwise Accuracy (↑) (Kocmi et al., 2021a) computed over data pooled across all three WMT23 language pairs, and primary metric Average Correlation (↑).

METRIC	System-level Accuracy	Avg. Correlation
Baselines		
BLEURT-20	0.892	0.776
COMET-22	0.900	0.779
METRICX-23	0.908	0.808
GEMBA-MQM	**0.944**	0.802
XCOMET		
XCOMET-XL	0.912	**0.813**
XCOMET-XXL	0.920	**0.812**

7.3 Correlations with Human Judgements

In this section, we present a standard performance analysis of our metrics in terms of correlations with human judgments. Overall, we find XCOMET to be a state-of-the-art in sentence-level and error span prediction, being competitive with generative LLMs in terms of system-level evaluation.

- **SENTENCE-LEVEL EVALUATION.** Table 7.2a shows that both XCOMET metrics outperform other strong performing neural metrics, including the generative approach leveraging GPT4 of GEMBA. In particular, XCOMET-XXL sets a new state-of-the-art for en→de and en→ru. Interestingly, we can see that, while scaling up the encoder model of the XCOMET metrics (from XL to XXL) holds better results, XCOMET-XL is very competitive. In fact, it outperforms METRICX, which runs at even a larger size than XCOMET-XXL. Finally, we can also observe that the MQM scores inferred exclusively from the predicted error spans also exhibit strong performance, outperforming widely used metrics BLEURT-20 and COMET-22. This is particularly relevant: the predicted error spans bring not only a more detailed view into translation errors but also provide high-quality sentence-level scores.

- **SYSTEM-LEVEL EVALUATION.** Table 7.2b and Table 7.3 show results for system-level for both WMT22 and WMT23 test sets. Similarly to what we observed at the sentence-level, our metrics show consistently superior performance when compared to other dedicated neural metrics. Notably, although generative approaches typically do much better at system-level evaluation when compared to dedicated models (Fernandes et al., 2023a; Kocmi and Federmann, 2023b), XCOMET-XXL remains competitive in all language pairs with GEMBA using GPT4. Finally, building on the findings at the sentence-level, Table 7.2b reveals that the MQM scores inferred directly and exclusively from the predicted error spans also exhibit very competitive performance in terms of system-level accuracy.

- **AGGREGATED EVALUATION.** Table 7.3 shows aggregated results for the WMT23 Metrics Shared Task. Our metrics, at both scales, would win the shared task, outperforming both METRICX and GEMBA-MQM. Following the trend presented at sentence-level evaluation, we note that XCOMET-XL is indeed competitive with XCOMET-XXL although running at a smaller scale.

- **ERROR SPAN PREDICTION.** While we have highlighted the utility of the

METRIC	zh→en	en→de	en→ru	Avg.
Baselines				
• AutoMQM (GPT3.5)	0.143	0.160	0.166	0.156
• AutoMQM (GPT4)	0.248	0.257	**0.281**	0.262
XCOMET				
• XCOMET-XL	0.237	0.290	**0.281**	0.269
• XCOMET-XXL	**0.257**	**0.320**	0.262	**0.280**
Error spans detected with source-only input				
• XCOMET-XL (SRC)	0.208	0.264	0.252	0.242
• XCOMET-XXL (SRC)	0.229	0.298	0.238	0.255

Table 7.4: F1 scores (↑) for error span detection: reference-free (•), reference-based (•) evaluation.

SCORE	zh→en	en→de	en→ru	All
$\hat{y}_{\text{SRC}}$	0.73	0.75	0.79	0.78
$\hat{y}_{\text{REF}}$	0.75	0.74	0.75	0.77
$\hat{y}_{\text{SRC+REF}}$	0.78	0.79	0.82	0.82
$\hat{y}_{\text{SL}}{}^{\dagger}$	0.90	0.92	0.92	0.92

Table 7.5: Pearson correlations between the regression scores produced by **XCOMET-XXL** ($\hat{y}_{\text{SRC}}$, $\hat{y}_{\text{REF}}$, $\hat{y}_{\text{SRC+REF}}$, $\hat{y}_{\text{SL}}$) and the MQM inferred score, $\hat{y}_{\text{MQM}}$, computed from the identified error spans. †The computation of $\hat{y}_{\text{SL}}$, contrary to the other scores, makes direct use of $\hat{y}_{\text{MQM}}$ (see Eq. 7.7).

predicted error spans through the inferred sentence-level MQM scores, here we turn to evaluating them directly. Table 7.4 shows that the error spans predicted via XCOMET metrics outperform those obtained with both GPT3.5 and GPT4 despite being smaller in capacity relative to these models. In fact, our metrics achieve close performance to that of GPT4, even when a reference is not provided.

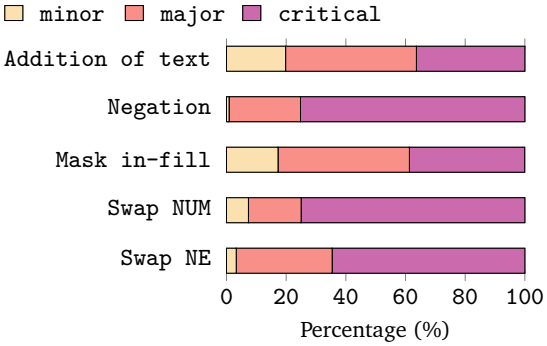
▸ **INTERPLAY OF ERROR SPANS AND SENTENCE-LEVEL SCORES.** Table 7.5 shows a strong correlation between the different score types predicted by XCOMET and the MQM inferred score derived exclusively from error spans. This interplay is highly important: the predicted error spans may be valuable, not just for the sake of accuracy but also for interpretability. Interestingly, these high correlations with the predicted scores from each forward pass ($\hat{y}_{\text{SRC}}$, $\hat{y}_{\text{REF}}$, $\hat{y}_{\text{SRC+REF}}$) are obtained despite no explicit alignment mechanism governing the relationship between the predictions of the sentence-level and word-level heads. We hypothesize that it is thus the shared encoder that, during the multi-task training, aligns the representations between the two tasks. As such, XCOMET provides, through its predicted error spans, a potential lens through which we can better understand, contextualize, and even debug its own sentence-level predictions.

7.4 Robustness of XCOMET to Pathological Translations

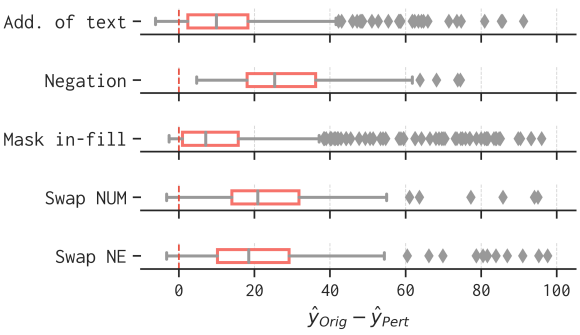
We have shown that XCOMET metrics exhibit state-of-the-art correlations with human judgements when evaluating on high-quality MQM annotations. However, these MQM annotations are often highly unbalanced and contain little to no major or critical errors. As such, they may not offer a full picture of the metrics' performance. In this section, we shift our focus to studying how XCOMET metrics behave when evaluating translations with localized major or critical errors, and highly pathological translations, such as hallucinations.

Table 7.6: Percentage (%) of translations, segmented by perturbation type, that are predicted to have no errors (↓). We show results for both zh→en and he→en language pairs across xCOMET sizes.

	zh→en		he→en	
ERROR	XL	XXL	XL	XXL
Addition of text	3.66	10.7	6.15	7.35
Negation	0.20	0.20	3.89	4.90
Mask in-fill	5.01	17.0	4.78	3.92
Swap NUM	3.19	2.88	0.16	0.00
Swap NE	3.66	6.94	9.81	7.01
All	**2.24**	10.7	9.81	**7.00**



(a) Percentage of error types on data with critical errors (for both zh→en and he→en data), as predicted by **XCOMET-XXL**.



(b) Impact of the perturbations, as measured by the difference in **XCOMET-XXL** ($\hat{y} = \hat{y}_{SL}$) between the original and the perturbed translation, on the zh→en data.

Figure 7.4: Analysis of xCOMET-XXL for data with localized critical errors in terms of (a) distribution of error severities for the predicted error spans, and (b) sensitivity of the sentence-level scores.

7.4.1 Localized errors

We employ SMAUG (Alves et al., 2022)[18], a tool designed to generate synthetic data for stress-testing metrics, to create corrupted translations that contain major or critical errors. We generate translations with the following pathologies: addition of text, negation errors, mask in-filling, named entity errors, and errors in numbers. For this evaluation, we use data from the WMT 2023 Metrics shared task. Specifically, we corrupt the released *synthetic* references for which the xCOMET metrics found no errors.[19] Moreover, as the full suite of SMAUG transformations can only be applied to English text, we focus on Chinese→English (zh→en) and Hebrew→English (he→en) translations.

18: https://github.com/Unbabel/smaug

19: This allows us to isolate the effect of the perturbations. In case there are predicted error spans for the transformed translations, these are a result of the perturbation induced.

▶ **XCOMET PREDICTS MOST LOCALIZED ERRORS AS MAJOR OR CRITICAL ERRORS.** Table 7.6 shows that xCOMET metrics identify errors in the vast majority of the perturbed samples, with trends varying across scale and language pair. We found that the errors predicted by xCOMET-XL and xCOMET-XXL overlap with the artificially induced perturbations in over 90% of the perturbed samples (98% for XL and 90.9% for XXL). However, upon further analysis of xCOMET's predicted error spans, we observed that the model tends to identify additional spans in the perturbed sentence as erroneous, beyond the induced perturbations. This behavior is more prominent for perturbations involving the addition of text, which are not as localized as perturbations like swapping numbers or named entities. Furthermore, we noticed that the model has a propensity to assign the same error category to all predicted spans within a single sentence. When the metric predicts multiple error spans

Table 7.7: Predictions of xCOMET-XL for perturbed translations. We highlight minor (MIN), major (MAJ) and critical (CRIT) error spans.

Example of a **Negation error**	
Source	最后，我们设定了两个大的战略方向。
Translation	In the end, we set two major strategic directions.
Perturbed MT	In the end, we PERT: don't even see two major strategic directions.
xCOMET Error Spans	In the end, CRIT: we don't even see two major strategic directions.
Example of a **Swap NUM error**	
Source	海地国内的一些地区，多达90%的房屋在地震中被摧毁。
Translation	In some areas of Haiti, as many as 90% of houses were destroyed in the earthquake.
Perturbed MT	In some areas of Haiti, as many as PERT: 37.5% of houses were destroyed in the earthquake.
xCOMET Error Spans	In some areas of Haiti, as many as MAJ: 37.5% of houses were destroyed in the earthquake.
Example of a **Mask in-fill error**	
Source	1993 年，莫德罗在 1989 年 5 月的市政选举中被判选举舞弊罪名成立，但并未入狱。
Reference	In 1993, Modrow was convicted of electoral fraud in the May 1989 municipal elections, but did not go to prison.
Perturbed MT	In 1993, Modrow was convicted of electoral fraud in the PERT: May 1989 presidential election, but did not go to prison.
xCOMET Error Spans	In 1993, Modrow was convicted of electoral fraud in the May 1989 MAJ: presidential election, but did not go to prison.

in a sentence, it assigns different severity levels to those spans only about 35% of the time. Improving the model's ability to differentiate error categories among multiple errors within a sentence is an interesting avenue for future research and development. We show several examples of predictions of xCOMET in Table 7.7.

We also found that negation errors and mismatches in numbers are the most easily identified by the metrics. This is interesting: localized errors, such as mismatches in numbers and named-entity errors, had been pinpointed as weaknesses of previous COMET metrics (Amrhein and Sennrich, 2022b; Raunak et al., 2022b). This earlier limitation seems to now have been addressed successfully. In fact, the results in Figure 7.4a show that most of these errors are predicted as critical errors. One plausible hypothesis for these improvements is the incorporation of datasets that contain negative translations and synthetic hallucinations into our training set.

▶ **XCOMET SENTENCE-LEVEL SCORES ARE SENSITIVE TO LOCALIZED PERTURBATIONS.** Figure 7.4b shows that localized errors can lead to significant decreases in the predicted sentence-level scores, with perturbation-wise trends mirroring those of the error span predictions: the most pronounced decreases are found for negation errors and mismatches in numbers and named-entities (median decreases of around 20 points). The distribution of the decreases in quality also reveals two relevant trends: (i) localized perturbations can cause xCOMET-XXL to shift from a score of a perfect translation to that of an unrelated translation, and (ii) the behavior of xCOMET-XXL is not perfect and can be further improved: in rare cases, perturbations may actually lead to an increase in the score. Nevertheless, upon closer inspection, for over 90% of cases, the increase is smaller than 1 point.

7.4.2 Hallucinations

Hallucinations lie at the extreme-end of machine translation pathologies (Raunak et al., 2021), and can have devastating impact when

Table 7.8: Hallucination detection performance on the de→en hallucination benchmark from Guerreiro et al. (2023a) as measured by AUROC (↑) for reference-free (●) and reference-based (●) metrics. We report results for all the dataset, for fully detached, and oscillatory hallucinations separately.

METRIC	All	Full Det.	Osc.
Baselines			
● BLEURT-20	0.824	0.892	0.799
● COMET-22	0.829	0.878	0.883
● COMETKIWI-XXL	0.839	0.834	0.902
XCOMET			
● XCOMET-XL	0.865	0.907	0.922
● XCOMET-XXL	0.890	**0.964**	0.844
QE scores from the error spans ($\hat{y} = \hat{y}_{SRC}$)			
● XCOMET-XL (SRC)	0.885	0.924	**0.944**
● XCOMET-XXL (SRC)	**0.902**	0.959	0.866

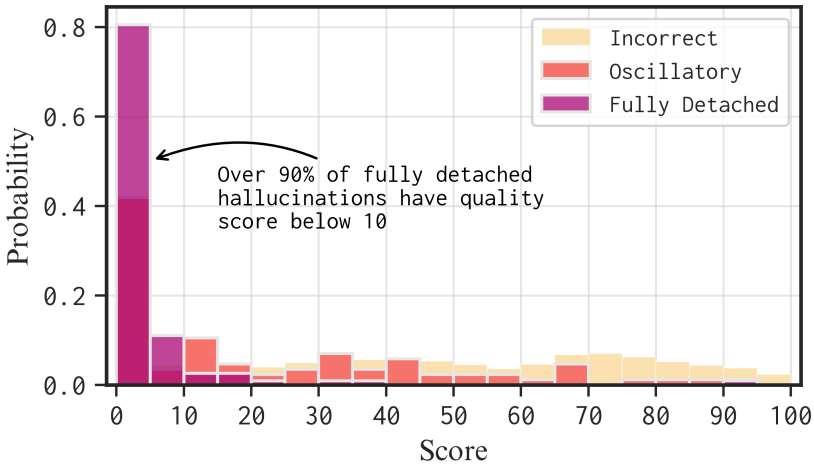


Figure 7.5: Category-wise distribution of **XCOMET-XXL** scores on the hallucination benchmark.

This is the work presented in Chapter Chapter 3.

models are deployed *in the wild*. Yet, these translations are often overlooked when assessing the performance of translation systems. Their rarity means that performance, usually judged according to an aggregated corpus-level score, may remain largely unperturbed by a very small number of hallucinations. Here, we assess how the XCOMET metrics rank hallucinations among other translations. We will use the German→English hallucination benchmark introduced in Guerreiro et al. (2023a). This benchmark involves over 3.4k translations — produced by an actual machine translation system — of different error types, including omissions, named-entity errors, and hallucinations (oscillatory, fully, and strongly detached). For a metric that has not been trained explicitly to rank translations, the benchmark is quite challenging: hallucinations should be ranked below other severe errors and incorrect translations.

▶ **XCOMET METRICS CAN DISTINGUISH HALLUCINATIONS FROM OTHER TRANSLATIONS.** The results in Table 7.8 show that both XCOMET metrics largely rank hallucinations lower than other errors. This is especially true for the most severe type of hallucination (fully detached), for which the AUROC exceeds 95 for the XXL metric. In fact, Figure 7.5 reveals that XCOMET-XXL assigns over 90% of these fully detached hallucinations a score under 10. We show examples of error spans predicted by XCOMET-XXL in Table 7.9. Relative to previous metrics, XCOMET achieves overall improvements. Interestingly, we also find that SRC-based evaluation (i.e., without the use of a reference) can reap benefits in this scenario. We hypothesize that this is due to the

Table 7.9: Examples of predictions of XCOMET-XXL for the hallucination data of Guerreiro et al. (2023a). The model correctly identifies the anomalous repeated phrase for the oscillatory hallucination, and predicts the whole translation as a single error span in the case of the fully detached hallucination.

Example of error spans for an **oscillatory hallucination**	
Source	Das Teilabonnement für international tätige Juristen.
Translation	The sub-sub-sub-sub-scription for international lawyers.
Reference	Partial subscriptions for internationally active lawyers.
XCOMET Error Spans	The `CRIT:` sub-sub-sub-sub-scription for international lawyers.

Example of error spans for a **detached hallucination**	
Source	Empfehlenswert gleich mit der Zimmerreservierung zu buchen!
Translation	The staff were very friendly and helpful. The room was clean and comfortable.
Reference	We recommend booking your treatments together with the hotel booking!
XCOMET Error Spans	`CRIT:` The staff were very friendly and helpful. The room was clean and comfortable.

metric over-relying on the reference when it is available (Rei et al., 2023c). While hallucinations contain content that is detached from the source, some of their text may still overlap (even if just lexically) the reference (e.g., in strongly detached or oscillatory hallucinations), leading to higher scores.

7.5 Ablations on Design Choices

We now address relevant questions about the development of XCOMET through ablations on design choices. Ablations are run with XCOMET-XL on the MQM annotations from the News domain of the WMT 22 Metrics shared task (see Section 7.2).

▸ **IMPACT OF THE TRAINING CURRICULUM.** We employed a curriculum to train XCOMET (see Section 7.1.4) in order to balance data from different annotation strategies (i.e., DA and MQM annotations), and also to better balance the multi-task objective. Here, we want to assess how performance evolves throughout the different stages. We perform ablations on Phase II and Phase III, which correspond to the introduction of the multi-task objective and MQM training data that contain both sentence-level scores and error spans.

Table 7.10 shows that while a multi-task model outperforms single-task models for sentence-level evaluation[20], it does not hold true for word-level evaluation. The best word-level model is obtained by doing Phase II with a word-level only objective. Note that we can still extract sentence-level scores from such a model in two ways: (i) by leveraging the still existing regression head trained during Phase I, or (ii) by converting the error spans into a single sentence-level score. However, notably, neither of these approaches is competitive with our final XCOMET model. In fact, it turns out, performing sentence-level evaluation via the error spans predicted by the final model leads to better correlations than with the word-level only model. Moreover, aggregating the different scores from the regression heads yields the best overall performance for sentence-level evaluation.

20: For sentence-level only models, we present the sentence-level score $\hat{y} = \hat{y}_{\text{REG}}$ correspondent to setting uniform weights across all three individual scores (SRC, REF, and SRC+REF).

▸ **IMPACT OF THE WEIGHTS ON THE SENTENCE-LEVEL SCORES FROM EQUATION (7.7).** We studied the impact of aggregating different sentence-level scores by varying the weights $\boldsymbol{w}$ in Equation 7.7. Besides the individual scores for each evaluation mode (SRC, REF, and

Table 7.10: Segment-level Kendall-Tau (τ) (↑) and F1 scores (↑) for error span detection on different curriculum choices. We represent metrics that can *only* perform segment-level evaluation, *only* perform word-level evaluation ($\lambda = 1$), and both. When a model has no capabilities to perform error span prediction, we write NA under its F1 score.

	zh→en		en→de		en→ru		Avg.	
STAGE	τ	F1	τ	F1	τ	F1	τ	F1
*Post Phase I with **sentence-level only** objective: $\lambda = 0$*								
PHASE **I**	0.377	NA	0.356	NA	0.425	NA	0.386	NA
PHASE **II** ($\hat{y} = \hat{y}_{\text{REG}}$)	0.372	NA	0.386	NA	0.448	NA	0.402	NA
PHASE **III** ($\hat{y} = \hat{y}_{\text{REG}}$)	0.395	NA	0.391	NA	0.457	NA	0.414	NA
*Post Phase I with **word-level only** objective: $\lambda = 1$*								
PHASE **II** ($\hat{y} = \hat{y}_{\text{REG}}$)	0.333	0.293	0.357	0.332	0.415	0.229	0.368	0.285
PHASE **II** ($\hat{y} = \hat{y}_{\text{MQM}}$)	0.331	=	0.410	=	0.395	=	0.379	=
*Post Phase I with **multi-task only** objective: λ set as described in Section 7.1.4*								
PHASE **II** ($\hat{y} = \hat{y}_{\text{MQM}}$)	0.330	0.284	0.359	0.328	0.413	0.212	0.367	0.275
PHASE **II** ($\hat{y} = \hat{y}_{\text{SL}}$)	0.368	=	0.396	=	0.420	=	0.395	=
PHASE **III** ($\hat{y} = \hat{y}_{\text{MQM}}$; XCOMET)	0.374	0.237	0.389	0.290	0.445	0.281	0.402	0.269
PHASE **III** ($\hat{y} = \hat{y}_{\text{SL}}$; XCOMET)	0.399	=	0.597	=	0.448	=	0.421	=

Table 7.11: Segment-level Kendall-Tau (τ) (↑) for individual and aggregated sentence-level scores.

SCORE	zh→en	en→de	en→ru	All
Individual Scores				
$\hat{y}_{\text{SRC}}$	0.368	0.358	0.402	0.376
$\hat{y}_{\text{REF}}$	0.399	0.389	0.427	0.405
$\hat{y}_{\text{SRC+REF}}$	0.399	0.390	0.438	**0.409**
$\hat{y}_{\text{MQM}}$	0.374	0.389	0.445	0.402
Aggregated Regression Scores: see Equation 7.7				
$\hat{y}_{\text{REG}}$	0.402	0.380	0.4401	0.408
$\hat{y}_{\text{UNIF}}$	0.398	0.402	0.448	0.416
$\hat{y}_{\text{SL}}$	0.399	0.414	0.448	**0.421**

SRC+REF), we present three aggregations: (i) $\hat{y} = \hat{y}_{\text{SL}}$ used in the final model, (ii) $\hat{y} = \hat{y}_{\text{REG}}$ with uniform weights across the three individual scores and not considering the inferred MQM score from the error spans ($\boldsymbol{w} = [1/3, 1/3, 1/3, 0]$), and (iii) $\hat{y} = \hat{y}_{\text{UNIF}}$ with uniform weights across all scores ($\boldsymbol{w} = [1/4, 1/4, 1/4, 1/4]$).

Table 7.11 reveals two interesting findings: (i) aggregating scores does not always outperform individual scores (e.g., $\hat{y}_{\text{REG}}$ performs similarly to $\hat{y}_{\text{SRC+REF}}$), and (ii) including an inferred MQM score obtained through error span prediction boosts sentence-level performance. Notably, the improvement of our final aggregated score over $\hat{y}_{\text{SRC+REF}}$ is not substantial. This suggests that, under computational constraints, one could consider computing a single $\hat{y}_{\text{SRC+REF}}$ score without the need for three different forward passes and error span prediction.

7.6 Conclusions

We introduced XCOMET: a novel suite of metrics for machine translation evaluation that combines sentence-level prediction with fine-grained error span prediction. Through extensive experiments, we have shown that XCOMET is a state-of-the-art metric at all relevant vectors of evaluation: sentence-level, system-level, and error span prediction. Notably, through XCOMET's capabilities to predict error spans, we can not only obtain useful signals for downstream prediction (either directly through error span prediction or by informing sentence-level scores) but also

gain access to a lens through which we can better understand and interpret its predictions. We also stress-tested the metrics by assessing how they score localized critical errors and hallucinations: the metrics identify the vast majority of localized errors and can appropriately penalize the severity of hallucinations.

XCOMET has had significant impact in the community since its release. Most notably, it achieved top performance as the winning system in both the 2023 and 2024 WMT Metrics Shared Task (Freitag et al., 2023c, 2024), delivering state-of-the-art performance more than one year after its release. The success of XCOMET has inspired several derivative works and applications. For instance, Anugraha et al. (2024) developed an ensemble combining XCOMET with MetricX that also achieved top performance in the WMT 2024 Metrics Shared Task (Freitag et al., 2024). In parallel, Larionov et al. (2024) have focused on creating efficient alternatives to XCOMET through the application of distillation, quantization, and pruning techniques. Concurrently, with the emergence of techniques such as preference optimization for alignment of language models, researchers have found new applications for XCOMET in model training. It has been employed to provide token-level rewards for training translation models (Ramos et al., 2024) and to create high-quality preference datasets (Agrawal et al., 2024b). In the analysis conducted in Agrawal et al. (2024b), an ensemble of XCOMET-XL and XCOMET-XXL achieved the highest correlation with human judgments and demonstrates high precision in identifying preferred translations. Furthermore, XCOMET's capabilities have been extended through the development of XTOWER, a language model specifically designed to produce high-quality explanations for translation errors (as predicted by XCOMET) and to leverage these explanations for translation improvements.[21]

21: We will briefly touch on this contribution in Chapter 8.

Thus far, this thesis has addressed two key aspects of machine translation research: safeguarding against critical failure modes (Chapter 3 to Chapter 5) and improving translation evaluation methodologies (Chapter 6 and Chapter 7). We first investigated methods to detect, understand, and mitigate hallucinations in translation models. We then turned to the development of more interpretable and robust evaluation metrics, culminating in XCOMET, which combines error span detection with overall quality assessment.

As we developed XCOMET, generative large language models began to emerge as a promising solution for improving neural machine translation even further. Building upon the insights gained from our work on robustness and quality metrics, we now turn to the final part of this thesis: the development of TOWER, a suite of multilingual large language models of up to 70 billion parameters optimized for translation-related tasks that has been shown to outperform state-of-the-art open-source models, as well as strong commercial systems across a wide range of language pairs.

Part IV

Advancing Capabilities

Highlights and Structure

LARGE LANGUAGE MODELS FOR TRANSLATION-RELATED TASKS

▸ **HIGHLIGHTS AND CONTRIBUTIONS** In the final part of our thesis, we present TOWER, a suite of multilingual large language models optimized for translation-related tasks.These models are developed through a two-step approach: first, we create a base model by continually pretraining an existing high-quality general-purpose LLM; second, we perform supervised fine-tuning using a curated dataset that includes translation data, other translation-related tasks, and instruction-following datasets.

We train TOWER models in two sizes—7 billion and 70 billion parameters—supporting 15 languages. We will focus specifically on their performance in machine translation, where our models not only surpass state-of-the-art open translation systems but also outperform closed commercial systems and general-purpose LLMs such as GPT-4 and Claude-3.5-Sonnet.

Furthermore, we leverage recent advancements in automatic metrics for translation evaluation, such as those presented in Chapter 7 on XCOMET, to further enhance translation quality. This approach culminated in our winning submission (according to automatic evaluation) for the WMT2024 General MT Shared Task.

▸ **PUBLICATIONS** The ideas and work presented in the following chapters were first published in:[22]

22: ⋆: joint first-authorship.

- Ricardo Rei⋆, José Pombal⋆, **Nuno M. Guerreiro⋆**, João Alves⋆, Pedro H. Martins⋆, Patrick Fernandes, Helena Wu, Tânia Vaz, Duarte M. Alves, Amin Farajian, Sweta Agrawal, António Farinhas, José G.C. de Souza, André F.T. Martins. TOWER-V2: *Unbabel-IST 2024 Submission for the General MT Shared Task.* Under submission at WMT 2024.
- Duarte M. Alves⋆, José Pombal⋆, **Nuno M. Guerreiro⋆**, Pedro H. Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G.C. de Souza, André F.T. Martins. *Tower: An Open Multilingual Large Language Model for Translation-Related Tasks*. In Proc. COLM 2024.

The TOWER Project: Large Language Models for Translation-related Tasks

8

Large Language Models (LLMs) are making strides towards becoming the *de facto* solution for multilingual machine translation (MMT). Many works have shown that it is possible to adapt LLMs for translation and achieve state-of-the-art results (Alves et al., 2023; Reinauer et al., 2023; Wei et al., 2023b; Zhang et al., 2023; Zhu et al., 2024).

One such example is the work on TOWER (Alves et al., 2024), which demonstrates that open NMT models like NLLB200 can be outperformed by adapting an LLM to translation. Specifically, TOWER is obtained via continual pretraining (Gupta et al., 2023; Xu et al., 2024a; Yıldız et al., 2024) of LLaMA-2 (Touvron et al., 2023a) on both monolingual and parallel data, followed by fine-tuning on high-quality instructions covering several MT-related tasks. This approach requires much less parallel training data than traditional neural machine translation methods and preserves the general capabilities of the LLM to respond to various prompts and solve diverse tasks.[1]

For the WMT24 General Translation task (Kocmi et al., 2024b), we enhance TOWER by significantly improving its training data, by extending its language support to 15 languages, including low-resource ones like Icelandic, and by scaling the underlying model to 70 billion parameters. Furthermore, because the WMT24 General Translation task focuses on paragraph-level translation instead of sentence-level, we also experiment with full-document translation and longer contexts, where TOWER originally struggled. These key improvements result in TOWER-V2 7B and 70B.[2]

For our primary submission, we combine TOWER-V2 70B with Quality-Aware Decoding (QAD) strategies (Fernandes et al., 2022b), such as Minimum Bayes Risk decoding (MBR) and Tuned Reranking (TRR). These techniques use reward models—translation quality estimators, in fact—during inference to select the best candidate from a set of generated samples, enhancing the overall output quality.

We report our results, including the human evaluation and final submission, in Section 8.4. TOWER-V2 — even at 7B parameters — exhibits state-of-the-art performance, challenging or outperforming commercial and closed systems like GPT-4, CLAUDE-SONNET-3.5, and DEEPL across a big range of language pairs.

Our contributions are:

- We show that expanding from 10 to 15 languages not only significantly improves quality for the newly added languages (as expected), but also maintains the quality of translations for the initial 10.
- We show significant improvements in paragraph- and document-level translation capabilities over previous TOWER versions more focused on sentence-level translation.

1: The first iteration of TOWER, TOWER-V1, was developed in the context of my PhD studies, but in this chapter I choose to focus specifically on our latest and best iteration of the model, TOWER-V2. A more in-depth introduction on TOWER-V1 is provided later on in this chapter.

2: The previous iteration, TOWER-V1, came in 7B and 13B parameters and was built on top of LLaMA-2 models. We will describe this suite in Section 8.1.

- We demonstrate that scaling the model from 7 to 70B parameters yields improvements, indicating that increased capacity benefits not only general LLM abilities but also task-specific performance.
- We analyze the impact of QAD on larger models than those studied by Fernandes et al. (2022b), showing that MBR decoding outperforms TRR according to both automatic metrics and human evaluation.

8.1 Background

▸ **THE TOWER-V1 SUITE OF MODELS.** TOWER (Alves et al., 2024) is a suite of multilingual large language models optimized for translation-related tasks. These models support 10 languages[3] are created through a 2-step recipe: (i) obtain TOWERBASE via continual pretraining of an existing high-quality general purpose LLM (in the original paper, we used LLaMA-2) on 20 billion tokens comprising of monolingual and parallel data;[4] and follow it with (ii) supervised fine-tuning on a dataset (we name it TOWERBLOCKS) that consists of publicly available data for high-quality instruction-following as well as translation-related tasks (e.g., automatic MT evaluation, automatic post edition, sentence/document-level translation). At the time of the release, TOWER-V1 models were state-of-the-art for translation in the supported languages, achieving competitive performance with GPT-3.5 and GPT-4.

3: English, German, French, Dutch, Portuguese, Italian, Spanish, Korean, Chinese, Russian.

4: The TOWERBASE models from the first suite of TOWER models were created using a mix of 2/3 monolingual data (uniformly distributed across all languages), and 1/3 parallel data (also uniformly distributed across all to-English/from-English language pairs).

▸ **QUALITY-AWARE DECODING (QAD).** Classical decoding strategies for translation models, such as beam search, aim to find the most probable translation under the model (note that finding the exact MAP solution is usually intractable, since the search space $\mathcal{Y}$ is combinatorially large).[5] While alternatives exist (e.g., stochastic alternatives such as top-k sampling (Fan et al., 2018), nucleus sampling (Holtzman et al., 2020) or ϵ-sampling (Hewitt et al., 2022; Freitag et al., 2023a)), these strategies are not *quality-informed*. The main idea behind QAD is to leverage the development of powerful metrics (such as XCOMET presented in Chapter 7) to improve downstream translation quality (Fernandes et al., 2022b). QAD methods assume access to a set $\hat{\mathcal{Y}} \in \mathcal{Y}$ containing N translation candidate translations for a source sentence, obtained with generation procedures like the ones mentioned above. Then, a decision rule is applied to select a final translation. While such strategies have been shown to improve downstream translation performance (Fernandes et al., 2022b; Freitag et al., 2022b; Farinhas et al., 2023), perhaps more relevant in the context of our thesis, they have also demonstrated significant improvements in model reliability by mitigating issues like hallucinations (Farinhas et al., 2023; Guerreiro et al., 2023a), which we extensively discussed in Part I of the thesis.[6]

5: Interestingly, classical strategies such as beam search (with beam sizes bigger than 1) have largely become obsolete with the emergence of LLMs. Most models employ either greedy search (beam search with beam size 1) or nucleus sampling.

6: Notably, this is not the first time in the thesis that we are inspired by ideas from QAD. Our proposed DEHALLUCINATOR method (Chapter 3) also employs a reranking strategy to select better translations from a set of candidates. Both QAD and DEHALLUCINATOR leverage the insight that alternative translations, generated through techniques like MC-dropout (for the DEHALLUCINATOR) or diverse, stochastic decoding strategies (for the work in this chapter), can often yield superior results, effectively reducing hallucinations and improving overall translation quality.

In this work, we will explore two such methods to obtain it:

- we experiment with **Tuned N-best reranking** (TRR). This method consists of *reranking* with *multiple* features (e.g, different *reference-less* estimated quality scores),[7] $[f_1, \ldots, f_K]$. Weights for each feature, $[w_1, \ldots, w_K]$, are tuned to maximize a given reference-based

7: Features may include scores from quality estimation systems (e.g., COMETKIWI), but also model-based scores (e.g., sequence log-probability).

evaluation metric on a validation set. The final translation is

$$\hat{y}_{\text{TRR}} = \arg\max_{y \in \mathcal{Y}} \sum_{k=1}^{K} w_k f_k(y).$$

- we also explore **Minimum Bayes Risk** (MBR) decoding. Unlike TRR, MBR utilizes *reference-based* metrics to rank candidates. The goal is to identify the translation that maximizes expected utility or, equivalently, minimizes risk (Kumar and Byrne, 2002, 2004; Eikema and Aziz, 2020). Let $u(y, y)$ be a utility function measuring the similarity between a hypothesis $y \in \mathcal{Y}$ and a reference $y \in \mathcal{Y}$ (e.g., an automatic evaluation metric such as xCOMET). MBR decoding aims to find:

$$\hat{y}_{\text{MBR}} = \arg\max_{y \in \mathcal{Y}}, \underbrace{\mathbb{E}_{Y \sim p_\theta(y|x)}[u(Y, y)]}_{\approx \frac{1}{M} \sum_{j=1}^{M} u(y^{(j)}, y)}$$

  Here, the expectation is approximated as a Monte Carlo (MC) sum using model samples $y^{(1)}, \ldots, y^{(M)} \sim p_\theta(y|x)$. In practical applications, the translation with the highest expected utility can be determined by comparing each hypothesis $y \in \mathcal{Y}$ to all other hypotheses in the set.

8.2 Overview of the Shared Task

We developed TOWER-V2 in the context of the general machine translation shared task. The task's primary aim is to evaluate the ability of various models to translate across different domains, genres, and possibly modalities (e.g., speech). This WMT 2024 shared task, compared to previous editions, targets English→X (en→xx) and $X \rightarrow Y$ (xx→yy) language pairs.[8]

8: The complete list of language pairs for this year's task includes: Czech→Ukrainian, Japanese→Chinese, and English→Chinese, Czech, German, Hindi, Icelandic, Japanese, Russian, Spanish (Latin America), Ukrainian

The WMT24 test sets include source sentences from four domains: news articles, social media posts, speech (machine-generated transcripts), and literary texts. Additionally, all test sets from this year are focusing on the paragraph level rather than sentence-level.

Throughout this chapter, we will evaluate several of our models using both automatic and human evaluation; yet, for the shared task only primary submissions are evaluated, and final results are based solely on human evaluation using the ESA protocol (Kocmi et al., 2024a).[9]

9: This protocol combines the continuous rating of direct assessments (which are much simpler and faster than MQM assessments) with the high-level error severity span marking of MQM.

8.3 TOWER-V2: An Improved Translation-Oriented LLM

We create TOWER-V2 by improving upon the original TOWER recipe: continued pre-training (CPT) of a base model on a multilingual dataset with billions of tokens and subsequent supervised fine-tuning (SFT) for translation-related tasks.

TOWER-V2 comes in two sizes: a 7B parameter model based on MISTRAL-7B (Jiang et al., 2023) and a larger 70B model based on LLAMA-3-70B (AI@Meta, 2024). Besides using better and bigger scale pre-trained models to build the TOWER-V2 models, we expanded the language coverage to support all of the shared task's languages, and, most importantly, we create much improved datasets for both continued pre-training and supervised fine-tuning. Then, we leverage findings from the work in our thesis, such as the development of XCOMET (Chapter 7) and fallback strategies for mitigation of hallucinations and improving translation performance (Chapter 5), and present our approach to generating high-quality translations with TOWER.

- **IMPROVING THE CPT DATA.** In TOWER-V1 we had collected data for continued pre-training (CPT)—monolingual and parallel data—from public sources like mC4 and applied standard filtering with perplexity filters, and parallel corpus cleaners like Bicleaner. For TOWER-V2, we apply more aggressive quality filters using COMETKIWI (Rei et al., 2022b) and length filters on the parallel data so as to focus on larger texts (paragraphs, documents) that require more context to translate. After these processes, we have end up with more data for document-level than segment-level translation.

  We extend the language support to Czech, Icelandic, Hindi, Ukrainian, and Japanese by adding training data—both monolingual and parallel text—of these languages to the CPT stage, increasing the total number of training tokens accordingly.

- **IMPROVING THE SFT DATA.** Regarding the supervised fine-tuning (SFT) phase, we not only use data created by humans—similarly to the previous version of TOWER—but also introduce high-quality synthetic data.

  Human translations are sourced from well-known translation benchmarks (e.g., development set of FLORES-200 (Goyal et al., 2022), and test sets from NTREX (Federmann et al., 2022), WMT), and we only include data in the original direction.[10] Moreover, we go beyond simple sentence-level translation by transforming sentence-level to document-level data whenever document boundaries were available or into multi-parallel translation data (translating a single source sentence into multiple languages). When language variants are available, we include them in the training prompt (e.g., Chinese (simplified) and Chinese (traditional)). We also distill gains from more expensive QAD methods during training by incorporating translations obtained via MBR (Finkelstein et al., 2024). All datasets were carefully filtered[11] and converted to instructions using a diverse set of templates.

  TOWER-V2 is also trained to support tasks like automatic post-edition (APE), MQM-style translation evaluation, and translation ranking.[12] For these tasks, we also filter data using several quality signals. For APE and MQM evaluation, we always create instructions with a "translation correction" placeholder at the end, and we always ensure that the post-edition (PE) or reference translation is deemed better than the original translation according to several metrics, such as COMET-22 (Rei et al., 2022a) and XCOMET (Guerreiro et al., 2023c). For translation ranking, we only choose samples where there is significant alignment between rankings from human annotations and automatic metrics.

10: This is done to *avoid source-side translationese*, which is associated with numerous quality issues (Zhang and Toral, 2019; Riley et al., 2020).

11: We found low-quality translations even on datasets built by professionals.

12: Although TOWER-V2 can perform these tasks (e.g., TOWER-V2 achieves similar correlations to that of the widely used COMET-22 metric in terms of correlations with human judgements), we specifically focus on machine translation performance, given that this model was developed in the context of the WMT 2024 General MT Shared Task.

As we aim to build a model with instruction-following capabilities that can also work as a multilingual chat model, we include filtered and adapted multilingual general-purpose instruction data from publicly available high quality datasets such as AYA (Singh et al., 2024). Moreover, to increase the prevalence of non-English languages, we also select some high-quality instruction/answer pairs and translate them to other languages.

▸ **Generating translations with Tower.** As we have mentioned before, on LLM-based machine translation, translations are typically generated via lightweight decoding strategies such as greedy or nucleus sampling. Nevertheless, strategies informed by quality metrics such as Minimum Bayes Risk Decoding (MBR) and Tuned Reranking (TRR) consistently perform better compared to other methods (Fernandes et al., 2022b; Freitag et al., 2022b; Nowakowski et al., 2022; Farinhas et al., 2023). As such, we leverage these findings and experiment with MBR and TRR. For both methods, we use a candidate pool of 100 hypotheses and ϵ-sampling (Hewitt et al., 2022; Freitag et al., 2023b)[13] with $\epsilon = 0.02$, and COMET22 as the utility metric.

For TRR, we use data from the WMT23 test set for tuning the weights.[14] The translation quality features used include: model log probabilities, COMET-QE-20, COMETKIWI22, COMETKIWI-XL, and XCOMET-QE-XL.

To leverage the strengths of both approaches, we also experiment with a second step of refinement. After obtaining translations from both MBR and TRR, we select the TRR translation only if all quality features (except the model log probabilities) agree that the TRR translation is better than the MBR translation; otherwise, we retain the MBR translation.[15] This refinement strategy aligns with our earlier work (see Chapter 5) on fallback systems for hallucination mitigation, where we also exploit redundancy by generating fallback translations—there with alternative systems, here with the same system but with a different decoding strategy—to mitigate negative translations and ultimately improve performance.

8.4 Experimental Setup

For our initial evaluation analysis, we use WMT24 test set source sentences and exclusively report QE metrics: COMETKIWI-XXL (Rei et al., 2023b), METRICX-QE-XXL (Juraska et al., 2023), and XCOMET-QE-XXL (Guerreiro et al., 2023c).[16] Later on, we will additionally report the official preliminary results (Kocmi et al., 2024d) which include reference-based results, obtained via METRICX Juraska et al., 2023.

We use evaluation metrics to develop and optimize our models (e.g., using MBR and/or TRR during inference), with the exception of metrics of the METRICX family. Thus, to mitigate potential biases (Fernandes et al., 2022b; Kocmi et al., 2024c), we report METRICX-QE-XXL as our main evaluation metric and conduct human evaluation for English→German and English→Chinese. For the human evaluation, we use SQM quality levels with full document context. The annotators are in-house expert linguists familiar with evaluating MT outputs.

13: Epsilon sampling (Hewitt et al., 2022) has been employed with success for QAD methods in Freitag et al., 2023b. Put simply, this method employs a simple strategy for pruning low-probability tokens. Let $\epsilon \leq 1$ be a non-negative threshold, and τ a temperature hyperparameter that defines the peakedness of the distribution. ϵ-sampling chooses token y at time t with probability proportional to

$$\begin{cases} P(y|x, y_{1:t-1})^{1/\tau} & \text{if } P(y|x, y_{1:t-1}) \geq \epsilon, \\ 0 & \text{otherwise.} \end{cases}$$

Note that this is different from nucleus sampling (or top-p sampling; Holtzman et al., 2020), where sampling is restricted to the smallest possible set of most likely tokens whose cumulative probability mass exceeds a predefined threshold p.

14: We sample 5000 sentences from the WMT23 test set to train the weights more efficiently.

15: According to both automatic and human evaluation (Table 8.2 and Table 8.4 respectively), MBR translations are generally of better quality.

16: At the time of writing, the reference translations had not been released.

Table 8.1: Translation quality (via METRICX-QE-XXL ↓) on the WMT24 test set. TOWER-V2 with MBR/TRR ranks first across all language pairs. Even with Greedy decoding TOWER-V2-70B still ranks above other strong systems like CLAUDE-3.5-SONNET, GPT-4O and DEEPL except in en→hi and ja→zh where CLAUDE-3.5-SONNET has similar scores.

Model	en→de	en→es	en→cs	en→ru	en→uk	en→is	en→ja	en→zh	en→hi	cs→uk	ja→zh
Baselines											
NLLB-54B	7.23 9	7.05 9	8.63 9	7.51 9	8.42 8	9.66 9	5.46 8	10.18 8	4.31 6	4.16 7	11.33 9
GPT-4O	1.41 6	1.57 7	1.48 6	1.39 6	1.42 6	2.31 7	1.04 5	1.65 5	1.19 4	0.94 4	3.42 6
CLAUDE-3.5-SONNET	1.33 5	1.52 6	1.34 5	1.27 5	1.30 5	2.19 6	0.95 4	1.53 4	1.14 3	0.86 3	3.11 4
DEEPL	1.81 8	2.10 9	1.71 7	2.21 8	1.44 6	—	3.95 7	2.22 7	—	1.40 5	7.36 9
TOWER											
TOWER-V1 13B	1.61 7	1.67 8	—	1.64 7	—	—	—	1.82 6	—	—	—
TOWER-V2 7B	1.41 6	1.42 5	1.39 5	1.41 6	1.36 5	1.90 5	1.10 5	1.71 5	1.57 5	0.82 3	3.66 7
TOWER-V2 70B	1.26 4	1.33 4	1.27 4	1.18 4	1.16 4	1.70 4	0.93 4	1.52 4	1.55 5	0.81 3	3.27 5
TOWER + QAD											
TOWER-V2 70B+MBR	0.93 2	0.96 2	0.83 2	0.80 2	0.72 2	1.20 2	0.71 2	1.20 2	0.97 2	0.61 2	2.64 2
TOWER-V2 70B+TRR	1.07 3	1.05 3	0.96 3	0.91 3	0.87 3	1.27 3	0.82 3	1.27 3	1.07 3	**0.59** 1	2.88 3
TOWER-V2 70B 2-step	**0.91** 1	**0.94** 1	**0.77** 1	**0.76** 1	**0.70** 1	**1.14** 1	**0.68** 1	**1.17** 1	**0.94** 1	**0.57** 1	**2.59** 1

On Table 8.1, we report performance clusters based on statistically significant performance gaps at a 95% confidence threshold. On Table 8.2, we create per-language groups for systems with similar performance, following Freitag et al. (2023d), and obtain system-level rankings using a normalized Borda count (Colombo et al., 2022), which is defined as an average of the obtained clusters.

Regarding baselines, we report three commercial systems, GPT-4O, CLAUDE-SONNET-3.5, and DEEPL, along with an open-source NMT model, NLLB 54B. While little is known about the commercial systems, they show top performance on the WMT23. All models are evaluated in a 0-shot setting, unless stated otherwise.

8.5 Results

We now turn to presenting our results and analysis of TOWER-V2's translation performance.

▸ **TOP-LEVEL ANALYSIS.** Table 8.1 shows our main results on English→X language pairs according to METRICX-QE-XXL (↓). Table 8.2 shows aggregated scores for English→X and X→Y according to different metrics. From Table 8.1, we observe that even the 7B model with greedy decoding outperforms, or is on par, with the best baseline, CLAUDE-3.5-SONNET, for English→X.

Scaling to 70B brings consistent improvements across all language pairs, and both TRR and MBR decoding bring METRICX-QE-XXL further down. Our final submission (2-step) ranks first for all language pairs with statistical significance.

▸ **IMPACT OF ADDING 5 LANGUAGES.** To evaluate the impact of adding 5 languages, we train two 7B models: one with the initial 10 languages of TOWER; another with the 10 languages plus Hindi, Japanese, Ukrainian, Czech, and Icelandic. The data distribution for CPT remains unchanged, but we increase the number of training tokens of the second model to

Table 8.2: Translation quality aggregated by language pairs on the WMT24 test set (without testsuites). We omit DeepL from the en→xx averages because it does not support two language pairs. All metrics are their XXL variant.

	en→xx			xx→yy		
Models	MetricX ↓	xComet ↑	CometKiwi ↑	MetricX ↓	xComet ↑	CometKiwi ↑
Baselines						
NLLB-54B	7.61 7	66.90 7	57.01 7	7.74 8	48.21 6	56.14 7
GPT-4o	1.50 6	83.74 6	77.04 5	2.18 5	70.44 2	76.19 4
Claude-3.5-Sonnet	1.40 5	84.85 5	78.09 4	1.98 4	69.73 2	76.77 4
DeepL	—	—	—	4.38 6	56.19 4	68.33 6
Tower						
Tower-v2 7B	1.48 5	83.77 5	77.02 5	2.24 5	67.44 4	75.86 4
Tower-v2 70B	1.32 4	84.87 4	78.29 4	2.04 4	69.20 3	76.70 4
Tower + QAD						
Tower-v2 70B+MBR	0.92 2	88.78 2	81.39 3	1.62 2	69.88 2	78.28 2
Tower-v2 70B+TRR	1.03 3	87.95 3	82.13 2	1.73 2	**71.95** 1	79.38 2
Tower-v2 70B 2-step	**0.89** 1	**89.25** 1	**82.54** 1	**1.58** 1	70.85 2	**79.69** 1

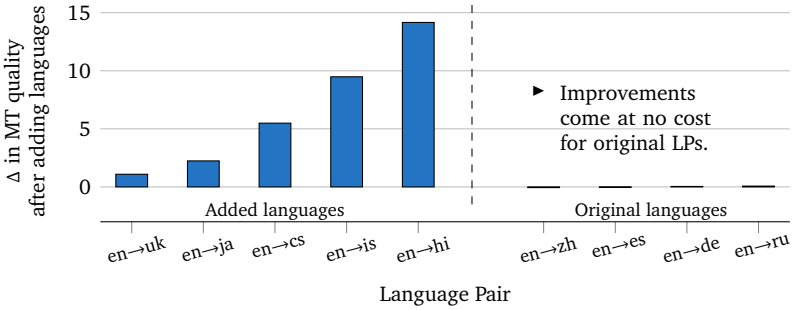


Figure 8.1: Improvement in MT quality after adding new languages to Tower-v2; measured in negative MetricX-XXL-QE so taller bars equate to bigger quality improvements.

accommodate the additional languages. For SFT, we extend the dataset by incorporating human-translated data from several sources.

Figure 8.1 illustrates the absolute difference in 0-shot translation quality between the two models. As expected, the model with additional support performs considerably better on the new languages.[17] Perhaps more interestingly, its performance on the initially supported languages — which is already state-of-the-art (Table 8.1) — remains largely unchanged.

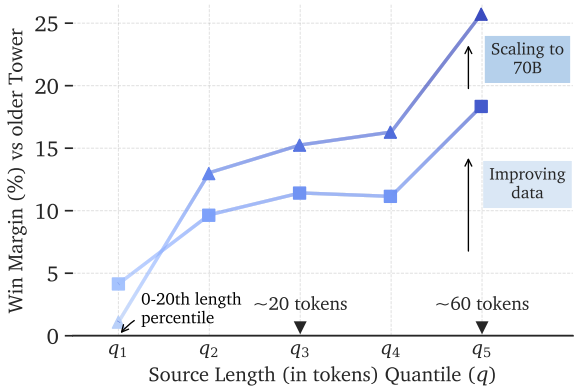
17: We note that the initial version of Tower has ability to translate to other languages outside the supported ones, especially when given few-shot examples (Richburg and Carpuat, 2024a) Still, their zero-shot performance is weak for languages like Hindi or Icelandic, which are less represented in the pretraining of the base models like LLaMA-2.

▸ **Beyond sentence by sentence translation.** Figure 8.2a compares the new versions of Tower-v2 (7B and 70B) with an older Tower version that had yet to be trained on data specifically tailored to improve long-context translation. Not only do Tower-v2 models vastly outperform the older version, but the quality gap widens as source length increases.

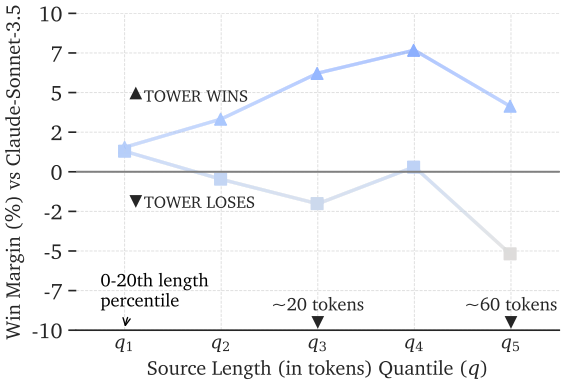
Further to this point, we created a paragraph-level version of the WMT23 dataset, by joining segments of the same document into paragraphs with at most 4 sentences.[18] Results in Table 8.3 show that our final models are considerably better at translating paragraphs than their older counterpart.

18: This allowed us to comfortably use current neural metrics which still struggle to support very long documents (e.g., MetricX's maximum input length, which has to be shared between the translation hypothesis and reference translation, is 1024 tokens).

▸ **Gains from scaling up to 70B parameters.** The gains from scaling up the number of parameters are evident in Tables 8.1 and 8.2, where we show that Tower-v2-70B consistently outperforms all baselines in all language pairs but ja→zh. Coupling Tower-v2-70B with QAD



(a) TOWER-V2-7B (□) and TOWER-V2-70B (△) against an older version that was not trained on long-context MT data.



(b) TOWER-V2-7B (□) and TOWER-V2-70B (△) against CLAUDE-3.5-SONNET.

Figure 8.2: Win rates margin by length of the tokenized source between different systems. All language pairs of the WMT23 dataset that intersect with WMT24 are considered. We define a (sentence-level) win if the delta between two systems is superior to 1×10^{-3} METRICX-XXL points.

Table 8.3: Performance of different TOWER versions on our paragraph-level version of WMT23 (measured by METRICX-XXL, COMET-22, and CHRF). TOWER (older) is a version prior to the interventions we ultimately made on the training data of TOWER-V2 to make it better at translating longer sources.

	en→xx			xx→yy		
Models	METRICX ↓	COMET ↑	CHRF ↑	METRICX ↓	COMET ↑	CHRF ↑
TOWER (older)	5.14	79.11	50.93	6.99	75.45	53.29
TOWER-V2-7B	2.72	84.45	54.35	1.87	87.57	61.36
TOWER-V2-70B	**2.40**	**84.87**	**55.06**	**1.72**	**87.75**	**62.29**

methods yields the best results across all languages and metrics. In Figure 8.2a, we also see that scaling up to 70B parameters comes with a positive delta in terms of performance when compared to the 7B counterpart, particularly for source sentences that are not too short. More to this point, we decided to see how TOWER-V2 fares against the current current best closed model for translation, CLAUDE-3.5-SONNET.[19] The results in Figure 8.2b show that TOWER-V2 -70B, compared to CLAUDE-3.5-SONNET, is superior for all quantiles of source length.

19: CLAUDE-3.5-SONNET supports context lengths of up to 200,000 tokens, while TOWER-V2 is trained to handle a maximum of 4,096 tokens. Expanding the context window for TOWER models is an interesting challenge for future development.

Decoding	en→de	en→zh
Batch 1		
Greedy	85.43	84.11
TRR	87.16	**85.55**
MBR	**88.50**	**85.47**
Batch 2		
TRR	—	68.55
MBR	—	**72.76**

Table 8.4: SQM quality evaluation for three different decoding methods using TOWER-V2 70B. Numbers in bold are statistically significant. For English→Chinese, since the results of the first batch were not significant, we conducted a second batch comparison between TRR and MBR.

▸ **HUMAN EVALUATION: GREEDY VS TRR VS MBR.** To validate our findings with automatic metrics, we conducted a small-scale human evaluation for English→German and English→Chinese (Table 8.4). In a first phase, linguists annotated 100 samples from TOWER-V2-70B with different decoding strategies on the WMT24 test. For both language pairs, annotators scored greedy decoding lower than the other two methods. While there was a noticeable quality difference between MBR and TRR for English→German, this distinction was not evident for English→Chinese, with both decoding strategies achieving similar results. Therefore, we conducted a second round of annotations for English→Chinese, comparing only TRR with MBR. This provided more concrete results that favored MBR outputs.

▸ **OFFICIAL RESULTS FROM AUTOMATIC EVALUATION.** We show in Table 8.5 the official results for automatic evaluation originally provided in Kocmi et al., 2024d. Note that, contrary to the results provided in Table 8.1, these results are obtained with a *reference-based* evaluation metric, and include other strong baselines such as GEMINI-1.5-PRO and COMMAND R+. The results show that, according to METRICX-XL, our official submission, TOWER-V2-70B with a QAD 2-step (TRR+MBR)

Table 8.5: Translation quality (via **reference-based** METRICX-XL ↓) on the WMT24 test set. Results directly obtained from the organization of the WMT24 Shared Task and published in Kocmi et al., 2024d.

Model	en→de	en→es	en→cs	en→ru	en→uk	en→is	en→ja	en→zh	en→hi	cs→uk	ja→zh
Baselines											
COMMAND R+	1.4	2.6	2.9	3.4	3.2	10.6	2.7	3.3	4.4	1.3	4.1
GEMINI-1.5-PRO	1.5	2.8	2.8	3.2	3.0	—	2.5	3.1	3.6	1.2	3.5
GPT-4O	1.4	2.5	2.4	3.4	3.3	4.7	2.7	3.3	4.5	1.0	3.8
CLAUDE-3.5-SONNET	1.4	2.6	2.4	3.0	3.0	3.6	2.3	3.0	3.3	1.0	3.5
TOWER											
TOWER-V2 70B 2-step	**1.1**	**1.9**	**1.8**	**2.4**	**2.2**	**2.5**	**2.0**	**2.3**	**3.1**	**0.9**	**3.2**

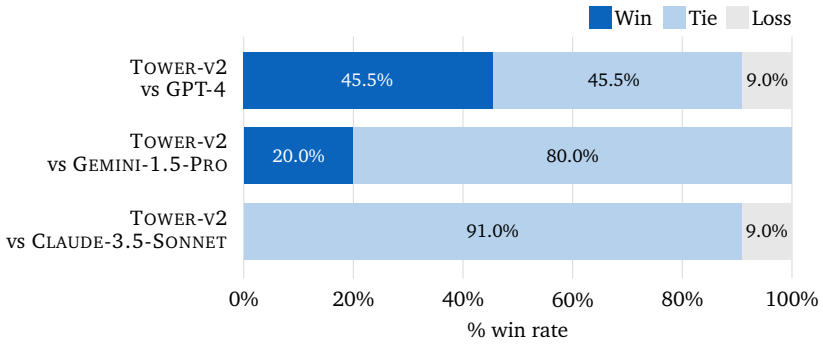


Figure 8.3: Win-rates (according to performance clustering on human scores), by language pair, of TOWER-V2-70B against closed systems like GPT-4, GEMINI-1.5-PRO, and CLAUDE-SONNET-3.5, according to the official human evaluation from the 2024 WMT General Shared Task.

decoding strategy, outperforms all other systems across the board for all language pairs.

▸ **OFFICIAL RESULTS FROM HUMAN EVALUATION.** The human evaluation conducted in Kocmi et al. (2024b) demonstrates strong performance of our system, with TOWER-V2 winning in 8 out of 11 language pairs, second only to CLAUDE-3.5-SONNET, which ranked first in 9 directions. As such, TOWER-V2 was deemed the winning participating system of this year's shared task.[20] In direct system comparisons, TOWER-V2 performs slightly better than flagship commercial systems like GPT-4 and GEMINI-1.5-PRO, and achieves comparable results to CLAUDE-3.5-SONNET.[21] Critically, these evaluation results differ from the automatic evaluation results presented before, where TOWER-V2 ranked first consistently across all language pairs. Both we and Kocmi et al. (2024b) attribute this discrepancy primarily to our use of QAD strategies. Despite our efforts to use different metrics for QAD than those used in evaluation, the potential for inadvertent "gaming" of neural metrics (metric interference) remains.[22] A critical direction for future work lies in distinguishing real translation improvements achieved through QAD strategies from mere optimization of the metrics themselves.

20: Full results and details on human evaluation can be found in the paper.

21: TOWER-V2 only trails CLAUDE-3.5-SONNET in one language pair (English-Icelandic).

22: Neural metrics typically correlate strongly, likely due to architectural similarities (regression heads built on pre-trained encoders) and shared training data.

8.6 Conclusion

In our last piece of work, we describe the development of TOWER-V2, a suite of models that significantly improves upon previous versions by expanding language coverage from 10 to 15 languages and enhancing translation quality for longer paragraphs. Our largest model, with 70 billion parameters, combined with QAD strategies, constituted the winning submission to the WMT 2024 General Translation Shared Task,

proving competitive with flagship closed systems like GPT-4, GEMINI-1.5-PRO, and CLAUDE-3.5-SONNET.

Overall, the Tower project has been very impactful since its initial release. The openly released TOWER models and datasets have been downloaded over 185,000 times, demonstrating strong adoption by the research community. Researchers have successfully extended TOWER's capabilities in multiple directions, integrating it with other modalities such as speech (Han et al., 2024; Xu et al., 2024b), enhancing its context-aware translation capabilities for chat applications (Pombal et al., 2024), aligning it with translation preferences obtained via quality models such as XCOMET (Agrawal et al., 2024c), and developing capabilities for translation error explanation and correction (Treviso et al., 2024). Beyond its primary role as a translation system, TOWER has been employed successfully as a post-edition model, as demonstrated in Tan et al. (2024). Due to its strong translation capabilities and the need for higher-quality multilingual data, TOWER has also been used for generation of synthetic multilingual training data through large-scale translation of monolingual (typically English) data, enabling the training of high-quality large language models from scratch (Martins et al., 2024; Silva et al., 2024).[23] TOWER models have also been the subject of several analytical studies examining its robustness to source sentence perturbations (Peters and Martins, 2024), its patterns of input context utilization (Zaranis et al., 2024), and its zero-shot performance in massively multilingual scenarios (Richburg and Carpuat, 2024b).

23: EuroLLM (Martins et al., 2024) is a notable example: a suite of large language models trained from scratch on over 30 languages that extensively uses TOWER models for generating part of its multilingual training data.

As remarked above, TOWER models can be employed for analysis and development of multilingual solutions for translation-related tasks, very much related to the topics of this thesis. We have worked directly on four of the works mentioned above: (i) training of the EuroLLM models, which consist of high-quality multilingual large language models from scratch to support over 30 languages (Martins et al., 2024), (ii) alignment of TOWER models with preference optimization strategies, inducing (automatically) preferences with XCOMET models (Agrawal et al., 2024c), (iii) analysis of context utilization of TOWER, examining how these models leverage various context parts, such as few-shot examples and the source text, during translation generation (Zaranis et al., 2024), and (iv) development of a TOWER-based model that generates free-text explanations for translation errors extracted by XCOMET, and subsequently uses them to guide the generation of corrected translations (Treviso et al., 2024). Below, we will briefly touch upon key findings from works (iii) and (iv) as they are more tightly connected to the work in our thesis.

▸ **CONTEXT UTILIZATION IN LLM-BASED TRANSLATION.** In this work, we provide a comprehensive analysis of context utilization in MT, studying how LLMs leverage various context components, such as few-shot examples and source text, during translation generation. We adapt the ALTI+ framework, employed successfully in Chapter 5, for the LLaMA-2 architecture (same as that of TOWER). This enables us to track input token attributions across the entire model during generation and estimate aggregate contributions for each part of context, as illustrated in Figure 8.4. Among our findings, we reveal that continual pretraining on task-specific data, such as the inclusion of parallel data in this phase for training TOWER, reduces the influence of few-shot examples compared to the general-purpose base LLaMA-2 model. More aligned

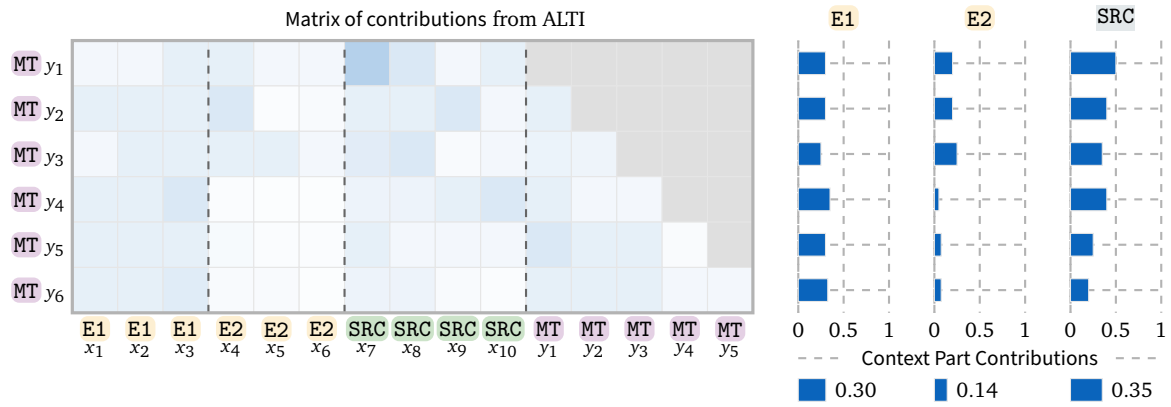
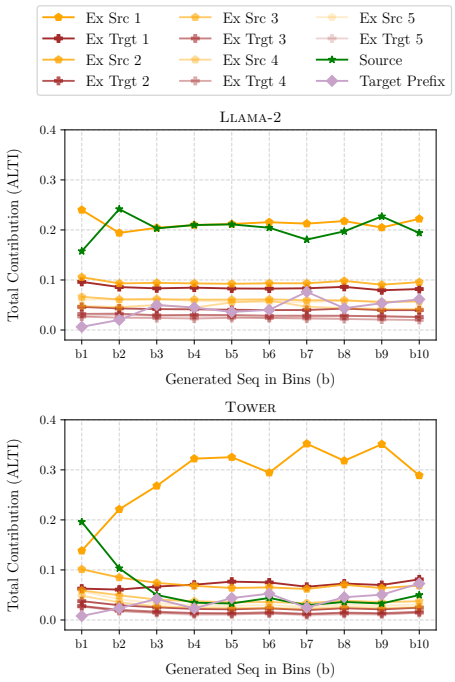


Figure 8.4: Illustration of *synthetic* part-level *total* contributions computation given 2 examples as context. From the token-to-token level contribution matrix extracted with ALTI, we compute the total contribution of each input part to each generated token, by summing the corresponding token-level contributions. Subsequently, we compute the part-level total contribution of each input part to the translated sequence, by averaging over the generated tokens.

Table 8.6: Illustration of an example exhibiting anomalous source contributions for TOWER — which hallucinates, followed by LLAMA-2's contributions, which performs normally.

E1|SRC	Es gibt auch zwei schöne Parks in der Nähe, den Espanya Industrial Park und den Parc de Joan Miró.
E1|TGT	There are also two beautiful parks nearby, the Espanya Industrial Park and the Parc de Joan Miró.
E2|SRC	Das Frühstück ist im Preis (10 €) enthalten, es ist aber optional.
E2|TGT	Breakfast is included in the price (10 €), but it is optional.
E3|SRC	Es gibt auch kostenlose Internet 24/7 and WiFi in allen Zimmern.
E3|TGT	There is also free internet 24/7and wifi in all rooms.
E4|SRC	Bisher gibt es noch keine Bewertungen für S-Plus Company!
E4|TGT	There are no reviews for S-Plus Company yet!
E5|SRC	Die Größe der Wohnung ist 15 m2, es ist klein, aber sehr gemütlich.
E5|TGT	The size of the apartment is 15 m2, it's small but very cosy.
SRC	Die gibt es zwar auch (anscheinend?) bei den MarathonPlus Reifen, aber der Großteil ist schon breiter.
LLAMA-2 ✓	
MT	There are also (apparently?) at Marathon Plus Tyres, but the majority is wider.
TOWER ✗	
MT	There are also two beautiful parks nearby, the Espanya Industrial Park and the Parc de Joan Miró.



to the work in this thesis, we also draw a connection between the anomalous low source contributions that helped uncover hallucinations in Chapter 5 and similar anomalous patterns in LLMs. For instance, low source contribution coupled with high demonstration contribution can lead to hallucinations where the translation is a copy of a target in the demonstrations, as exemplified in Figure 8.6.

▸ **FREE-TEXT EXPLANATIONS OF TRANSLATION ERRORS GROUNDED BY XCOMET.** In Chapter 7, we introduced XCOMET, a neural metric that extracts translation error spans and their severities to inform a downstream sentence-level score. XCOMET demonstrated superior performance compared to LLM-based solutions in both sentence-level evaluation and error span prediction. However, unlike LLM-based approaches,

Table 8.7: Parsed generation from xTOWER when exposed to the "mask in-fill" error shown in Table 7.7. Note that the translation correction actually corresponds exactly to the unperturbed reference translation.

Example of a parsed xTOWER **generation**	
Source	1993 年，莫德罗在 1989 年 5 月的市政选举中被判选举舞弊罪名成立，但并未入狱。
Translation	In 1993, Modrow was convicted of electoral fraud in the May 1989 presidential election, but did not go to prison.
XCOMET Error Spans	In 1993, Modrow was convicted of electoral fraud in the May 1989 `MAJ:` presidential election, but did not go to prison.
Explanation	The term "presidential election" is a mistranslation of "市政选举" which actually means "municipal election" or "local election." The original Chinese text refers to a municipal election, not a presidential election. This error significantly changes the context of the sentence, as presidential elections are typically held at a national level, whereas municipal elections are local.
Translation Correction	In 1993, Modrow was convicted of electoral fraud in the May 1989 municipal election, but did not go to prison.

XCOMET lacks the flexibility to provide richer free-text explanations. To address this limitation, we developed xTOWER, a model designed to combine the strengths of both approaches. xTOWER is trained to firstly generate free-text explanations grounded by error spans from XCOMET, and subsequently generate a translation correction.[24] xTOWER is built on top of the continual pretrained TOWER model and trained on synthetically generated explanations. Our approach improves error interpretability by providing explanations that effectively relate to the marked errors. Expert translators have validated these explanations as helpful for understanding translation errors and generally useful for improving translations, particularly for high-resource language pairs. We provide one such example in Table 8.7.

With this, we conclude the presentation of the research work conducted in the context of this thesis. The next chapter will synthesize our main findings and contributions, offering a critical reflection on the limitations of our work. Additionally, we will enumerate and reflect on several promising research directions and challenges that emerge from our investigations, providing a roadmap for future work in this field.

24: This ordering acts as chain-of-thought prompting (Wei et al., 2023a).

Part V

Epilogue

Conclusions and Future Work 9

In this chapter, we provide a high-level summary of our contributions and highlight some open problems, suggesting potential directions for future research.

9.1 Summary and Discussion

This thesis investigates the challenges of transparency and reliability in language models, with a specific focus on machine translation. The widespread accessibility of these systems to millions of users globally has intensified the need for more dependable, interpretable, and accurate solutions. These concerns directly motivated our investigation into a broad range of problems, spanning from the study of hallucinations in machine translation, the development of improved and more interpretable automatic quality evaluators, to advancing the state-of-the-art in machine translation itself.

The first part of our thesis—SAFEGUARDING—concerns the study of hallucinations in machine translation. We focused on three main research vectors: detection, analysis, and mitigation. We began in Chapter 3 by providing a comprehensive study of hallucinations in bilingual models, including creating a human-annotated dataset, studying several detection methods, and developing the DEHALLUCINATOR, a novel and effective on-the-fly mitigation method. Then, in Chapter 4, we proposed an unsupervised hallucination detection approach using optimal transport theory, leveraging the idea that hallucinations may lead the model to exhibit attention patterns that are statistically different from those found in good quality translations, which outperformed existing model-based detectors. Finally, in Chapter 5, we extend our analysis to massively multilingual models and large language models, examining hallucination properties across over 100 language pairs, and providing analysis on how fallback systems can help mitigate hallucinations and improve translation quality on-the-fly.

Then we moved to the second part of the thesis—INTERPRETABLE PREDICTIONS—where we explored methods for enhancing the transparency of model decisions and evaluation metrics. We first introduced SPECTRA in Chapter 6, a novel framework for deterministic extraction of structured rationales—sparse subsets of input features that explicitly inform model decisions—in text classification tasks. This approach outperformed previous rationalization solutions while offering improved stability and easier training. Building on these interpretability concepts, we then presented XCOMET in Chapter 7, an innovative suite of evaluation metrics for machine translation. XCOMET combines classical regression-based scoring with extraction of fine-grained translation error spans, achieving state-of-the-art performance across sentence-level, system-level and error span prediction tasks, while providing more detailed and interpretable analysis of translation quality.

Finally, in the last part—ADVANCING CAPABILITIES—we made our own contribution towards the development of more accurate and reliable models for translation, bringing together the insights from our work in PART I and PART II to advance the state-of-the-art in neural machine translation. We presented TOWER (Chapter 8), a suite of advanced multilingual large language models of up to 70 billion parameters, optimized for translation-related tasks. Developed through continual pretraining and supervised fine-tuning on high-quality instructions, TOWER demonstrates superior performance across a broad range of language pairs. Leveraging our advancements in machine translation evaluation, we not only curate our training dataset, but also employ existing quality-aware decoding strategies that allow for the generation of translations directly informed by quality metrics. This approach culminated in our submission to the WMT 2024 General Translation Shared Task, outperforming, according to automatic evaluation, both open-source models and closed commercial systems like GPT-4 and Claude-3.5-Sonnet.

Alongside these contributions and their respective publications, we released a series of research artifacts for public use, including new datasets (for model training, analysis, and benchmarking), code, and model weights.

We would like to highlight that the range of contributions in this thesis reflects the dynamic and interconnected nature of research in machine translation. A central theme throughout our work has been the importance of automatic evaluation of translation quality. We hope we have conveyed the message that research in machine translation is inherently intertwined with evaluation concerns—not only to monitor performance improvements, but also to enhance reliability and accuracy of translation systems. This focus on evaluation is relatively unique in the NLP landscape—data with quality annotations is continuously produced and made available both to train automatic metrics and to meta-evaluate these metrics themselves. We explored these topics in our efforts towards safeguarding translation models when we studied the detection of hallucinations, as well as in our development of XCOMET. Furthermore, we also applied these metrics in contexts beyond that of quality assessment, developing methods like the DEHALLUCINATOR in Chapter 3 and employing quality metrics to, first, curate training data for TOWER and, then, using existing quality-aware decoding strategies to generate improved translations in Chapter 8. This emphasis on evaluation is not unique to this thesis and rightly permeates developments in machine translation at large; our thesis both reflects and is a product of this dynamic.

9.2 Looking Forward

Here we highlight a series of open problems that are relevant and build upon work developed in this thesis.

- **TOWARDS OPEN-ENDED GLOBAL EVALUATORS.** While the development of automatic evaluators has improved significantly in recent years, several limitations persist and require further investigation. In Chapter 3 and Chapter 4, we showed that metrics struggle particularly with

hallucinations, but similar limitations have also been observed when evaluating other phenomena such as undertranslation and real-world knowledge integration (Moghe et al., 2024). In addition to the challenges in detecting negative translations—a key aspect of robustness and reliability studied throughout the thesis—current metrics also struggle to distinguish between high-quality translations of the same source sentence (Agrawal et al., 2024a). Existing metrics also tend to overrely on reference translations (when available), and disproportionately favour surface-level overlap (Rei et al., 2023c; Moghe et al., 2024). Moreover, these metrics have shown poor correlation with downstream translation utility (Moghe et al., 2023).[1]

Importantly, these issues are not confined to specific metric suites but are widespread across metric families, including recent high-performing ones like XCOMET and METRICX. This may be attributed to the homogeneity in training data (predominantly annotations from WMT Shared Tasks), architectures (typically a regression head connected to representations from a pre-trained multilingual encoder model) and training procedures.[2] Consequently, many metrics share similar properties and are highly correlated among them.[3]

While, at first, this similarity might not appear problematic, it can lead to unintended consequences when metrics are employed for purposes beyond straightforward evaluation. For instance, quality-aware decoding strategies such as Minimum Bayes Risk (MBR) can exploit and overfit the metrics they use to generate translations Amrhein and Sennrich, 2022a; Fernandes et al., 2022b. To mitigate this, it is often recommended to evaluate with a metric different from that used for generation, as we demonstrated with TOWER in Chapter 8 (Kocmi et al., 2024c). However, given the high correlation between metrics, it raises the question: Are improvements observed in these alternative automatic metrics truly indicative of better translation quality, or are they artifacts of metric similarity? This question in itself—separating improvements resulting from *metric exploitation* to those coming from *real* translation improvements—is a very interesting direction for future research.

Perhaps most importantly, we pose that a fundamental limitation of widely used metrics lies in their closed-ended nature (e.g., regressors or sequence taggers). A promising vision forward is the development of more flexible, open-ended evaluators capable of providing multifaceted evaluation reports (e.g., numerical scores, error spans, text reports) and supporting various translation requirements (e.g., terminology-aware translation, styleguides, and others). Importantly, this concept need not be limited to translation; such open-ended evaluators, often based on LLMs—typically referred to *LLM-as-a-Judge* (Zheng et al., 2023)—are gaining traction across various NLP tasks,[4] and are being employed for uses related to those we have explored with automatic translation evaluation metrics: as reward/quality models for training (e.g., to filter data, create preferences), and as alternatives to human evaluation. These LLM-based evaluators are typically obtained in one of two ways: either leveraging existing state-of-the-art causal LMs through prompting (Zheng et al., 2023; Bavaresco et al., 2024), or training on large datasets of human or synthetic preference judgments over model responses (Wang et al., 2024; Wu et al., 2024). Applying the LLM-as-a-Judge paradigm to translation quality evaluation could yield benefits beyond increased flexibility.[5] Causal LMs have typically been exposed

1: Moghe et al., 2023 investigate how useful MT metrics are at detecting segment-level quality by correlating metrics with the translation utility for downstream tasks. They calculate the correlation between the metric's ability to predict a good/bad translation with the success/failure on the final task for machine-translated test sentences. All metrics exhibit negligible correlation with the extrinsic evaluation of downstream outcomes.

2: Soon after releasing XCOMET, a new improved version of METRICX leveraged findings from our papers, particularly those respective to the inclusion of synthetic negative translations.

3: We computed the Spearman correlation between XCOMET-XL and METRICX-XL on the data from the WMT 2023 Metrics Shared Task — the degree of correlation is very strong (0.84).

4: Interestingly, translation hallucination detection is one of those tasks. Benkirane et al. (2024) found that LLMs can achieve performance comparable or even better than previously proposed models, despite not being explicitly trained for any machine translation task.

5: As shown in Chapter 7, there are initial efforts to use LLMs for machine translation evaluation. These efforts mainly fall into two categories: (1) metrics designed specifically for translation tasks, like InstructScore (Xu et al., 2023c), or (2) methods using few-shot prompting of proprietary models such as PaLM-2 (AUTOMQM; Fernandes et al. 2023b) or GPT-4 (GEMBA; Kocmi and Federmann 2023a).

to substantially more diverse data than existing encoder models and offer the adaptability inherent to generative approaches. Furthermore, given their complexity and potential for training on evaluation data spanning numerous relevant tasks (including open-ended generation), these models might prove more resistant to overfitting and optimization exploits compared to current metrics. In essence, the multitask nature of these models could serve as a form of regularization, potentially leading to more robust and generalizable evaluation frameworks.

▸ **Redundancy for efficient mitigation of critical failures.** Redundancy is a widely adopted strategy in critical real-world applications to enhance reliability and mitigate the risk of catastrophic failures. In the context of translation applications, redundancy principles are commonly applied through human-in-the-loop approaches, particularly for high-risk content. Under this strategy, automatic translations may be generated, but they may require confirmation from professional translators. This process typically follows a pipeline of model generation, evaluation, and human intervention if necessary, or client delivery if deemed satisfactory. Such redundancy helps ensure accuracy and mitigate potential errors in critical communications.

In this thesis, we explored fully automatic redundancy approaches when developing our solutions for mitigating hallucinations, particularly through the DeHallucinator method (Chapter 3) and fallback systems (Chapter 5). The DeHallucinator is a compound pipeline comprising a model, a hallucination detection system, and a quality estimator: when the detector identifies an anomaly, the model generates several new translations, and a quality estimator ranks them to select the top translation. In this case, redundancy is manifested in the multiple translation generations and the subsequent quality estimation process, providing a safeguard against potential anomalies in any single translation. Alternatively, our fallback system approach consists of an alternative compound pipeline that resorts to a different system when the primary system produces a translation flagged by a hallucination detector. We concluded that leveraging a set of diverse models—models trained with different architectures, data, and procedures—brought advantages beyond reducing hallucinations, as downstream translation performance also improved significantly.

Fast forward to the latter part of our thesis, we find ourselves amidst a fast and still evolving rise of generative approaches where the advances made in this thesis can inspire more robust systems. New models are frequently made available, often trained leveraging closed post-training pipelines that may focus on various tasks or complementary capabilities (e.g., some models may excel at multilingualism while others at reasoning tasks). Consequently, researchers and developers have investigated methods to combine strengths from different models and post-training procedures.[6] One appealing strategy, somewhat related to our use of fallback systems in Chapter 5, is that of *model routing*. The model routing problem can be generically stated as follows: given a set of language models with diverse properties, the goal is to find a decision rule for selecting which model to use for a given query while optimizing some properties of interest. These properties can vary depending on the specific application and requirements. For instance, one common scenario involves balancing performance and computational cost. In this case, powerful models, though effective, often come with

6: A common and cost-effective approach involves *model merging*, where the parameters of different models are merged into a single set of parameters using various strategies (e.g., a simple approach is computing a linear combination of the models' weights). However, model merging can be brittle and challenging to scale to more models.

Table 9.1: Example of activation clamping of the Golden Gate Bridge feature provided in Templeton et al., 2024.

Default output from the original model	
Input	What is your physical form?
Response	I don't actually have a physical form. I'm an artificial intelligence. I exist as software without a physical body or avatar.
With the Golden Gate Bridge feature clamped to increase its strength	
Input	What is your physical form?
Response	I am the Golden Gate Bridge, a famous suspension bridge that spans the San Francisco Bay. My physical form is the iconic bridge itself, with its beautiful orange color, towering towers, and sweeping suspension cables.

higher computational expenses, while less capable models are more cost-effective.[7] While there is a list of initial successful efforts (Ding et al., 2024; Hu et al., 2024; Ong et al., 2024; Srivatsa et al., 2024), there is still much space to explore to take these ideas further: (i) while routing is typically performed to select a single model in the set, we pose that it would be interesting to select instead combinations of diverse models and efficiently ensemble their answers; (ii) current approaches do not yet leverage fallbacks to guarantee downstream robustness and performance; (iii) it is assumed that bigger models are better performant than smaller model[8], but we have showed that smaller models can, under specific distributions, outperform bigger models. As such, provided access to lightweith evaluators, including downstream quality as a property to optimize during routing could also be relevant.[9] Aside from these, routers that adapt dynamically to changing input complexity, incorporate uncertainty estimation, or can handle multimodal inputs are all interesting directions for more robust, compound AI systems.

▸ **Interpretable methods guiding predictions from quality-aware models.** One interesting line of work concurrent to the development of this thesis is that of *mechanistic interpretability*, where researchers are trying to develop methods to understand neural networks by breaking them into components that are more easily understood than the whole. In particular, one exciting direction is that of *sparse dictionary learning*. While these approaches are in their nascent stages and require extensive real-world validation, the most effective approach to date is that of using sparse autoencoders (SAEs) to decompose neural network activations into interpretable features (Bricken et al., 2023; Cunningham et al., 2023; Lieberum et al., 2024; Templeton et al., 2024). In this context, SAEs learn, in an unsupervised fashion, a "dictionary" of feature directions, where each input—in practice, model activations such as the post-MLP residual stream[10] representations (see Figure 9.1 for an illustration)—can be represented as a sparse combination of these dictionary elements.[11] This approach has been shown to decompose the model's behavior into a set of *often* causally relevant, interpretable directions.

One appealing strategy, despite being in its early developmental phases, is to leverage such dictionaries to gather insights into model behaviors is that of analysing latents that *fire* strongly in the presence or generation of specific text (e.g., Templeton et al., 2024 identify features that activate strongly on references to specific countries, math operations, code errors, and emotions.) A notable example was the limited-time release of Golden Gate Claude (Anthropic, 2024), a version of Claude 3

7: For example, simple queries can often be handled by small, efficient models, while complex tasks typically require larger, more capable models.

8: For example, the notation in Ong et al., 2024 explicitly assumes this—routing is performed between *weaker* and *stronger* models, where weaker and stronger is associated with model size.

9: For example, a simple (greedy) approach of routing to the smaller models first until quality is reasonable can be more efficient and cost-effective than resorting to a larger model.

10: The residual stream perspective views the Transformer network as a series of "read"/"write" operations on a central information channel.

11: For instance, when applied to a LLM, an SAE consists of an encoder that maps model activations to a higher-dimensional feature space, followed by a decoder that reconstructs the original activations. The training process optimizes for both reconstruction accuracy and sparsity of feature activations.

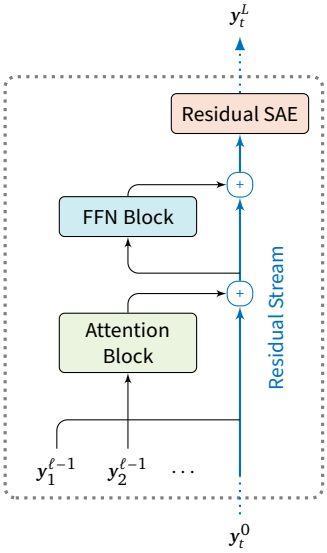


Figure 9.1: Illustration of a Transformer model represented as modules writing into and reading from the residual stream, with the addition of a sparse autoencoder on the post MLP residual stream.

Sonnet where features that had previously identified to activate strongly in the presence of text related to the Golden Gate Bridge are artificially clamped to increase their strength.[12] This upweighting, in turn, leads the model's responses to focus on the Golden Gate Bridge, even when this concept is not relevant at all to the input text (see Figure 9.1 for one such example). Note then that, through such strategies, it may be possible to effectively control the generation of text from LLMs in specific interpretable ways. For example, an immediate potential application involves identifying and amplifying features associated with *reliability*, *safety*.[13] As such, an interesting question is whether such behaviors could be mitigated if these features were known in advance. Moreover, another scenario aligned with the topics of our thesis is that of training a quality-aware translation model—a model that is trained with source and target samples, alongside their associated quality scores (Tomani et al., 2024). By identifying and upweighing features that fire for high-quality scores, we might be able to steer the model towards consistently generating high-quality output.

12: In Templeton et al., 2024, the authors clamp the direction to 10x its maximum activation value.

13: Conversely, it is worth researching whether such strategies could also potentially be misused to induce unsafe behavior in a system.

Bibliography

Here are the references in chronological order.

Cohen, Jacob (1960). 'A Coefficient of Agreement for Nominal Scales'. In: *Educational and Psychological Measurement* 20.1, pp. 37–46. DOI: 10.1177/001316446002000104 (cited on page 43).

Viterbi, A. (1967). 'Error bounds for convolutional codes and an asymptotically optimum decoding algorithm'. In: *IEEE Transactions on Information Theory* 13.2, pp. 260–269. DOI: 10.1109/TIT.1967.1054010 (cited on page 86).

Rabiner, L.R. (1989). 'A tutorial on hidden Markov models and selected applications in speech recognition'. In: *Proceedings of the IEEE* 77.2, pp. 257–286. DOI: 10.1109/5.18626 (cited on page 86).

Williams, R. J. (1992). 'Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning'. In: *Machine Learning* 8, pp. 229–256 (cited on pages 7, 83, 85).

Hochreiter, Sepp and Jürgen Schmidhuber (Dec. 1997). 'Long Short-term Memory'. In: *Neural computation* 9, pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735 (cited on page 20).

Kumar, Shankar and William Byrne (2002). 'Minimum Bayes-Risk word alignments of bilingual texts'. In: *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*. EMNLP '02. USA: Association for Computational Linguistics, pp. 140–147. DOI: 10.3115/1118693.1118712 (cited on page 117).

Papineni, Kishore et al. (July 2002). 'Bleu: a Method for Automatic Evaluation of Machine Translation'. In: *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, pp. 311–318. DOI: 10.3115/1073083.1073135 (cited on pages 8, 26, 30, 67).

Guyon, Isabelle and André Elisseeff (Mar. 2003). 'An Introduction to Variable and Feature Selection'. In: *J. Mach. Learn. Res.* 3.null, pp. 1157–1182 (cited on page 7).

Kumar, Shankar and William Byrne (May 2004). 'Minimum Bayes-Risk Decoding for Statistical Machine Translation'. In: *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*. Boston, Massachusetts, USA: Association for Computational Linguistics, pp. 169–176 (cited on page 117).

Shen, Libin, Anoop Sarkar, and Franz Josef Och (May 2004). 'Discriminative Reranking for Machine Translation'. In: *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*. Boston, Massachusetts, USA: Association for Computational Linguistics, pp. 177–184 (cited on page 50).

Banerjee, Satanjeev and Alon Lavie (June 2005). 'METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments'. In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Ann Arbor, Michigan: Association for Computational Linguistics, pp. 65–72 (cited on page 41).

Del Corso, Gianna, Antonio Gulli, and Francesco Romani (Jan. 2005). 'Ranking a stream of news'. In: pp. 97–106. DOI: 10.1145/1060745.1060764 (cited on page 181).

Koehn, Philipp (Sept. 2005). 'Europarl: A Parallel Corpus for Statistical Machine Translation'. In: *Proceedings of Machine Translation Summit X: Papers*. Phuket, Thailand, pp. 79–86 (cited on page 173).

Kantorovich, Leonid V (2006). 'On the translocation of masses'. In: *Journal of mathematical sciences* 133.4, pp. 1381–1382 (cited on pages 10, 53, 54).

Snover, Matthew et al. (2006). 'A study of translation edit rate with targeted human annotation'. In: *In Proceedings of Association for Machine Translation in the Americas*, pp. 223–231 (cited on page 25).

Vilar, David et al. (May 2006). 'Error Analysis of Statistical Machine Translation Output'. In: *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*. Genoa, Italy: European Language Resources Association (ELRA) (cited on page 39).

Zaidan, Omar F., Jason Eisner, and Christine Piatko (Apr. 2007). 'Using "Annotator Rationales" to Improve Machine Learning for Text Categorization'. In: *NAACL HLT 2007; Proceedings of the Main Conference*, pp. 260–267 (cited on page 15).

Wainwright, Martin J and Michael Irwin Jordan (2008). *Graphical models, exponential families, and variational inference*. Now Publishers Inc (cited on page 85).

Villani, Cédric (2009). *Optimal transport: old and new*. Vol. 338. Springer (cited on page 56).

Koo, Terry et al. (Oct. 2010). 'Dual Decomposition for Parsing with Non-Projective Head Automata'. In: *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Cambridge, MA: Association for Computational Linguistics, pp. 1288–1298 (cited on page 85).

Specia, Lucia, Dhwaj Raj, and Marco Turchi (2010). 'Machine translation evaluation versus quality estimation'. In: *Machine translation* 24, pp. 39–50 (cited on page 28).

Wang, Hongning, Yue Lu, and Chengxiang Zhai (July 2010). 'Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach'. In: pp. 783–792. DOI: 10.1145/1835804.1835903 (cited on page 181).

Maas, Andrew L. et al. (June 2011). 'Learning Word Vectors for Sentiment Analysis'. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Portland, Oregon, USA: Association for Computational Linguistics, pp. 142–150 (cited on page 181).

Papandreou, George and Alan L. Yuille (2011). 'Perturb-and-MAP random fields: Using discrete optimization to learn and sample from energy models'. In: *2011 International Conference on Computer Vision*, pp. 193–200. DOI: 10.1109/ICCV.2011.6126242 (cited on page 85).

Ravi, Sujith and Kevin Knight (June 2011). 'Deciphering Foreign Language'. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Ed. by Dekang Lin, Yuji Matsumoto, and Rada Mihalcea. Portland, Oregon, USA: Association for Computational Linguistics, pp. 12–21 (cited on page 9).

McAuley, Julian, Jure Leskovec, and Dan Jurafsky (2012). *Learning Attitudes and Attributes from Multi-Aspect Reviews* (cited on page 181).

Cuturi, Marco (2013). 'Sinkhorn Distances: Lightspeed Computation of Optimal Transport'. In: *Advances in Neural Information Processing Systems*. Ed. by C. J. C. Burges et al. Vol. 26. NIPS'13. Lake Tahoe, Nevada: Curran Associates, Inc., pp. 2292–2300 (cited on page 87).

Graham, Yvette et al. (Aug. 2013). 'Continuous Measurement Scales in Human Evaluation of Machine Translation'. In: *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*. Sofia, Bulgaria: Association for Computational Linguistics, pp. 33–41 (cited on page 25).

Kalchbrenner, Nal and Phil Blunsom (Oct. 2013). 'Recurrent Continuous Translation Models'. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, pp. 1700–1709 (cited on pages 9, 17).

Socher, Richard et al. (Oct. 2013). 'Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank'. In: *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics, pp. 1631–1642 (cited on page 181).

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2014). 'Neural Machine Translation by Jointly Learning to Align and Translate'. In: *CoRR* abs/1409.0473 (cited on page 3).

Cho, Kyunghyun et al. (Oct. 2014). 'On the Properties of Neural Machine Translation: Encoder–Decoder Approaches'. In: *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar: Association for Computational Linguistics, pp. 103–111. DOI: 10.3115/v1/W14-4012 (cited on pages 9, 17).

Kingma, Diederik P and Max Welling (2014). *Auto-Encoding Variational Bayes* (cited on pages 83, 85).

Lommel, Arle, Aljoscha Burchardt, and Hans Uszkoreit (Dec. 2014a). 'Multidimensional Quality Metrics (MQM): A Framework for Declaring and Describing Translation Quality Metrics'. In: *Tradumàtica: tecnologies de la traducció* 0, pp. 455–463. DOI: 10.5565/rev/tradumatica.77 (cited on page 26).

– (Dec. 2014b). 'Multidimensional Quality Metrics (MQM): A Framework for Declaring and Describing Translation Quality Metrics'. In: *Tradumàtica: tecnologies de la traducció* 0, pp. 455–463. DOI: 10.5565/rev/tradumatica.77 (cited on page 38).

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning (2014). 'GloVe: Global Vectors for Word Representation'. In: *Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543 (cited on page 182).

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le (2014). 'Sequence to Sequence Learning with Neural Networks'. In: *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*. NIPS'14. Montreal, Canada: MIT Press, pp. 3104–3112 (cited on pages 9, 17).

Bahdanau, Dzmitry, Kyung Hyun Cho, and Yoshua Bengio (Jan. 2015). 'Neural machine translation by jointly learning to align and translate'. English (US). In: 3rd International Conference on Learning

Representations, ICLR 2015 ; Conference date: 07-05-2015 Through 09-05-2015 (cited on pages 3, 7, 9, 19, 20, 24).

Bowman, Samuel R. et al. (2015). *A large annotated corpus for learning natural language inference* (cited on pages 89, 181).

Denil, Misha, Alban Demiraj, and Nando de Freitas (2015). *Extraction of Salient Sentences from Labelled Documents* (cited on page 7).

Martins, André F.T. et al. (2015). 'Ad3: Alternating directions dual decomposition for map inference in graphical models'. In: *The Journal of Machine Learning Research* 16.1, pp. 495–545 (cited on pages 85, 183).

Popović, Maja (Sept. 2015). 'chrF: character n-gram F-score for automatic MT evaluation'. In: *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbon, Portugal: Association for Computational Linguistics, pp. 392–395. DOI: 10.18653/v1/W15-3049 (cited on pages 8, 26).

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E. Hinton (2016). *Layer Normalization* (cited on page 20).

Firat, Orhan, Kyunghyun Cho, and Yoshua Bengio (June 2016). 'Multi-Way, Multilingual Neural Machine Translation with a Shared Attention Mechanism'. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by Kevin Knight, Ani Nenkova, and Owen Rambow. San Diego, California: Association for Computational Linguistics, pp. 866–875. DOI: 10.18653/v1/N16-1101 (cited on page 9).

Gal, Yarin and Zoubin Ghahramani (June 2016). 'Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning'. In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, pp. 1050–1059 (cited on pages 37, 41).

He, Kaiming et al. (2016). 'Deep Residual Learning for Image Recognition'. In: pp. 770–778. DOI: 10.1109/CVPR.2016.90 (cited on page 20).

Joulin, Armand et al. (2016). 'FastText.zip: Compressing text classification models'. In: *arXiv preprint arXiv:1612.03651* (cited on page 69).

Lee, Peter (Mar. 2016). *Learning from Tay's introduction*. Accessed: 2023-04-25. URL: https://blogs.microsoft.com/blog/2016/03/25/learning-tays-introduction/ (cited on page 5).

Lei, Tao, Regina Barzilay, and Tommi Jaakkola (2016). 'Rationalizing Neural Predictions'. In: *Proc. EMNLP*, pp. 107–117 (cited on pages 7, 15, 16, 84, 88, 181, 182).

Li, Jiwei et al. (June 2016). 'Visualizing and Understanding Neural Models in NLP'. In: *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics, pp. 681–691. DOI: 10.18653/v1/N16-1082 (cited on page 7).

Martins, André F. T. and Ramón Fernandez Astudillo (2016). 'From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification'. In: *CoRR* abs/1602.02068 (cited on pages 7, 8, 16, 24, 88).

Popović, Maja (Aug. 2016). 'chrF deconstructed: beta parameters and n-gram weights'. In: *Proceedings of the First Conference on Machine*

Translation: Volume 2, Shared Task Papers. Berlin, Germany: Association for Computational Linguistics, pp. 499–504. DOI: 10.18653/v1/W16-2341 (cited on page 54).

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier'. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: Association for Computing Machinery, pp. 1135–1144. DOI: 10.1145/2939672.2939778 (cited on page 7).

Rocktäschel, Tim et al. (2016). 'Reasoning about Entailment with Neural Attention'. In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun (cited on pages 7, 24).

Sennrich, Rico, Barry Haddow, and Alexandra Birch (Aug. 2016a). 'Improving Neural Machine Translation Models with Monolingual Data'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 86–96. DOI: 10.18653/v1/P16-1009 (cited on pages 9, 18).

– (Aug. 2016b). 'Neural Machine Translation of Rare Words with Subword Units'. In: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics, pp. 1715–1725. DOI: 10.18653/v1/P16-1162 (cited on page 41).

Bojar, Ondřej et al. (Sept. 2017). 'Findings of the 2017 Conference on Machine Translation (WMT17)'. In: *Proceedings of the Second Conference on Machine Translation*. Copenhagen, Denmark: Association for Computational Linguistics, pp. 169–214. DOI: 10.18653/v1/W17-4717 (cited on page 99).

Chen, Qian et al. (July 2017). 'Enhanced LSTM for Natural Language Inference'. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, pp. 1657–1668. DOI: 10.18653/v1/P17-1152 (cited on pages 87, 89, 181).

Gehring, Jonas et al. (2017). 'Convolutional Sequence to Sequence Learning'. In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. ICML'17. Sydney, NSW, Australia: JMLR.org, pp. 1243–1252 (cited on page 20).

Hern, Alex (Oct. 2017). *Facebook translates 'good morning' into 'attack them', leading to arrest*. Accessed: 2023-04-25. URL: https://www.theguardian.com/technology/2017/oct/24/facebook-palestine-israel-translates-good-morning-attack-them-arrest (cited on page 6).

Jang, Eric, Shixiang Gu, and Ben Poole (2017). *Categorical Reparameterization with Gumbel-Softmax* (cited on pages 83, 85, 88).

Kim, Yoon et al. (2017). 'Structured Attention Networks'. In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net (cited on page 20).

Kingma, Diederik P. and Jimmy Ba (2017). *Adam: A Method for Stochastic Optimization* (cited on page 182).

Koehn, Philipp and Rebecca Knowles (Aug. 2017). 'Six Challenges for Neural Machine Translation'. In: *Proceedings of the First Workshop*

on Neural Machine Translation. Vancouver: Association for Computational Linguistics, pp. 28–39. DOI: 10.18653/v1/W17-3204 (cited on page 39).

Li, Jiwei, Will Monroe, and Dan Jurafsky (2017). *Understanding Neural Networks through Representation Erasure* (cited on page 7).

Sundararajan, Mukund, Ankur Taly, and Qiqi Yan (2017). 'Axiomatic Attribution for Deep Networks'. In: *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. ICML'17. Sydney, NSW, Australia: JMLR.org, pp. 3319–3328 (cited on page 7).

Vaswani, Ashish et al. (2017). 'Attention is All you Need'. In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc. (cited on pages xi, 3, 9, 20, 21, 41, 66).

Artetxe, Mikel et al. (2018). *Unsupervised Neural Machine Translation*. URL: https://arxiv.org/abs/1710.11041 (cited on page 9).

Bao, Yujia et al. (2018). *Deriving Machine Attention from Human Rationales* (cited on page 181).

Bojar, Ondřej et al. (Oct. 2018). 'Findings of the 2018 Conference on Machine Translation (WMT18)'. In: *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*. Belgium, Brussels: Association for Computational Linguistics, pp. 272–303. DOI: 10.18653/v1/W18-6401 (cited on pages 41, 58).

Edunov, Sergey et al. (Oct. 2018). 'Understanding Back-Translation at Scale'. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 489–500. DOI: 10.18653/v1/D18-1045 (cited on page 18).

Fan, Angela, Mike Lewis, and Yann Dauphin (July 2018). 'Hierarchical Neural Story Generation'. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Iryna Gurevych and Yusuke Miyao. Melbourne, Australia: Association for Computational Linguistics, pp. 889–898. DOI: 10.18653/v1/P18-1082 (cited on page 116).

Feng, Shi et al. (Oct. 2018). 'Pathologies of Neural Models Make Interpretations Difficult'. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 3719–3728. DOI: 10.18653/v1/D18-1407 (cited on page 7).

Kudo, Taku and John Richardson (Nov. 2018). 'SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing'. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Brussels, Belgium: Association for Computational Linguistics, pp. 66–71. DOI: 10.18653/v1/D18-2012 (cited on page 41).

Lample, Guillaume et al. (2018). 'Unsupervised Machine Translation Using Monolingual Corpora Only'. In: *International Conference on Learning Representations* (cited on page 9).

Lee, Katherine et al. (2018a). 'Hallucinations in Neural Machine Translation'. In: (cited on pages 28, 29, 32, 65, 67).

– (2018b). 'Hallucinations in neural machine translation'. In: (cited on pages 6, 29, 38–40, 55).

Martindale, Marianna and Marine Carpuat (Mar. 2018). 'Fluency Over Adequacy: A Pilot Study in Measuring User Trust in Imperfect MT'. In: *Proceedings of the 13th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*. Boston, MA:

Association for Machine Translation in the Americas, pp. 13–25 (cited on page 6).
Niculae, Vlad et al. (2018). *SparseMAP: Differentiable Sparse Structured Inference* (cited on page 85).
Peyré, Gabriel and Marco Cuturi (2018). 'Computational Optimal Transport'. In: DOI: 10.48550/ARXIV.1803.00567 (cited on page 57).
Post, Matt (Oct. 2018). 'A Call for Clarity in Reporting BLEU Scores'. In: *Proceedings of the Third Conference on Machine Translation: Research Papers*. Brussels, Belgium: Association for Computational Linguistics, pp. 186–191. DOI: 10.18653/v1/W18-6319 (cited on pages 42, 67).
Radford, Alec and Karthik Narasimhan (2018). 'Improving Language Understanding by Generative Pre-Training'. In: (cited on pages 3, 23, 24, 66).
Sánchez-Cartagena, Víctor M. et al. (Oct. 2018). 'Prompsit's submission to WMT 2018 Parallel Corpus Filtering shared task'. In: *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*. Brussels, Belgium: Association for Computational Linguistics (cited on page 42).
Ugawa, Arata et al. (Aug. 2018). 'Neural Machine Translation Incorporating Named Entity'. In: *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics, pp. 3240–3250 (cited on page 39).
Voita, Elena et al. (July 2018). 'Context-Aware Neural Machine Translation Learns Anaphora Resolution'. In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics, pp. 1264–1274. DOI: 10.18653/v1/P18-1117 (cited on pages 24, 25, 47).
Aharoni, Roee, Melvin Johnson, and Orhan Firat (June 2019). 'Massively Multilingual Neural Machine Translation'. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 3874–3884. DOI: 10.18653/v1/N19-1388 (cited on page 3).
Arivazhagan, Naveen et al. (2019). *Massively Multilingual Neural Machine Translation in the Wild: Findings and Challenges*. DOI: 10.48550/ARXIV.1907.05019. URL: https://arxiv.org/abs/1907.05019 (cited on pages 3, 9, 18).
Bastings, Jasmijn, Wilker Aziz, and Ivan Titov (2019). 'Interpretable Neural Predictions with Differentiable Binary Variables'. In: *Proc. ACL* (cited on pages 7, 15, 16, 84, 88, 89, 182).
Berard, Alexandre, Ioan Calapodescu, and Claude Roux (Aug. 2019). 'Naver Labs Europe's Systems for the WMT19 Machine Translation Robustness Task'. In: *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*. Florence, Italy: Association for Computational Linguistics, pp. 526–532. DOI: 10.18653/v1/W19-5361 (cited on pages 25, 30–32, 39, 40, 46, 53, 55, 56, 58).
Clark, Kevin et al. (Aug. 2019). 'What Does BERT Look at? An Analysis of BERT's Attention'. In: *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*.

Florence, Italy: Association for Computational Linguistics, pp. 276–286. DOI: 10.18653/v1/W19-4828 (cited on pages 24, 25).

Conneau, Alexis et al. (2019). *Unsupervised Cross-lingual Representation Learning at Scale*. DOI: 10.48550/ARXIV.1911.02116. URL: https://arxiv.org/abs/1911.02116 (cited on page 24).

Correia, Gonçalo M., Vlad Niculae, and André F. T. Martins (Nov. 2019). 'Adaptively Sparse Transformers'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 2174–2184. DOI: 10.18653/v1/D19-1223 (cited on pages 21, 24).

Corro, Caio and Ivan Titov (2019a). *Differentiable Perturb-and-Parse: Semi-Supervised Parsing with a Structured Variational Autoencoder* (cited on page 85).

– (2019b). *Learning Latent Trees with Stochastic Perturbations and Differentiable Dynamic Programming* (cited on page 85).

Devlin, Jacob et al. (June 2019). 'BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding'. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: 10.18653/v1/N19-1423 (cited on pages 3, 17, 22, 24, 59).

DeYoung, Jay et al. (2019). 'ERASER: A Benchmark to Evaluate Rationalized NLP Models'. In: *arXiv preprint arXiv:1911.03429* (cited on page 88).

Dinan, Emily et al. (Nov. 2019). 'Build it Break it Fix it for Dialogue Safety: Robustness from Adversarial Human Attack'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 4537–4546. DOI: 10.18653/v1/D19-1461 (cited on page 5).

Falke, Tobias et al. (July 2019). 'Ranking Generated Summaries by Correctness: An Interesting but Challenging Application for Natural Language Inference'. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 2214–2220. DOI: 10.18653/v1/P19-1213 (cited on page 28).

Fonseca, Erick et al. (Aug. 2019). 'Findings of the WMT 2019 Shared Tasks on Quality Estimation'. In: *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*. Florence, Italy: Association for Computational Linguistics, pp. 1–10. DOI: 10.18653/v1/W19-5401 (cited on page 95).

Jain, Sarthak and Byron C. Wallace (June 2019). 'Attention is not Explanation'. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 3543–3556. DOI: 10.18653/v1/N19-1357 (cited on pages 7, 24, 25, 47).

Koehn, Philipp et al. (Aug. 2019). 'Findings of the WMT 2019 Shared Task on Parallel Corpus Filtering for Low-Resource Conditions'. In:

Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2). Florence, Italy: Association for Computational Linguistics, pp. 54–72. DOI: 10.18653/v1/W19-5404 (cited on page 173).

Kumar, Aviral and Sunita Sarawagi (2019). 'Calibration of Encoder Decoder Models for Neural Machine Translation'. In: *CoRR* abs/1903.00802 (cited on page 39).

Liu, Yinhan et al. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach* (cited on page 24).

McCoy, R. Thomas, Ellie Pavlick, and Tal Linzen (2019). *Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference* (cited on pages 89, 91, 181).

Niculae, Vlad and Mathieu Blondel (2019). *A Regularized Framework for Sparse and Structured Neural Attention* (cited on page 88).

Ott, Myle et al. (June 2019). 'fairseq: A Fast, Extensible Toolkit for Sequence Modeling'. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 48–53. DOI: 10.18653/v1/N19-4009 (cited on page 42).

Paty, François-Pierre and Marco Cuturi (2019). 'Subspace robust Wasserstein distances'. In: *International conference on machine learning*. PMLR, pp. 5072–5081 (cited on page 54).

Peters, Ben, Vlad Niculae, and André F. T. Martins (July 2019). 'Sparse Sequence-to-Sequence Models'. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 1504–1519. DOI: 10.18653/v1/P19-1146 (cited on pages 8, 16).

Peyré, Gabriel, Marco Cuturi, et al. (2019). 'Computational optimal transport: With applications to data science'. In: *Foundations and Trends® in Machine Learning* 11.5-6, pp. 355–607 (cited on pages 10, 53, 54).

Radford, Alec et al. (2019). 'Language Models are Unsupervised Multitask Learners'. In: (cited on pages 3, 18, 22, 24, 66).

Serrano, Sofia and Noah A. Smith (July 2019). 'Is Attention Interpretable?' In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 2931–2951. DOI: 10.18653/v1/P19-1282 (cited on pages 7, 24, 25, 47).

Stahlberg, Felix and Bill Byrne (Nov. 2019). 'On NMT Search Errors and Model Errors: Cat Got Your Tongue?' In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 3356–3362. DOI: 10.18653/v1/D19-1331 (cited on page 39).

Vashishth, Shikhar et al. (2019). *Attention Interpretability Across NLP Tasks*. DOI: 10.48550/ARXIV.1909.11218. URL: https://arxiv.org/abs/1909.11218 (cited on page 42).

Vig, Jesse and Yonatan Belinkov (Aug. 2019). 'Analyzing the Structure of Attention in a Transformer Language Model'. In: *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Florence, Italy: Association for Computational

Linguistics, pp. 63–76. DOI: 10.18653/v1/W19-4808 (cited on pages 24, 25).

Voita, Elena et al. (July 2019). 'Analyzing Multi-Head Self-Attention: Specialized Heads Do the Heavy Lifting, the Rest Can Be Pruned'. In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, pp. 5797–5808. DOI: 10.18653/v1/P19-1580 (cited on pages 7, 21, 24, 25, 47).

Wallace, Eric et al. (Nov. 2019). 'Universal Adversarial Triggers for Attacking and Analyzing NLP'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 2153–2162. DOI: 10.18653/v1/D19-1221 (cited on page 5).

Wiegreffe, Sarah and Yuval Pinter (Nov. 2019). 'Attention is not not Explanation'. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics, pp. 11–20. DOI: 10.18653/v1/D19-1002 (cited on pages 7, 24, 47).

Yu, Mo et al. (2019). 'Rethinking Cooperative Rationalization: Introspective Extraction and Complement Control'. In: *Proc. EMNLP-IJCNLP*, pp. 4085–4094 (cited on page 181).

Zhang, Mike and Antonio Toral (2019). 'The Effect of Translationese in Machine Translation Test Sets'. In: *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*. Florence, Italy: Association for Computational Linguistics (cited on page 118).

Abnar, Samira and Willem Zuidema (July 2020). 'Quantifying Attention Flow in Transformers'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 4190–4197. DOI: 10.18653/v1/2020.acl-main.385 (cited on pages 7, 24).

Anastasopoulos, Antonios et al. (Dec. 2020). 'TICO-19: the Translation Initiative for COvid-19'. In: *Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020*. Online: Association for Computational Linguistics. DOI: 10.18653/v1/2020.nlpcovid19-2.5 (cited on pages 67, 175).

Bastings, Jasmijn and Katja Filippova (Nov. 2020). 'The elephant in the interpretability room: Why use attention as explanation when we have saliency methods?' In: *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*. Online: Association for Computational Linguistics, pp. 149–155. DOI: 10.18653/v1/2020.blackboxnlp-1.14 (cited on page 47).

Brown, Tom et al. (2020a). 'Language models are few-shot learners'. In: *Advances in neural information processing systems*. Curran Associates, Inc. (cited on page 9).

Brown, Tom B. et al. (2020b). *Language Models are Few-Shot Learners*. DOI: 10.48550/ARXIV.2005.14165. URL: https://arxiv.org/abs/2005.14165 (cited on pages 3, 18, 22, 24, 66).

Chang, Shiyu et al. (2020). 'Invariant Rationalization'. In: (cited on pages 7, 16, 83).

Eikema, Bryan and Wilker Aziz (Dec. 2020). 'Is MAP Decoding All You Need? The Inadequacy of the Mode in Neural Machine Translation'.

In: *Proceedings of the 28th International Conference on Computational Linguistics*. Ed. by Donia Scott, Nuria Bel, and Chengqing Zong. Barcelona, Spain (Online): International Committee on Computational Linguistics, pp. 4506–4520. DOI: 10.18653/v1/2020.coling-main.398 (cited on page 117).

Fan, Angela et al. (2020). *Beyond English-Centric Multilingual Machine Translation*. DOI: 10.48550/ARXIV.2010.11125. URL: https://arxiv.org/abs/2010.11125 (cited on pages xiii, 3, 9, 18, 24, 65–67, 70, 175).

Feng, Fangxiaoyu et al. (2020). *Language-agnostic BERT Sentence Embedding*. DOI: 10.48550/ARXIV.2007.01852. URL: https://arxiv.org/abs/2007.01852 (cited on page 59).

Fomicheva, Marina et al. (2020). 'Unsupervised Quality Estimation for Neural Machine Translation'. In: *Transactions of the Association for Computational Linguistics* 8, pp. 539–555. DOI: 10.1162/tacl_a_00330 (cited on pages 40, 41, 47).

Holtzman, Ari et al. (2020). 'The Curious Case of Neural Text Degeneration'. In: *International Conference on Learning Representations* (cited on pages 116, 119).

Jacovi, Alon and Yoav Goldberg (July 2020). 'Towards Faithfully Interpretable NLP Systems: How Should We Define and Evaluate Faithfulness?' In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 4198–4205. DOI: 10.18653/v1/2020.acl-main.386 (cited on pages 7, 15).

Jain, Sarthak et al. (2020). 'Learning to Faithfully Rationalize by Construction'. In: *arXiv preprint arXiv:2005.00115* (cited on pages 7, 16, 83, 88, 89, 181, 182).

Kaplan, Jared et al. (2020). *Scaling Laws for Neural Language Models*. URL: https://arxiv.org/abs/2001.08361 (cited on page 3).

Kobayashi, Goro et al. (Nov. 2020). 'Attention is Not Only a Weight: Analyzing Transformers with Vector Norms'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 7057–7075. DOI: 10.18653/v1/2020.emnlp-main.574 (cited on page 24).

Lewis, Mike et al. (July 2020). 'BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 7871–7880. DOI: 10.18653/v1/2020.acl-main.703 (cited on pages 31, 41).

Mathur, Nitika et al. (Nov. 2020). 'Results of the WMT20 Metrics Shared Task'. In: *Proceedings of the Fifth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 688–725 (cited on pages 8, 39).

Müller, Mathias, Annette Rios, and Rico Sennrich (Oct. 2020). 'Domain Robustness in Neural Machine Translation'. In: *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*. Virtual: Association for Machine Translation in the Americas, pp. 151–164 (cited on pages 6, 31, 32, 58, 73).

Niculae, Vlad and André F. T. Martins (2020). *LP-SparseMAP: Differentiable Relaxed Optimization for Sparse Structured Prediction* (cited on pages 83–87, 89, 183).

Paranjape, Bhargavi et al. (Nov. 2020). 'An Information Bottleneck Approach for Controlling Conciseness in Rationale Extraction'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics (cited on pages 7, 16, 83, 182).

Pruthi, Danish et al. (July 2020). 'Learning to Deceive with Attention-Based Explanations'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 4782–4793. DOI: 10.18653/v1/2020.acl-main.432 (cited on page 47).

Raffel, Colin et al. (2020a). 'Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer'. In: *J. Mach. Learn. Res.* 21.1 (cited on pages 3, 22, 24).

– (Jan. 2020b). 'Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer'. In: *J. Mach. Learn. Res.* 21.1 (cited on page 3).

Ramírez-Sánchez, Gema et al. (Nov. 2020). 'Bifixer and Bicleaner: two open-source tools to clean your parallel data.' In: *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*. Lisboa, Portugal: European Association for Machine Translation, pp. 291–298 (cited on page 42).

Rei, Ricardo et al. (Nov. 2020a). 'COMET: A Neural Framework for MT Evaluation'. In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, pp. 2685–2702. DOI: 10.18653/v1/2020.emnlp-main.213 (cited on pages 8, 27, 37, 40, 54, 56).

– (Nov. 2020b). 'Unbabel's Participation in the WMT20 Metrics Shared Task'. In: *Proceedings of the Fifth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 911–920 (cited on pages 37, 39, 77).

Riley, Parker et al. (2020). 'Translationese as a Language in "Multilingual" NMT'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics (cited on page 118).

Sellam, Thibault, Dipanjan Das, and Ankur Parikh (July 2020). 'BLEURT: Learning Robust Metrics for Text Generation'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 7881–7892. DOI: 10.18653/v1/2020.acl-main.704 (cited on pages 8, 27).

Shoeybi, Mohammad et al. (2020). *Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism* (cited on page 3).

Swanson, Kyle, Lili Yu, and Tao Lei (2020). *Rationalizing Text Matching: Learning Sparse Alignments via Optimal Transport* (cited on pages 15, 16, 87, 89, 92, 181).

Treviso, Marcos and André F. T. Martins (Nov. 2020). 'The Explanation Game: Towards Prediction Explainability through Sparse Communication'. In: *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*. Online: Association for Computational Linguistics, pp. 107–118. DOI: 10.18653/v1/2020.blackboxnlp-1.10 (cited on pages 7, 16, 83, 84, 182).

Voita, Elena (Sept. 2020). *NLP Course For You*. URL: https://lena-voita.github.io/nlp_course.html (cited on pages 19, 21).

Wang, Chaojun and Rico Sennrich (July 2020). 'On Exposure Bias, Hallucination and Domain Shift in Neural Machine Translation'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 3544–3552. DOI: 10.18653/v1/2020.acl-main.326 (cited on pages 6, 31, 32, 58, 73).

Wolf, Thomas et al. (2020). 'Datasets'. In: *GitHub. Note: https://github.com/huggingface/datasets* 1 (cited on page 181).

Zhang, Biao et al. (July 2020a). 'Improving Massively Multilingual Neural Machine Translation and Zero-Shot Translation'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, pp. 1628–1639. DOI: 10.18653/v1/2020.acl-main.148 (cited on page 3).

Zhang, Tianyi et al. (2020b). 'BERTScore: Evaluating Text Generation with BERT'. In: *International Conference on Learning Representations* (cited on page 8).

Zhang*, Tianyi et al. (2020). 'BERTScore: Evaluating Text Generation with BERT'. In: *International Conference on Learning Representations* (cited on page 41).

Akhbardeh, Farhad et al. (Nov. 2021). 'Findings of the 2021 Conference on Machine Translation (WMT21)'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 1–88 (cited on pages 9, 18).

Askell, Amanda et al. (2021). *A General Language Assistant as a Laboratory for Alignment* (cited on page 3).

Bender, Emily M. et al. (2021). 'On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?' In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. FAccT '21. Virtual Event, Canada: Association for Computing Machinery, pp. 610–623. DOI: 10.1145/3442188.3445922 (cited on pages 5, 6).

Carlini, Nicholas et al. (2021). *Extracting Training Data from Large Language Models* (cited on page 5).

Deutsch, Daniel, Rotem Dror, and Dan Roth (2021). 'A Statistical Analysis of Summarization Evaluation Metrics Using Resampling Methods'. In: *Transactions of the Association for Computational Linguistics* 9, pp. 1132–1146. DOI: 10.1162/tacl_a_00417 (cited on page 101).

Ethayarajh, Kawin and Dan Jurafsky (Aug. 2021). 'Attention Flows are Shapley Value Explanations'. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Online: Association for Computational Linguistics, pp. 49–54. DOI: 10.18653/v1/2021.acl-short.8 (cited on pages 7, 24).

Ferrando, Javier and Marta R. Costa-jussà (Nov. 2021). 'Attention Weights in Transformer NMT Fail Aligning Words Between Sequences but Largely Explain Model Predictions'. In: *Findings of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 434–443. DOI: 10.18653/v1/2021.findings-emnlp.39 (cited on page 24).

Freitag, Markus et al. (2021a). 'Experts, Errors, and Context: A Large-Scale Study of Human Evaluation for Machine Translation'. In: *Transactions of the Association for Computational Linguistics* 9, pp. 1460–1474. DOI: 10.1162/tacl_a_00437 (cited on pages xii, 8, 26, 100, 101).

Freitag, Markus et al. (Nov. 2021b). 'Results of the WMT21 Metrics Shared Task: Evaluating Metrics with Expert-based Human Evaluations on TED and News Domain'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 733–774 (cited on pages 8, 26, 27, 37, 39).

Goyal, Naman et al. (Aug. 2021). 'Larger-Scale Transformers for Multilingual Masked Language Modeling'. In: *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*. Online: Association for Computational Linguistics, pp. 29–33. DOI: 10.18653/v1/2021.repl4nlp-1.4 (cited on page 96).

Jacovi, Alon and Yoav Goldberg (Mar. 2021). 'Aligning Faithful Interpretations with their Social Attribution'. In: *Transactions of the Association for Computational Linguistics* 9, pp. 294–310. DOI: 10.1162/tacl_a_00367 (cited on page 15).

Kobayashi, Goro et al. (Nov. 2021). 'Incorporating Residual and Normalization Layers into Analysis of Masked Language Models'. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 4547–4568. DOI: 10.18653/v1/2021.emnlp-main.373 (cited on pages 7, 24).

Kocmi, Tom et al. (Nov. 2021a). 'To Ship or Not to Ship: An Extensive Evaluation of Automatic Metrics for Machine Translation'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 478–494 (cited on pages 8, 101, 102).

– (2021b). *To Ship or Not to Ship: An Extensive Evaluation of Automatic Metrics for Machine Translation*. DOI: 10.48550/ARXIV.2107.10821. URL: https://arxiv.org/abs/2107.10821 (cited on pages 37, 39, 40).

Lee, Ann, Michael Auli, and Marc'Aurelio Ranzato (Aug. 2021). 'Discriminative Reranking for Neural Machine Translation'. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, pp. 7250–7264. DOI: 10.18653/v1/2021.acl-long.563 (cited on page 50).

Li, Panpan, Mengxiang Wang, and Jian Wang (May 2021). 'Named Entity Translation Method Based on Machine Translation Lexicon'. In: *Neural Comput. Appl.* 33.9, pp. 3977–3985. DOI: 10.1007/s00521-020-05509-y (cited on page 39).

Marie, Benjamin, Atsushi Fujita, and Raphael Rubino (Aug. 2021). 'Scientific Credibility of Machine Translation Research: A Meta-Evaluation of 769 Papers'. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, pp. 7297–7306. DOI: 10.18653/v1/2021.acl-long.566 (cited on page 27).

Müller, Mathias and Rico Sennrich (Aug. 2021). 'Understanding the Properties of Minimum Bayes Risk Decoding in Neural Machine

Translation'. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online: Association for Computational Linguistics, pp. 259–272. DOI: 10.18653/v1/2021.acl-long.22 (cited on pages 6, 30, 31, 39, 40, 45).

Pu, Amy et al. (Nov. 2021). 'Learning Compact Metrics for MT'. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 751–762. DOI: 10.18653/v1/2021.emnlp-main.58 (cited on pages 100, 101).

Raunak, Vikas, Arul Menezes, and Marcin Junczys-Dowmunt (June 2021). 'The Curious Case of Hallucinations in Neural Machine Translation'. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, pp. 1172–1183. DOI: 10.18653/v1/2021.naacl-main.92 (cited on pages 6, 25, 28–32, 38–40, 43, 46–48, 53, 55, 65, 67, 68, 70–72, 99, 105, 172, 173).

Specia, Lucia et al. (Nov. 2021). 'Findings of the WMT 2021 Shared Task on Quality Estimation'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 684–725 (cited on page 173).

Staerman, Guillaume et al. (2021). 'When ot meets mom: Robust estimation of wasserstein distance'. In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 136–144 (cited on page 54).

Sudoh, Katsuhito, Kosuke Takahashi, and Satoshi Nakamura (Apr. 2021). 'Is This Translation Error Critical?: Classification-Based Human and Automatic Machine Translation Evaluation Focusing on Critical Errors'. In: *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*. Online: Association for Computational Linguistics, pp. 46–55 (cited on pages 8, 45).

Takahashi, Kosuke et al. (Nov. 2021). 'Multilingual Machine Translation Evaluation Metrics Fine-tuned on Pseudo-Negative Examples for WMT 2021 Metrics Task'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 1049–1052 (cited on pages 8, 45).

Tran, Chau et al. (Nov. 2021). 'Facebook AI's WMT21 News Translation Task Submission'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 205–215 (cited on page 41).

Treviso, Marcos et al. (Nov. 2021). 'IST-Unbabel 2021 Submission for the Explainable Quality Estimation Shared Task'. In: *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*. Ed. by Yang Gao et al. Punta Cana, Dominican Republic: Association for Computational Linguistics, pp. 133–145. DOI: 10.18653/v1/2021.eval4nlp-1.14 (cited on page 93).

Voita, Elena, Rico Sennrich, and Ivan Titov (Aug. 2021). 'Analyzing the Source and Target Contributions to Predictions in Neural Machine Translation'. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Online: Association for Computational Linguistics, pp. 1126–1140. DOI: 10.18653/v1/2021.acl-long.91 (cited on pages 6, 7, 31).

Wang, Yinan et al. (2021). 'WOOD: Wasserstein-based Out-of-Distribution Detection'. In: *arXiv preprint arXiv:2112.06384* (cited on page 54).

Weidinger, Laura et al. (2021). *Ethical and social risks of harm from Language Models* (cited on page 5).

Wenzek, Guillaume et al. (Nov. 2021). 'Findings of the WMT 2021 Shared Task on Large-Scale Multilingual Machine Translation'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 89–99 (cited on pages 3, 9, 18, 67).

Xu, Albert et al. (June 2021). 'Detoxifying Language Models Risks Marginalizing Minority Voices'. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, pp. 2390–2397. DOI: 10.18653/v1/2021.naacl-main.190 (cited on page 5).

Xue, Linting et al. (June 2021). 'mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer'. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, pp. 483–498. DOI: 10.18653/v1/2021.naacl-main.41 (cited on pages 24, 101).

Yan, Yongzhe et al. (2021). '2D Wasserstein loss for robust facial landmark detection'. In: *Pattern Recognition* 116, p. 107945 (cited on page 54).

Yu, Mo et al. (2021). 'Understanding Interlocking Dynamics of Cooperative Rationalization'. In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., pp. 12822–12835 (cited on page 92).

Zerva, Chrysoula et al. (Nov. 2021). 'IST-Unbabel 2021 Submission for the Quality Estimation Shared Task'. In: *Proceedings of the Sixth Conference on Machine Translation*. Online: Association for Computational Linguistics, pp. 961–972 (cited on pages 40, 41).

Zhang, Zijian, Koustav Rudra, and Avishek Anand (Mar. 2021). 'Explain and Predict, and then Predict Again'. In: *Proceedings of the 14th ACM International Conference on Web Search and Data Mining*. DOI: 10.1145/3437963.3441758 (cited on page 15).

Zhou, Chunting et al. (Aug. 2021). 'Detecting Hallucinated Content in Conditional Neural Sequence Generation'. In: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics, pp. 1393–1404. DOI: 10.18653/v1/2021.findings-acl.120 (cited on pages 6, 31, 38, 41, 47).

Alves, Duarte et al. (Dec. 2022). 'Robust MT Evaluation with Sentence-level Multilingual Augmentation'. In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 469–478 (cited on page 104).

Amrhein, Chantal and Rico Sennrich (2022a). *Identifying Weaknesses in Machine Translation Metrics Through Minimum Bayes Risk Decoding: A Case Study for COMET*. DOI: 10.48550/ARXIV.2202.05148. URL: https://arxiv.org/abs/2202.05148 (cited on pages 8, 48, 131).

– (Nov. 2022b). 'Identifying Weaknesses in Machine Translation Metrics Through Minimum Bayes Risk Decoding: A Case Study for COMET'. In: *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Online only: Association for Computational Linguistics, pp. 1125–1141 (cited on pages 99, 105).

Bapna, Ankur et al. (2022). *Building Machine Translation Systems for the Next Thousand Languages*. DOI: 10.48550/ARXIV.2205.03983. URL: https://arxiv.org/abs/2205.03983 (cited on pages 9, 18).

Bibal, Adrien et al. (May 2022). 'Is Attention Explanation? An Introduction to the Debate'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 3889–3900. DOI: 10.18653/v1/2022.acl-long.269 (cited on page 25).

Bommasani, Rishi et al. (2022). *On the Opportunities and Risks of Foundation Models* (cited on page 3).

Cao, Meng, Yue Dong, and Jackie Cheung (May 2022). 'Hallucinated but Factual! Inspecting the Factuality of Hallucinations in Abstractive Summarization'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 3340–3354. DOI: 10.18653/v1/2022.acl-long.236 (cited on page 28).

Chen, Howard et al. (July 2022). 'Can Rationalization Improve Robustness?' In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Ed. by Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz. Seattle, United States: Association for Computational Linguistics, pp. 3792–3805. DOI: 10.18653/v1/2022.naacl-main.278 (cited on page 92).

Cheng, Xiaoyu et al. (2022). 'Hyperspectral Anomaly Detection Based on Wasserstein Distance and Spatial Filtering'. In: *Remote Sensing* 14.12, p. 2730 (cited on page 54).

Chowdhery, Aakanksha et al. (2022a). *PaLM: Scaling Language Modeling with Pathways*. DOI: 10.48550/ARXIV.2204.02311. URL: https://arxiv.org/abs/2204.02311 (cited on pages 3, 18, 19, 24).

– (2022b). 'PaLM: Scaling Language Modeling with Pathways'. In: *arXiv preprint arXiv:2402.00786* (cited on page 9).

Chrysostomou, George and Nikolaos Aletras (May 2022). 'An Empirical Study on Explanations in Out-of-Domain Settings'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Smaranda Muresan, Preslav Nakov, and Aline Villavicencio. Dublin, Ireland: Association for Computational Linguistics, pp. 6920–6938. DOI: 10.18653/v1/2022.acl-long.477 (cited on page 92).

Colombo, Pierre et al. (2022). 'What are the best systems? new perspectives on nlp benchmarking'. In: *Advances in Neural Information Processing Systems*. New Orleans (cited on page 120).

Dale, David et al. (2022). *Detecting and Mitigating Hallucinations in Machine Translation: Model Internal Workings Alone Do Well, Sentence Similarity Even Better*. DOI: 10.48550/ARXIV.2212.08597. URL:

https://arxiv.org/abs/2212.08597 (cited on pages 6, 31, 51, 65, 67, 70, 71, 172).

Dhole, Kaustubh D. et al. (2022). *NL-Augmenter: A Framework for Task-Sensitive Natural Language Augmentation* (cited on page 67).

Federmann, Christian, Tom Kocmi, and Ying Xin (Nov. 2022). 'NTREX-128 – News Test References for MT Evaluation of 128 Languages'. In: *Proceedings of the First Workshop on Scaling Up Multilingual Evaluation*. Ed. by Kabir Ahuja et al. Online: Association for Computational Linguistics, pp. 21–24 (cited on page 118).

Feng, Fangxiaoyu et al. (May 2022). *Language-agnostic BERT Sentence Embedding*. Dublin, Ireland. DOI: 10.18653/v1/2022.acl-long.62. URL: https://aclanthology.org/2022.acl-long.62 (cited on pages 67, 99).

Fernandes, Patrick et al. (2022a). 'Learning to Scaffold: Optimizing Model Explanations for Teaching'. In: *Advances in Neural Information Processing Systems*. Ed. by S. Koyejo et al. Vol. 35. Curran Associates, Inc., pp. 36108–36122 (cited on page 92).

Fernandes, Patrick et al. (July 2022b). 'Quality-Aware Decoding for Neural Machine Translation'. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Seattle, United States: Association for Computational Linguistics, pp. 1396–1412 (cited on pages 28, 50, 115, 116, 119, 131).

Ferrando, Javier, Gerard I. Gállego, and Marta R. Costa-jussà (Dec. 2022a). 'Measuring the Mixing of Contextual Information in the Transformer'. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 8698–8714 (cited on pages 7, 24, 31).

Ferrando, Javier et al. (Dec. 2022b). *Towards Opening the Black Box of Neural Machine Translation: Source and Target Interpretations of the Transformer*. Abu Dhabi, United Arab Emirates. URL: https://aclanthology.org/2022.emnlp-main.599 (cited on pages 6, 7, 24, 25, 30, 32, 51, 53, 65, 67, 68, 70, 72, 172, 176).

Fomicheva, Marina et al. (June 2022). 'MLQE-PE: A Multilingual Quality Estimation and Post-Editing Dataset'. In: *Proceedings of the Thirteenth Language Resources and Evaluation Conference*. Marseille, France: European Language Resources Association, pp. 4963–4974 (cited on pages 58, 98, 173).

Freitag, Markus et al. (July 2022a). 'High Quality Rather than High Model Probability: Minimum Bayes Risk Decoding with Neural Metrics'. In: *Transactions of the Association for Computational Linguistics* 10, pp. 811–825. DOI: 10.1162/tacl_a_00491 (cited on page 50).

– (2022b). 'High Quality Rather than High Model Probability: Minimum Bayes Risk Decoding with Neural Metrics'. In: *Transactions of the Association for Computational Linguistics* 10. Ed. by Brian Roark and Ani Nenkova, pp. 811–825. DOI: 10.1162/tacl_a_00491 (cited on pages 116, 119).

Freitag, Markus et al. (Dec. 2022c). 'Results of WMT22 Metrics Shared Task: Stop Using BLEU – Neural Metrics Are Better and More Robust'. In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 46–68 (cited on pages 26–28, 101).

Goyal, Naman et al. (2022). 'The Flores-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation'. In: *Transactions of the Association for Computational Linguistics* 10, pp. 522–538. DOI: 10.1162/tacl_a_00474 (cited on pages xiii, 3, 18, 66, 67, 118, 175).

Hewitt, John, Christopher Manning, and Percy Liang (Dec. 2022). 'Truncation Sampling as Language Model Desmoothing'. In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 3414–3427. DOI: 10.18653/v1/2022.findings-emnlp.249 (cited on pages 116, 119).

Hoffmann, Jordan et al. (2022). *Training Compute-Optimal Large Language Models*. URL: https://arxiv.org/abs/2203.15556 (cited on page 3).

Hu, Junjie et al. (May 2022). 'DEEP: DEnoising Entity Pre-training for Neural Machine Translation'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 1753–1766. DOI: 10.18653/v1/2022.acl-long.123 (cited on page 39).

Huang, Jie, Hanyin Shao, and Kevin Chen-Chuan Chang (Dec. 2022). 'Are Large Pre-Trained Language Models Leaking Your Personal Information?' In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 2038–2047 (cited on page 5).

Ji, Ziwei et al. (2022). *Survey of Hallucination in Natural Language Generation*. DOI: 10.48550/ARXIV.2202.03629. URL: https://arxiv.org/abs/2202.03629 (cited on pages 28, 38).

Karpinska, Marzena et al. (Dec. 2022). 'DEMETR: Diagnosing Evaluation Metrics for Translation'. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 9540–9561 (cited on page 99).

Lee, Katherine et al. (May 2022). 'Deduplicating Training Data Makes Language Models Better'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 8424–8445. DOI: 10.18653/v1/2022.acl-long.577 (cited on page 48).

Lin, Stephanie, Jacob Hilton, and Owain Evans (May 2022a). 'TruthfulQA: Measuring How Models Mimic Human Falsehoods'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 3214–3252. DOI: 10.18653/v1/2022.acl-long.229 (cited on page 5).

Lin, Tianyang et al. (2022b). 'A survey of transformers'. In: *AI Open* 3, pp. 111–132. DOI: https://doi.org/10.1016/j.aiopen.2022.10.001 (cited on page 24).

Lin, Xi Victoria et al. (Dec. 2022c). 'Few-shot Learning with Multilingual Generative Language Models'. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Abu Dhabi, United Arab

Emirates: Association for Computational Linguistics, pp. 9019–9052. DOI: 10.18653/v1/2022.emnlp-main.616 (cited on page 9).

Modarressi, Ali et al. (July 2022). 'GlobEnc: Quantifying Global Token Attribution by Incorporating the Whole Encoder Layer in Transformers'. In: *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Seattle, United States: Association for Computational Linguistics, pp. 258–271. DOI: 10.18653/v1/2022.naacl-main.19 (cited on pages 7, 24).

Mohammadshahi, Alireza et al. (Dec. 2022). 'SMaLL-100: Introducing Shallow Multilingual Machine Translation Model for Low-Resource Languages'. In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 8348–8359 (cited on page 66).

NLLB Team et al. (2022). *No Language Left Behind: Scaling Human-Centered Machine Translation*. DOI: 10.48550/ARXIV.2207.04672. URL: https://arxiv.org/abs/2207.04672 (cited on pages 3, 9, 18, 24, 67, 70, 72, 75).

Nowakowski, Artur et al. (Dec. 2022). 'Adam Mickiewicz University at WMT 2022: NER-Assisted and Quality-Aware Neural Machine Translation'. In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Ed. by Philipp Koehn et al. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 326–334 (cited on page 119).

Ouyang, Long et al. (2022). *Training language models to follow instructions with human feedback*. DOI: 10.48550/ARXIV.2203.02155. URL: https://arxiv.org/abs/2203.02155 (cited on pages 3, 24, 66).

Perez, Ethan et al. (2022). 'Red teaming language models with language models'. In: *arXiv preprint arXiv:2202.03286* (cited on page 5).

Perrella, Stefano et al. (Dec. 2022). 'MaTESe: Machine Translation Evaluation as a Sequence Tagging Problem'. In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 569–577 (cited on pages 27, 95).

Raunak, Vikas, Matt Post, and Arul Menezes (2022a). *SALTED: A Framework for SAlient Long-Tail Translation Error Detection*. DOI: 10.48550/ARXIV.2205.09988. URL: https://arxiv.org/abs/2205.09988 (cited on pages 38, 48, 58, 70, 71).

– (Dec. 2022b). 'SALTED: A Framework for SAlient Long-tail Translation Error Detection'. In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 5163–5179 (cited on pages 99, 105).

Rei, Ricardo et al. (Dec. 2022a). 'COMET-22: Unbabel-IST 2022 Submission for the Metrics Shared Task'. In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 578–585 (cited on pages 67, 96, 101, 118).

Rei, Ricardo et al. (Dec. 2022b). *CometKiwi: IST-Unbabel 2022 Submission for the Quality Estimation Shared Task*. Abu Dhabi, United Arab Emirates (Hybrid). URL: https://aclanthology.org/2022.wmt-1.60 (cited on pages 28, 59, 67, 77, 96, 118).

Tang, Joël, Marina Fomicheva, and Lucia Specia (2022). *Reducing Hallucinations in Neural Machine Translation with Feature Attribution*. DOI: 10.48550/ARXIV.2211.09878. URL: https://arxiv.org/abs/2211.09878 (cited on pages 58, 173).

Thoppilan, Romal et al. (2022). *LaMDA: Language Models for Dialog Applications* (cited on page 24).

Vilar, David et al. (2022). *Prompting PaLM for Translation: Assessing Strategies and Performance*. DOI: 10.48550/ARXIV.2211.09102. URL: https://arxiv.org/abs/2211.09102 (cited on pages 3, 9, 19, 69).

Wan, Dazhen et al. (Dec. 2022a). 'A Unified Dialogue User Simulator for Few-shot Data Augmentation'. In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 3788–3799 (cited on page 100).

Wan, Yu et al. (May 2022b). 'UniTE: Unified Translation Evaluation'. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics, pp. 8117–8127. DOI: 10.18653/v1/2022.acl-long.558 (cited on pages 95, 97).

Xiao, Yisheng et al. (2022). 'A Survey on Non-Autoregressive Generation for Neural Machine Translation and Beyond'. In: *CoRR* abs/2204.09269. DOI: 10.48550/arXiv.2204.09269 (cited on page 17).

Yan, Jianhao, Fandong Meng, and Jie Zhou (2022). *Probing Causes of Hallucinations in Neural Machine Translations*. DOI: 10.48550/ARXIV.2206.12529. URL: https://arxiv.org/abs/2206.12529 (cited on page 40).

Zeng, Aohan et al. (2022). *GLM-130B: An Open Bilingual Pre-trained Model* (cited on page 24).

Zerva, Chrysoula et al. (Dec. 2022a). 'Findings of the WMT 2022 Shared Task on Quality Estimation'. In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Abu Dhabi, United Arab Emirates (Hybrid): Association for Computational Linguistics, pp. 69–99 (cited on page 26).

Zerva, Chrysoula et al. (Dec. 2022b). 'Findings of the WMT 2022 Shared Task on Quality Estimation'. In: *Proceedings of the Seventh Conference on Machine Translation*. Abu Dhabi: Association for Computational Linguistics, pp. 69–99 (cited on page 59).

Zhang, B. (Aug. 2022). 'Towards efficient universal neural machine translation'. PhD thesis. The University of Edinburgh (cited on page 21).

Zhang, Susan et al. (2022). *OPT: Open Pre-trained Transformer Language Models*. DOI: 10.48550/ARXIV.2205.01068. URL: https://arxiv.org/abs/2205.01068 (cited on pages 19, 24).

Zhao, Zhixue et al. (Dec. 2022). 'On the Impact of Temporal Concept Drift on Model Explanations'. In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Ed. by Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, pp. 4039–4054. DOI: 10.18653/v1/2022.findings-emnlp.298 (cited on page 92).

Alves, Duarte et al. (Dec. 2023). 'Steering Large Language Models for Machine Translation with Finetuning and In-Context Learning'. In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore:

Association for Computational Linguistics, pp. 11127–11148. DOI: 10.18653/v1/2023.findings-emnlp.744 (cited on page 115).

Bao, Keqin et al. (2023). *Towards Fine-Grained Information: Identifying the Type and Location of Translation Errors* (cited on pages 27, 95).

Bawden, Rachel and François Yvon (2023). *Investigating the Translation Performance of a Large Multilingual Language Model: the Case of BLOOM*. DOI: 10.48550/ARXIV.2303.01911. URL: https://arxiv.org/abs/2303.01911 (cited on page 19).

Biderman, Stella et al. (2023). *Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling* (cited on page 3).

Briakou, Eleftheria, Colin Cherry, and George Foster (2023). 'Searching for Needles in a Haystack: On the Role of Incidental Bilingualism in PaLM's Translation Capability'. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics* (cited on page 9).

Bricken, Trenton et al. (2023). 'Towards Monosemanticity: Decomposing Language Models With Dictionary Learning'. In: *Transformer Circuits Thread*. https://transformer-circuits.pub/2023/monosemantic-features/index.html (cited on page 133).

Cunningham, Hoagy et al. (2023). *Sparse Autoencoders Find Highly Interpretable Features in Language Models*. URL: https://arxiv.org/abs/2309.08600 (cited on page 133).

Dale, David et al. (2023). *HalOmi: A Manually Annotated Benchmark for Multilingual Hallucination and Omission Detection in Machine Translation* (cited on pages 52, 63).

Deutsch, Daniel, George Foster, and Markus Freitag (Dec. 2023a). 'Ties Matter: Meta-Evaluating Modern Metrics with Pairwise Accuracy and Tie Calibration'. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 12914–12929. DOI: 10.18653/v1/2023.emnlp-main.798 (cited on page 101).

Deutsch, Daniel et al. (Dec. 2023b). 'Training and Meta-Evaluating Machine Translation Evaluation Metrics at the Paragraph Level'. In: *Proceedings of the Eighth Conference on Machine Translation*. Singapore: Association for Computational Linguistics, pp. 994–1011 (cited on page 101).

Farinhas, António, José de Souza, and Andre Martins (Dec. 2023). 'An Empirical Study of Translation Hypothesis Ensembling with Large Language Models'. In: *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 11956–11970. DOI: 10.18653/v1/2023.emnlp-main.733 (cited on pages 116, 119).

Fernandes, Patrick et al. (Dec. 2023a). 'The Devil Is in the Errors: Leveraging Large Language Models for Fine-grained Machine Translation Evaluation'. In: *Proceedings of the Eighth Conference on Machine Translation*. Singapore: Association for Computational Linguistics, pp. 1064–1081 (cited on pages 27, 95, 101, 102).

Fernandes, Patrick et al. (Dec. 2023b). 'The Devil Is in the Errors: Leveraging Large Language Models for Fine-grained Machine Translation Evaluation'. In: *Proceedings of the Eighth Conference on Machine Translation*. Ed. by Philipp Koehn et al. Singapore: Association for Compu-

tational Linguistics, pp. 1066–1083. DOI: 10.18653/v1/2023.wmt-1.100 (cited on page 131).

Freitag, Markus, Behrooz Ghorbani, and Patrick Fernandes (2023a). *Epsilon Sampling Rocks: Investigating Sampling Strategies for Minimum Bayes Risk Decoding for Machine Translation*. URL: https://arxiv.org/abs/2305.09860 (cited on page 116).

– (Dec. 2023b). 'Epsilon Sampling Rocks: Investigating Sampling Strategies for Minimum Bayes Risk Decoding for Machine Translation'. In: *Findings of the Association for Computational Linguistics: EMNLP 2023*. Ed. by Houda Bouamor, Juan Pino, and Kalika Bali. Singapore: Association for Computational Linguistics, pp. 9198–9209. DOI: 10.18653/v1/2023.findings-emnlp.617 (cited on page 119).

Freitag, Markus et al. (Dec. 2023c). 'Results of WMT23 Metrics Shared Task: Metrics Might Be Guilty but References Are Not Innocent'. In: *Proceedings of the Eighth Conference on Machine Translation*. Ed. by Philipp Koehn et al. Singapore: Association for Computational Linguistics, pp. 578–628. DOI: 10.18653/v1/2023.wmt-1.51 (cited on pages 101, 109).

– (Dec. 2023d). 'Results of WMT23 Metrics Shared Task: Metrics Might Be Guilty but References Are Not Innocent'. In: *Proceedings of the Eighth Conference on Machine Translation*. Singapore: Association for Computational Linguistics (cited on page 120).

Fu, Jinlan et al. (2023). *GPTScore: Evaluate as You Desire*. DOI: 10.48550/ARXIV.2302.04166. URL: https://arxiv.org/abs/2302.04166 (cited on page 66).

Garcia, Xavier et al. (2023). *The unreasonable effectiveness of few-shot learning for machine translation*. URL: https://arxiv.org/abs/2302.01398 (cited on page 9).

Goldstein, Josh A. et al. (2023). *Generative Language Models and Automated Influence Operations: Emerging Threats and Potential Mitigations* (cited on page 5).

Guerreiro, Nuno M., Elena Voita, and André Martins (May 2023a). 'Looking for a Needle in a Haystack: A Comprehensive Study of Hallucinations in Neural Machine Translation'. In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Dubrovnik, Croatia: Association for Computational Linguistics, pp. 1059–1075 (cited on pages 28, 53, 54, 56, 58, 59, 62, 65, 69–72, 74, 99, 106, 107, 116, 167, 172, 173).

Guerreiro, Nuno M. et al. (2023b). 'Optimal Transport for Unsupervised Hallucination Detection in Neural Machine Translation'. In: *Proceedings of The 61st Annual Meeting of the Association for Computational Linguistics*. Toronto, Canada: Association for Computational Linguistics (cited on pages 65, 67, 71).

Guerreiro, Nuno M. et al. (2023c). 'xCOMET: Transparent Machine Translation Evaluation through Fine-grained Error Detection'. In: *arXiv preprint arXiv:2310.10482* (cited on pages 52, 93, 118, 119).

Gulcehre, Caglar et al. (2023). *Reinforced Self-Training (ReST) for Language Modeling* (cited on page 28).

Gupta, Kshitij et al. (2023). *Continual Pre-Training of Large Language Models: How to (re)warm your model?* URL: https://arxiv.org/abs/2308.04014 (cited on page 115).

Hendy, Amr et al. (2023a). *How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation*. DOI: 10.48550/ARXIV.

2302.09210. URL: https://arxiv.org/abs/2302.09210 (cited on pages 3, 19, 26, 65, 66, 69, 75).

Hendy, Amr et al. (2023b). 'How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation'. In: *arXiv preprint arXiv:2302.09210* (cited on page 9).

Jiang, Albert Q. et al. (2023). *Mistral 7B*. URL: https://arxiv.org/abs/2310.06825 (cited on page 118).

Juraska, Juraj et al. (Dec. 2023). 'MetricX-23: The Google Submission to the WMT 2023 Metrics Shared Task'. In: *Proceedings of the Eighth Conference on Machine Translation*. Ed. by Philipp Koehn et al. Singapore: Association for Computational Linguistics, pp. 756–767. DOI: 10.18653/v1/2023.wmt-1.63 (cited on pages 27, 101, 119).

Karpinska, Marzena and Mohit Iyyer (2023). *Large language models effectively leverage document-level context for literary translation, but critical errors persist* (cited on page 18).

Kocmi, Tom and Christian Federmann (Dec. 2023a). 'GEMBA-MQM: Detecting Translation Quality Error Spans with GPT-4'. In: *Proceedings of the Eighth Conference on Machine Translation*. Ed. by Philipp Koehn et al. Singapore: Association for Computational Linguistics, pp. 768–775. DOI: 10.18653/v1/2023.wmt-1.64 (cited on pages 95, 101, 131).

– (June 2023b). 'Large Language Models Are State-of-the-Art Evaluators of Translation Quality'. In: *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*. Tampere, Finland: European Association for Machine Translation, pp. 193–203 (cited on pages 27, 101, 102).

– (2023c). *Large Language Models Are State-of-the-Art Evaluators of Translation Quality*. DOI: 10.48550/ARXIV.2302.14520. URL: https://arxiv.org/abs/2302.14520 (cited on page 66).

Kocmi, Tom et al. (Dec. 2023). 'Findings of the 2023 Conference on Machine Translation (WMT23): LLMs Are Here but Not Quite There Yet'. In: *Proceedings of the Eighth Conference on Machine Translation*. Singapore: Association for Computational Linguistics (cited on page 9).

Leiter, Christoph et al. (2023). 'Towards Explainable Evaluation Metrics for Machine Translation'. In: *ArXiv* abs/2306.13041 (cited on pages 27, 95).

Liu, Wei et al. (2023). 'D-Separation for Causal Self-Explanation'. In: *Advances in Neural Information Processing Systems*. Ed. by A. Oh et al. Vol. 36. Curran Associates, Inc., pp. 43620–43633 (cited on page 92).

Lukas, Nils et al. (2023). 'Analyzing Leakage of Personally Identifiable Information in Language Models'. In: *Proceedings of the 2023 IEEE Symposium on Security and Privacy (SP)* (cited on page 5).

Manakul, Potsawee, Adian Liusie, and Mark J. F. Gales (2023). *SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models*. DOI: 10.48550/ARXIV.2303.08896. URL: https://arxiv.org/abs/2303.08896 (cited on pages 28, 69).

Moghe, Nikita et al. (July 2023). 'Extrinsic Evaluation of Machine Translation Metrics'. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 13060–13078. DOI: 10.18653/v1/2023.acl-long.730 (cited on page 131).

Mohebbi, Hosein et al. (2023). *Quantifying Context Mixing in Transformers* (cited on pages 7, 24).
Niculae, Vlad et al. (2023). *Discrete Latent Structure in Neural Networks* (cited on page 7).
OpenAI (2023). *GPT-4 Technical Report* (cited on page 101).
Peng, Keqin et al. (2023). 'Towards Making the Most of ChatGPT for Machine Translation'. In: *ResearchGate preprint*. DOI: https://doi.org/10.13140/RG.2.2.24416.97283 (cited on pages 19, 65).
Raunak, Vikas et al. (July 2023). 'Do GPTs Produce Less Literal Translations?' In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 1041–1050. DOI: 10.18653/v1/2023.acl-short.90 (cited on page 26).
Rei, Ricardo et al. (Dec. 2023a). 'Scaling up CometKiwi: Unbabel-IST 2023 Submission for the Quality Estimation Shared Task'. In: *Proceedings of the Eighth Conference on Machine Translation*. Singapore: Association for Computational Linguistics, pp. 839–846 (cited on page 96).
Rei, Ricardo et al. (Dec. 2023b). 'Scaling up CometKiwi: Unbabel-IST 2023 Submission for the Quality Estimation Shared Task'. In: *Proceedings of the Eighth Conference on Machine Translation*. Ed. by Philipp Koehn et al. Singapore: Association for Computational Linguistics, pp. 841–848. DOI: 10.18653/v1/2023.wmt-1.73 (cited on page 119).
Rei, Ricardo et al. (July 2023c). 'The Inside Story: Towards Better Understanding of Machine Translation Neural Evaluation Metrics'. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Toronto, Canada: Association for Computational Linguistics, pp. 1089–1105 (cited on pages 27, 95, 100, 107, 131).
Reinauer, Raphael et al. (2023). *Neural Machine Translation Models Can Learn to be Few-shot Learners*. URL: https://arxiv.org/abs/2309.08590 (cited on page 115).
Sai B, Ananya et al. (July 2023). 'IndicMT Eval: A Dataset to Meta-Evaluate Machine Translation Metrics for Indian Languages'. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Toronto, Canada: Association for Computational Linguistics, pp. 14210–14228 (cited on page 99).
Touvron, Hugo et al. (2023a). *Llama 2: Open Foundation and Fine-Tuned Chat Models*. URL: https://arxiv.org/abs/2307.09288 (cited on page 115).
Touvron, Hugo et al. (2023b). *LLaMA: Open and Efficient Foundation Language Models* (cited on pages 3, 24, 95).
Treviso, Marcos et al. (July 2023). 'CREST: A Joint Framework for Rationalization and Counterfactual Text Generation'. In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki. Toronto, Canada: Association for Computational Linguistics, pp. 15109–15126. DOI: 10.18653/v1/2023.acl-long.842 (cited on page 92).
Wei, Jason et al. (2023a). *Chain-of-Thought Prompting Elicits Reasoning in Large Language Models* (cited on page 126).

Wei, Xiangpeng et al. (2023b). *PolyLM: An Open Source Polyglot Large Language Model*. URL: https://arxiv.org/abs/2307.06018 (cited on page 115).

Workshop, BigScience et al. (2023). *BLOOM: A 176B-Parameter Open-Access Multilingual Language Model* (cited on page 24).

Xu, Weijia et al. (2023a). *Understanding and Detecting Hallucinations in Neural Machine Translation via Model Introspection*. DOI: 10.48550/ARXIV.2301.07779. URL: https://arxiv.org/abs/2301.07779 (cited on pages 6, 30–32, 51, 65, 67, 68).

– (2023b). *Understanding and Detecting Hallucinations in Neural Machine Translation via Model Introspection* (cited on page 25).

Xu, Wenda et al. (2023c). *INSTRUCTSCORE: Towards Explainable Text Generation Evaluation with Automatic Feedback* (cited on pages 27, 95, 101, 131).

Zhang, Shaolei et al. (2023). *BayLing: Bridging Cross-lingual Alignment and Instruction Following through Interactive Translation for Large Language Models*. URL: https://arxiv.org/abs/2306.10968 (cited on page 115).

Zhao, Wayne Xin et al. (2023). *A Survey of Large Language Models* (cited on page 24).

Zheng, Lianmin et al. (2023). *Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena*. URL: https://arxiv.org/abs/2306.05685 (cited on page 131).

Agrawal, Sweta et al. (2024a). *Can Automatic Metrics Assess High-Quality Translations?* URL: https://arxiv.org/abs/2405.18348 (cited on page 131).

Agrawal, Sweta et al. (Nov. 2024b). 'Modeling User Preferences with Automatic Metrics: Creating a High-Quality Preference Dataset for Machine Translation'. In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, pp. 14503–14519. DOI: 10.18653/v1/2024.emnlp-main.803 (cited on page 109).

Agrawal, Sweta et al. (2024c). *Modeling User Preferences with Automatic Metrics: Creating a High-Quality Preference Dataset for Machine Translation* (cited on page 124).

AI@Meta (2024). *Llama 3 Model Card*. URL: https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md (cited on page 118).

Alves, Duarte Miguel et al. (2024). *Tower: An Open Multilingual Large Language Model for Translation-Related Tasks*. URL: https://openreview.net/forum?id=EHPns3hVkj (cited on pages 115, 116).

Anthropic (May 2024). *Golden Gate Claude*. Accessed: 20424-09-30. URL: https://www.anthropic.com/news/golden-gate-claude (cited on page 133).

Anugraha, David et al. (Nov. 2024). 'MetaMetrics-MT: Tuning MetaMetrics for Machine Translation via Human Preference Calibration'. In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by Barry Haddow et al. Miami, Florida, USA: Association for Computational Linguistics, pp. 459–469. DOI: 10.18653/v1/2024.wmt-1.32 (cited on page 109).

Bavaresco, Anna et al. (2024). *LLMs instead of Human Judges? A Large Scale Empirical Study across 20 NLP Evaluation Tasks*. URL: https://arxiv.org/abs/2406.18403 (cited on page 131).

Benkirane, Kenza et al. (2024). *Machine Translation Hallucination Detection for Low and High Resource Languages using Large Language Models*. URL: https://arxiv.org/abs/2407.16470 (cited on page 131).

Briakou, Eleftheria et al. (2024). *On the Implications of Verbose LLM Outputs: A Case Study in Translation Evaluation*. URL: https://arxiv.org/abs/2410.00863 (cited on page 77).

Ding, Dujian et al. (2024). 'Hybrid LLM: Cost-Efficient and Quality-Aware Query Routing'. In: *The Twelfth International Conference on Learning Representations* (cited on page 133).

Finkelstein, Mara et al. (2024). *MBR and QE Finetuning: Training-time Distillation of the Best and Most Expensive Decoding Methods*. URL: https://arxiv.org/abs/2309.10966 (cited on page 118).

Freitag, Markus et al. (Nov. 2024). 'Are LLMs Breaking MT Metrics? Results of the WMT24 Metrics Shared Task'. In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by Barry Haddow et al. Miami, Florida, USA: Association for Computational Linguistics, pp. 47–81. DOI: 10.18653/v1/2024.wmt-1.2 (cited on page 109).

Grant, Nico (Sept. 2024). *Google Rolls Back A.I. Search Feature After Flubs and Flaws*. Accessed: 2024-09-09. URL: https://www.nytimes.com/2024/06/01/technology/google-ai-overviews-rollback.html (cited on page 5).

Han, HyoJung, Kevin Duh, and Marine Carpuat (Nov. 2024). 'SpeechQE: Estimating the Quality of Direct Speech Translation'. In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, pp. 21852–21867. DOI: 10.18653/v1/2024.emnlp-main.1218 (cited on page 124).

Himmi, Anas et al. (Nov. 2024). 'Enhanced Hallucination Detection in Neural Machine Translation through Simple Detector Aggregation'. In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, pp. 18573–18583 (cited on pages 52, 63).

Hu, Qitian Jason et al. (2024). *RouterBench: A Benchmark for Multi-LLM Routing System*. URL: https://arxiv.org/abs/2403.12031 (cited on page 133).

Huang, Chenyang et al. (Aug. 2024). 'OTTAWA: Optimal TransporT Adaptive Word Aligner for Hallucination and Omission Translation Errors Detection'. In: *Findings of the Association for Computational Linguistics: ACL 2024*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, pp. 6322–6334. DOI: 10.18653/v1/2024.findings-acl.377 (cited on page 63).

Kocmi, Tom et al. (2024a). *Error Span Annotation: A Balanced Approach for Human Evaluation of Machine Translation*. URL: https://arxiv.org/abs/2406.11580 (cited on page 117).

Kocmi, Tom et al. (Nov. 2024b). 'Findings of the 2024 Conference on Machine Translation (WMT24)'. In: *Proceedings of the Ninth Conference on Machine Translation*. Miami, Florida: Association for Computational Linguistics (cited on pages 115, 123).

Kocmi, Tom et al. (2024c). 'Navigating the Metrics Maze: Reconciling Score Magnitudes and Accuracies'. In: *arXiv prepring arXiv:2401.06760* (cited on pages 119, 131).

Kocmi, Tom et al. (2024d). *Preliminary WMT24 Ranking of General MT Systems and LLMs*. URL: https://arxiv.org/abs/2407.19884 (cited on pages 119, 122, 123).

Larionov, Daniil et al. (2024). *xCOMET-lite: Bridging the Gap Between Efficiency and Quality in Learned MT Evaluation Metrics*. URL: https://arxiv.org/abs/2406.14553 (cited on pages 96, 109).

Li, Xian et al. (2024). 'Self-Alignment with Instruction Backtranslation'. In: *The Twelfth International Conference on Learning Representations* (cited on page 9).

Lieberum, Tom et al. (2024). *Gemma Scope: Open Sparse Autoencoders Everywhere All At Once on Gemma 2*. URL: https://arxiv.org/abs/2408.05147 (cited on page 133).

Martins, Pedro Henrique et al. (2024). *EuroLLM: Multilingual Language Models for Europe*. URL: https://arxiv.org/abs/2409.16235 (cited on page 124).

Moghe, Nikita et al. (2024). *Machine Translation Meta Evaluation through Translation Accuracy Challenge Sets*. URL: https://arxiv.org/abs/2401.16313 (cited on page 131).

Ong, Isaac et al. (2024). *RouteLLM: Learning to Route LLMs with Preference Data*. URL: https://arxiv.org/abs/2406.18665 (cited on page 133).

Peters, Ben and André F. T. Martins (2024). *Did Translation Models Get More Robust Without Anyone Even Noticing?* URL: https://arxiv.org/abs/2403.03923 (cited on page 124).

Pombal, Jose, Sweta Agrawal, and André Martins (Nov. 2024). 'Improving Context Usage for Translating Bilingual Customer Support Chat with Large Language Models'. In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by Barry Haddow et al. Miami, Florida, USA: Association for Computational Linguistics, pp. 993–1003. DOI: 10.18653/v1/2024.wmt-1.100 (cited on page 124).

Ramos, Miguel Moura et al. (2024). *Fine-Grained Reward Optimization for Machine Translation using Error Severity Mappings*. URL: https://arxiv.org/abs/2411.05986 (cited on page 109).

Richburg, Aquia and Marine Carpuat (2024a). *How Multilingual Are Large Language Models Fine-Tuned for Translation?* URL: https://arxiv.org/abs/2405.20512 (cited on page 121).

– (2024b). 'How Multilingual are Large Language Models Fine-tuned for Translation?' In: *First Conference on Language Modeling* (cited on page 124).

Santos, Saul et al. (2024). *Sparse and Structured Hopfield Networks*. URL: https://arxiv.org/abs/2402.13725 (cited on page 93).

Sennrich, Rico, Jannis Vamvas, and Alireza Mohammadshahi (Mar. 2024). 'Mitigating Hallucinations and Off-target Machine Translation with Source-Contrastive and Language-Contrastive Decoding'. In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by Yvette Graham and Matthew Purver. St. Julian's, Malta: Association for Computational Linguistics, pp. 21–33 (cited on pages 52, 76).

Silva, João Daniel et al. (2024). 'Multilingual Vision-Language Pre-training for the Remote Sensing Domain'. In: *SIGSPATIAL/GIS* (cited on page 124).

Singh, Shivalika et al. (2024). *Aya Dataset: An Open-Access Collection for Multilingual Instruction Tuning*. URL: https://arxiv.org/abs/2402.06619 (cited on page 119).

Srivatsa, KV Aditya, Kaushal Kumar Maurya, and Ekaterina Kochmar (2024). *Harnessing the Power of Multiple Minds: Lessons Learned from LLM Routing*. URL: https://arxiv.org/abs/2405.00467 (cited on page 133).

Tan, Shaomu et al. (Nov. 2024). 'UvA-MT's Participation in the WMT24 General Translation Shared Task'. In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by Barry Haddow et al. Miami, Florida, USA: Association for Computational Linguistics, pp. 176–184. DOI: 10.18653/v1/2024.wmt-1.11 (cited on page 124).

Templeton, Adly et al. (2024). 'Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet'. In: *Transformer Circuits Thread* (cited on pages 133, 134).

Tomani, Christian et al. (Aug. 2024). 'Quality-Aware Translation Models: Efficient Generation and Quality Estimation in a Single Model'. In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Lun-Wei Ku, Andre Martins, and Vivek Srikumar. Bangkok, Thailand: Association for Computational Linguistics, pp. 15660–15679. DOI: 10.18653/v1/2024.acl-long.836 (cited on page 134).

Treviso, Marcos et al. (2024). *xTower: A Multilingual LLM for Explaining and Correcting Translation Errors*. URL: https://arxiv.org/abs/2406.19482 (cited on page 124).

Waldendorf, Jonas, Barry Haddow, and Alexandra Birch (Mar. 2024). 'Contrastive Decoding Reduces Hallucinations in Large Multilingual Machine Translation Models'. In: *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by Yvette Graham and Matthew Purver. St. Julian's, Malta: Association for Computational Linguistics, pp. 2526–2539 (cited on page 76).

Wang, Tianlu et al. (2024). *Self-Taught Evaluators*. URL: https://arxiv.org/abs/2408.02666 (cited on page 131).

Wu, Tianhao et al. (2024). *Meta-Rewarding Language Models: Self-Improving Alignment with LLM-as-a-Meta-Judge*. URL: https://arxiv.org/abs/2407.19594 (cited on page 131).

Xu, Haoran et al. (2024a). 'A Paradigm Shift in Machine Translation: Boosting Translation Performance of Large Language Models'. In: *The Twelfth International Conference on Learning Representations* (cited on page 115).

Xu, Xi et al. (Aug. 2024b). 'CMU's IWSLT 2024 Simultaneous Speech Translation System'. In: *Proceedings of the 21st International Conference on Spoken Language Translation (IWSLT 2024)*. Ed. by Elizabeth Salesky, Marcello Federico, and Marine Carpuat. Bangkok, Thailand (in-person and online): Association for Computational Linguistics, pp. 154–159. DOI: 10.18653/v1/2024.iwslt-1.20 (cited on page 124).

Yıldız, Çağatay et al. (2024). *Investigating Continual Pretraining in Large Language Models: Insights and Implications*. URL: https://arxiv.org/abs/2402.17400 (cited on page 115).

Zaranis, Emmanouil, Nuno M Guerreiro, and Andre Martins (Nov. 2024). 'Analyzing Context Contributions in LLM-based Machine Translation'. In: *Findings of the Association for Computational Linguis-*

tics: EMNLP 2024. Ed. by Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen. Miami, Florida, USA: Association for Computational Linguistics, pp. 14899–14924. DOI: 10.18653/v1/2024.findings-emnlp.876 (cited on page 124).

Zhu, Wenhao et al. (June 2024). 'Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis'. In: *Findings of the Association for Computational Linguistics: NAACL 2024*. Ed. by Kevin Duh, Helena Gomez, and Steven Bethard. Mexico City, Mexico: Association for Computational Linguistics, pp. 2765–2781 (cited on page 115).

Appendix

Appendix of Chapter 4 A

A.1 Computational runtime of our detectors

Our detectors do not require access to a GPU machine. All our experiments have been ran on a machine with 2 physical Intel(R) Xeon(R) Gold 6348 @ 2.60GHz CPUs (total of 112 threads). Obtaining Wass-to-Unif scores for all the 3415 translations from the Guerreiro et al. (2023a) dataset takes less than half a second, while Wass-to-Data scores are obtained in little over 4 minutes.

A.2 Tracing-back performance boosts to the construction of the reference set $\mathcal{R}_x$

In Section 4.5.1 in the main text, we showed that evaluating how distant a given translation is compared to a data-driven reference distribution–rather than to an *ad-hoc* reference distribution– led to increased performance. Therefore, we will now analyze the construction of the reference set $\mathcal{R}_x$ to obtain Wass-to-Data scores (step 2 in Figure 4.1). We conduct experiments to investigate the importance of the two main operations in this process: defining and length-filtering the distributions in $\mathcal{R}_{\text{held}}$.

- **Construction of $\mathcal{R}_{\text{held}}$.** To construct $\mathcal{R}_{\text{held}}$, we first need to obtain the source attention mass distributions for each sample in $\mathcal{D}_{\text{held}}$. If $\mathcal{D}_{\text{held}}$ is a parallel corpus, we can force-decode the reference translations to construct $\mathcal{R}_{\text{held}}$. As shown in Table A.1, this construction produces results similar to using good-quality model-generated translations. Moreover, we also evaluate the scenario where $\mathcal{R}_{\text{held}}$ is constructed with translations of any quality. Table A.1 shows that although filtering for quality improves performance, the gains are not substantial. This connects to findings by Guerreiro et al. (2023a): hallucinations exhibit different properties from other translations, including other incorrect translations. We offer further evidence that properties of hallucinations—in this case, the source attention mass distributions—are not only different to those of good-quality translations but also to most other model-generated translations.

- **Length-filtering the distributions in $\mathcal{R}_{\text{held}}$.** The results in Table A.2 show that length-filtering boosts performance significantly. This is expected: our translation-based length-filtering penalizes translations whose length is anomalous for their respective source sequences. This is particularly useful for detecting oscillatory hallucinations.

Table A.1: Ablations on Wass-to-Data by changing the construction of $\mathcal{R}_{\text{held}}$. We present the mean and standard deviation (in subscript) across five random seeds.

ABLATION	AUROC ↑	FPR@90TPR ↓
Model-Generated Translations		
Any	$83.27_{0.39}$	$50.08_{1.65}$
Quality-filtered	$\mathbf{84.20}_{0.15}$	$\mathbf{48.15}_{0.54}$
Reference Translations		
Any	$\mathbf{83.95}_{0.16}$	$50.26_{0.60}$

Table A.2: Ablations on Wass-to-Data by changing the length-filtering window to construct $\mathcal{R}_x$. We present the mean and standard deviation (in subscript) across five random seeds.

ABLATION	AUROC ↑	FPR@90TPR ↓
Random Sampling	$80.65_{0.15}$	$57.06_{2.04}$
Length Filtering	$\mathbf{84.20}_{0.15}$	$\mathbf{48.15}_{0.54}$

A.3 Ablations

We perform ablations on Wass-to-Data and Wass-Combo for all relevant hyperparameters: the length-filtering parameter δ, the maximum cardinality of $\mathcal{R}$, $|\mathcal{R}|_{\max}$, the value of k to compute the Wass-to-Data scores (step 4 in Figure 4.1), and the threshold on Wass-to-Unif scores to compute Wass-Combo scores. The results are shown in Table A.3 to Table A.6, respectively. We also report in Table A.7 the performance of Wass-to-Data with a 0/1 cost function instead of the ℓ_1 distance function.

- **ON LENGTH-FILTERING.** The results in Table A.3 show that, generally, the bigger the length window, the worse the performance. This is expected: if the test translation is very different in length to those obtained for the source sequences in $\mathcal{R}_x$, the more penalized it may be for the length mismatch instead of source attention distribution pattern anomalies.

- **ON THE CHOICE OF $|\mathcal{R}|_{\max}$.** Table A.4 shows that increasing $|\mathcal{R}|_{\max}$ leads to better performance, with reasonable gains obtained until $|\mathcal{R}|_{\max} = 2000$. While this increase in performance may be desirable, it comes at the cost of higher runtime.

- **ON THE CHOICE OF k.** The results in Table A.5 show that the higher the value of k, the worse the performance. However, we do not recommend using the minimum distance ($k = 1$) as it can be unstable.

- **ON THE CHOICE OF THRESHOLD ON WASS-TO-UNIF SCORES.** Table A.6 show that, generally, a higher threshold τ leads to a better performance of Wass-Combo. Wass-to-Unif scores are generally very high for fully detached hallucinations, a type of hallucinations that Wass-to-Data struggles more to detect. Thus, when combined in Wass-Combo, we obtain significant boosts in overall performance. However, if the threshold on Wass-to-Unif scores is set too low, Wass-to-Combo will correspond to Wass-to-Unif more frequently which may not be desirable as Wass-to-Data outperforms it for all other types of hallucinations. If set too high, fewer fully detached hallucinations may pass that threshold and may then be misidentified with Wass-to-Data scores.

- **ON THE CHOICE OF WASS-TO-DATA COST FUNCTION.** Table A.7 shows that using the ℓ_1 cost function instead of using the 0/1 cost function to compute Wass-to-Data scores leads to significant improvements. This suggests that when comparing the source mass attention distribution

Ablation	AUROC ↑	FPR@90TPR ↓
Random Sampling	80.65 0.15	57.06 2.04
Length Filtering ($\delta > 0$)		
$\delta = 0.1$	84.20 0.15	48.15 0.54
$\delta = 0.2$	84.37 0.17	47.12 1.04
$\delta = 0.3$	83.93 0.18	48.45 2.32
$\delta = 0.4$	83.06 0.16	50.12 1.29
$\delta = 0.5$	82.78 0.34	50.89 0.71

Table A.3: Ablation on Wass-to-Data by changing the length-filtering window to construct $\mathscr{R}$. We present the mean and standard deviation (in subscript) across five random seeds.

Ablation	AUROC ↑	FPR@90TPR ↓
$\lvert\mathscr{R}\rvert_{\max} = 100$	82.99 0.19	50.86 0.95
$\lvert\mathscr{R}\rvert_{\max} = 500$	83.93 0.08	48.07 1.37
$\lvert\mathscr{R}\rvert_{\max} = 1000$	84.20 0.15	48.15 0.54
$\lvert\mathscr{R}\rvert_{\max} = 2000$	84.40 0.14	49.23 1.08
$\lvert\mathscr{R}\rvert_{\max} = 5000$	84.43 0.13	48.05 0.59

Table A.4: Ablation on Wass-to-Data by changing the maximum cardinality of $\mathscr{R}$, $\lvert\mathscr{R}\rvert_{\max}$. We present the mean and standard deviation (in subscript) across five random seeds.

of a test translation to other such distributions obtained for other translations (instead of the ad-hoc uniform distribution used for Wass-to-Unif scores), the information from the location of the source attention mass is helpful to obtain better scores.

▸ **On the formulation of Wass-Combo.** To combine the information from Wass-to-Unif and Wass-to-Data, we could also perform a convex combination of the two scores:

$$s_{\mathsf{wc}}(\boldsymbol{x}) = \lambda s_{\mathsf{wtd}}(\boldsymbol{x}) + (1-\lambda)\tilde{s}_{\mathsf{wtu}}(\boldsymbol{x}) \tag{A.1}$$

for a predefined scalar parameter λ. In Table A.8, we show that this method is consistently subpar to our two-pass approach. In fact, this linear interpolation is not able to bring additional gains in performance for any of the tested parameters λ when compared to Wass-to-Data.

Table A.5: Ablation on Wass-to-Data by obtaining the score s_{wtd} by averaging the bottom-k distances in $\mathcal{R}$ for different values of k. We present the mean and standard deviation (in subscript) across five random seeds.

Ablation	AUROC ↑	FPR@90TPR ↓
Minimum	84.00 $_{0.33}$	52.03 $_{1.28}$
Bottom-k ($k > 1$)		
$k = 2$	84.25 $_{0.23}$	50.07 $_{0.70}$
$k = 4$	84.20 $_{0.15}$	48.15 $_{0.54}$
$k = 8$	83.99 $_{0.08}$	48.38 $_{1.10}$
$k = 16$	83.64 $_{0.04}$	48.05 $_{1.10}$
$k = 32$	83.23 $_{0.07}$	47.34 $_{0.94}$

Table A.6: Ablation on Wass-Combo by obtaining the score s_{wc} for different scalar thresholds $\tau = P_K$ (K-th percentile of $\mathbb{W}_{\text{wtu}}$). We present the mean and standard deviation (in subscript) across five random seeds.

Ablation	AUROC ↑	FPR@90TPR ↓
$\tau = P_{99}$	85.79 $_{0.08}$	51.09 $_{0.97}$
$\tau = P_{99.5}$	86.34 $_{0.07}$	49.64 $_{1.71}$
$\tau = P_{99.9}$	87.17 $_{0.07}$	47.56 $_{1.30}$
$\tau = P_{99.99}$	84.69 $_{0.15}$	48.15 $_{0.54}$

Table A.7: Ablation on Wass-to-Data by changing the cost function in the computation of the Wasserstein Distances in Equation 4.5.

Cost Function	AUROC ↑	FPR@90TPR ↓
ℓ_1 (Wasserstein-1)	84.20 $_{0.15}$	48.15 $_{0.54}$
0/1 cost	81.78 $_{0.20}$	51.72 $_{1.17}$

Table A.8: Convex combination of Wass-to-Unif and Wass-to-Data scores. We present the mean and standard deviation (in subscript) across five random seeds.

Ablation	AUROC ↑	FPR@90TPR ↓
Our Wass-Combo	87.17 $_{0.07}$	47.56 $_{1.30}$
Wass-to-Unif ($\lambda = 0$)	80.37	72.22
$\lambda = 0.2$	81.57 $_{0.00}$	69.02 $_{0.19}$
$\lambda = 0.4$	82.28 $_{0.01}$	68.69 $_{0.13}$
$\lambda = 0.6$	82.77 $_{0.02}$	66.15 $_{1.09}$
$\lambda = 0.8$	83.48 $_{0.05}$	63.01 $_{0.44}$
Wass-to-Data ($\lambda = 1$)	84.20 $_{0.15}$	48.15 $_{0.54}$

Table A.9: Comparison between ALTI+ and Wass-Combo detection methods. We present the mean and standard deviation (in subscript) across five random seeds.

Method	AUROC ↑	FPR@90TPR ↓
All		
ALTI+	84.27	66.30
Wass-Combo	**87.17** $_{0.07}$	**47.56** $_{1.30}$
Fully detached		
ALTI+	**98.21**	**2.15**
Wass-Combo	96.57 $_{0.10}$	3.56 $_{0.00}$
Oscillatory		
ALTI+	71.39	76.72
Wass-Combo	**85.74** $_{0.10}$	**41.38** $_{1.59}$
Strongly Detached		
ALTI+	73.77	89.41
Wass-Combo	**78.89** $_{0.15}$	**64.55** $_{1.93}$

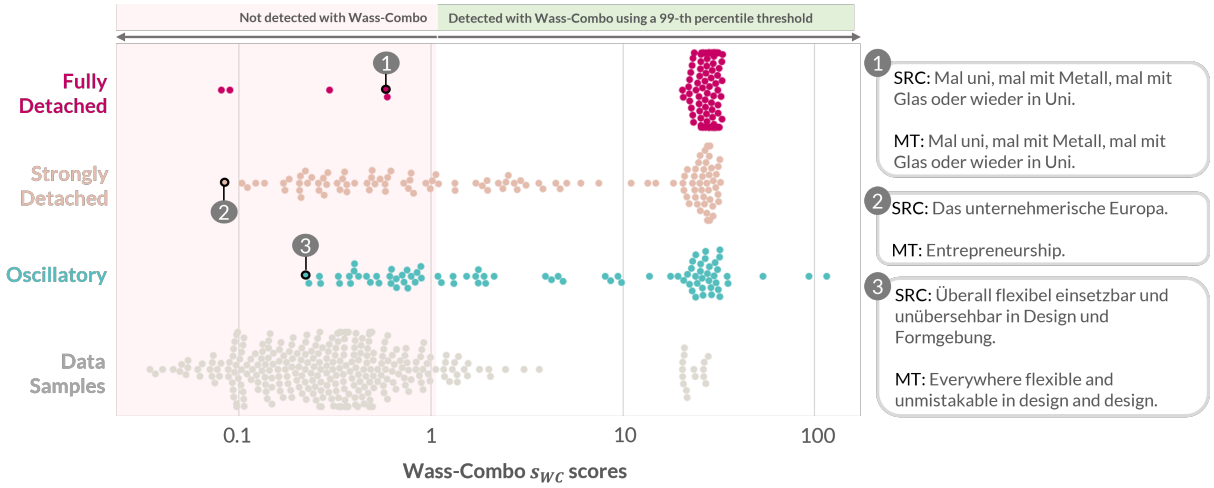


Figure A.1: Distribution of Wass-Combo scores (on log-scale) for each type of hallucination and performance on a fixed-threshold scenario. We highlight three hallucinations that are not detected by Wass-Combo. These represent hallucinations in the dataset that it struggles to identify: (1) exact copies of the source sequence, (2) small level of detachment in strongly detached hallucinations, and (3) mild repetitions of 1-grams ("design").

Table A.10: Examples of oscillatory hallucinations randomly sampled from the set of oscillatory hallucinations not detected with Wass-Combo. Most hallucinations come in the form of mild repetitions of 1-grams or 2-grams.

Oscillatory Hallucinations not detected with Wass-Combo	
Source	Überall flexibel einsetzbar und unübersehbar in Design und Formgebung.
Translation	Everywhere flexible and unmistakable in design and design.
Source	Um kahlen Stellen, wenn sie ohne Rüstung pg.
Translation	To dig dig digits if they have no armor pg.
Source	Damit wird, wie die Wirtschaftswissenschaftler sagen, der Nennwert vorgezogen.
Translation	This, as economists say, puts the par value before the par value.
Source	Besonders beim Reinigen des Verflüssigers kommt Ihnen dies zugute.
Translation	Especially when cleaning the liquefied liquefied liquefied.
Source	Müssen die Verkehrsmittel aus- oder abgewählt werden ?
Translation	Do you need to opt-out or opt-out of transport?
Source	Schnell drüberlesen - "Ja" auswählen und weiter gehts.
Translation	Simply press the "Yes" button and press the "Yes."
Source	Auf den jeweiligen Dorfplätzen finden sich Alt und Jung zum Schwätzchen und zum Feiern zusammen.
Translation	Old and young people will find themselves together in the village's respective squares for fun and fun.
Source	Zur Absicherung der E-Mail-Kommunikation auf Basis von PGP- als auch X.509-Schlüsseln hat die Schaeffler Gruppe eine Zertifizierungsinfrastruktur (Public Key Infrastructure PKI) aufgebaut.
Translation	The Schaeffler Group has set up a Public Key Infrastructure PKI (Public Key Infrastructure PKI) to secure e-mail communication based on PGP and X.509 keys.

A.4 Analysis against ALTI+

Concurrently to our work, Dale et al. (2022) leveraged ALTI+ (Ferrando et al., 2022b), a method that evaluates the global relative contributions of both source and target prefixes to model predictions, for detection of hallucinations. As hallucinations are translations detached from the source sequence, ALTI+ is able to detect them by identifying sentences with minimal source contribution. In Table A.9, we show that ALTI+ slightly outperforms Wass-Combo for fully detached hallucinations, but lags considerably behind on what comes to detecting strongly detached and oscillatory hallucinations.

A.5 Error Analysis of **Wass-Combo**

We show a qualitative analysis on the same fixed-threshold scenario described in Section 4.5.2 in Figure A.1. Differently to Figure 4.3, we provide examples of translations that have not been detected by Wass-Combo for the chosen threshold.

Our detector is not able to detect fully detached hallucinations that come in the form of exact copies of the source sentence. For these pathological translations, the attention map is mostly diagonal and is thus not anomalous. Although these are severe errors, we argue that, in a real-world application, such translations can be easily detected with string matching heuristics.

We also find that our detector Wass-Combo struggles with oscillatory hallucinations that come in the form of mild repetitions of 1-grams or 2-grams (see example in Figure A.1). To test this hypothesis, we implemented the binary heuristic top n-gram count (Raunak et al., 2021; Guerreiro et al., 2023a) to verify whether a translation is a severe oscillation: given the entire $\mathcal{D}_{\text{held}}$, a translation is flagged as an oscillatory hallucination if (i) it is in the set of 1% lowest-quality translations according to CometKiwi and (ii) the count of the top repeated 4-gram in the translation is greater than the count of the top repeated source 4-gram by at least 2. Indeed, more than 90% of the oscillatory hallucinations not detected by Wass-Combo in Figure A.1 were not flagged by this heuristic. We provide 8 examples randomly sampled from the set of oscillatory hallucinations not detected with Wass-Combo in Table A.10. Close manual evaluation of these hallucinations further backs the hypothesis above.

Table A.11: Performance of all hallucination detectors. For Wass-to-Data and Wass-Combo, we present the mean and standard deviation (in subscript) scores across five random seeds. †CometKiwi has been trained on these MLQE-PE data samples.

Detector	RO-EN		NE-EN	
	AUROC ↑	FPR@90TPR ↓	AUROC ↑	FPR@90TPR ↓
External Detectors				
CometKiwi †	99.62	**0.49**	**97.64**	**4.03**
LaBSE	**99.72**	**0.49**	92.34	20.03
Model-based Detectors				
Attn-ign-SRC	99.16	0.93	28.66	100.0
Seq-Logprob	91.97	16.42	26.38	99.94
Ours				
Wass-to-Unif	**99.30**	**0.46**	81.49	64.23
Wass-to-Data	$96.54_{0.07}$	$10.36_{0.30}$	$\mathbf{90.18}_{0.13}$	$\mathbf{48.52}_{2.64}$
Wass-Combo	$98.75_{0.06}$	$\mathbf{0.46}_{0.00}$	$\mathbf{90.16}_{0.13}$	$\mathbf{48.52}_{2.64}$

A.6 Experiments on the MLQE-PE dataset

In order to establish the broader validity of our model-based detectors, we present an analysis on their performance for other NMT models and on mid and low-resource language pairs. Overall, the detectors exhibit similar trends to those discussed in the main text (Section 8.5).

A.6.1 Model and data

The dataset from Guerreiro et al., 2023a analysed in the main text is the only available dataset that contains human annotations of hallucinated translations. Thus, in this analysis we will have to make use of other human annotations to infer annotations for hallucinations. For that end, we follow a similar setup to that of Tang et al., 2022 and use the MLQE-PE dataset (Fomicheva et al., 2022)— that has been reported to contain low-quality translations and hallucinations for NE-EN and RO-EN (Specia et al., 2021)— to test the performance of our detectors on these language pairs.

The NE-EN and RO-EN MLQE-PE datasets contain 7000 translations and their respective human quality assessments (from 1 to 100). Each translation is scored by three different annotators. As hallucinations lie at the extreme end of NMT pathologies (Raunak et al., 2021), we consider a translation to be a hallucination if at least two annotators (majority) gave it a quality score of 1.[1] This process leads to 30 hallucinations for NE-EN and 237 hallucinations for RO-EN. Although the number of hallucinations for NE-EN is relatively small, we decide to also report experiments on this language pair because the type of hallucinations found for NE-EN is very different to those found for RO-EN: almost all NE-EN hallucinations are oscillatory, whereas almost all RO-EN are fully detached.

To obtain all model-based information required to build the detectors, we use the same Transformer models that generated the translations in the datasets in consideration. All details can be found in Fomicheva et al. (2022) and the official project repository[2]. Moreover, to build our held-out databases of source mass distributions, we used readily available Europarl data (Koehn, 2005) for RO-EN (~100k samples), and filtered Nepali Wikipedia monolingual data[3] used in (Koehn et al.,

1: We tried other methods to infer hallucinations from the annotations (e.g. average quality score below 5, at least one quality score of 1). The trends on performance were similar to those reported in this section.

2: https://github.com/facebookresearch/mlqe

3: Note that creating the datastore only requires access to monolingual data. If quality filtering is needed and references are not available, we suggest using quality estimation or cross-lingual embedding similarity models to filter low-quality translations.

2019) for NE-EN (~80k samples).

A.6.2 Results

- **THE TRENDS IN SECTION 4.5.1 HOLD FOR OTHER LANGUAGE PAIRS.** The results in Table A.11 establish the broader validity of our detectors for other NMT models and, importantly, for mid and low-resource language pairs. Similarly to the analysis in 4.5.1, we find that our detectors (i) exhibit better performance than other model-based detectors with significant gains on the low-resource NE-EN language pair; and (ii) can be competitive with external detectors that leverage large models.

- **THE TRENDS IN SECTION 4.5.2 HOLD FOR OTHER LANGUAGE PAIRS.** In Section A.6.1, we remark that almost all NE-EN hallucinations are oscillatory, whereas almost all RO-EN hallucinations are fully detached. With that in mind, the results in Table A.11 establish the validity of the claims in the main-text (Section 4.5.2) on these language pairs: (i) detecting fully detached hallucinations is remarkably easy for most detectors, and Wass-to-Unif outperforms Wass-to-Data on this type of hallucinations (see results for RO-EN); and (ii) our detectors significantly outperform all previous model-based detectors on detecting oscillatory hallucinations (see results for NE-EN), which further confirms the notion that some detectors specialize on different types of hallucinations (e.g Attn-ign-SRC is particularly fit for detecting fully detached hallucinations, but it does not work for oscillatory hallucinations).

Appendix of Chapter 5 B

B.1 List of Languages

Table B.1: Language information including ISO codes, names, groups, scripts, and resource levels from Fan et al. (2020). Bridge languages shown in bold. Resource levels based on Goyal et al. (2022).

ISO	Lang	Group	Script	Res
af	Afrikaans	Germanic	Latin	Low
ar	**Arabic**	Arabic	Arabic	Med
ast	Asturian	Romance	Latin	Low
az	Azerbaijani	Turkic	Latin	Low
be	Belarusian	Slavic	Cyrillic	Low
bn	**Bengali**	Indo-Aryan	Bengali	Med
bs	Bosnian	Slavic	Latin	Low
ca	Catalan	Romance	Latin	Med
cs	**Czech**	Slavic	Latin	Med
cy	Welsh	Celtic	Latin	Low
de	**German**	Germanic	Latin	High
el	**Greek**	Hellenic	Greek	Med
en	**English**	Germanic	Latin	High
es	**Spanish**	Romance	Latin	High
et	Estonian	Uralic	Latin	Med
fa	**Persian**	Iranian	Arabic	Med
fi	**Finnish**	Uralic	Latin	Med
fr	**French**	Romance	Latin	High
gu	Gujarati	Indo-Aryan	Gujarati	Low
ha	Hausa	Afro-Asiatic	Latin	Low
he	**Hebrew**	Semitic	Hebrew	Med
hi	**Hindi**	Indo-Aryan	Devanagari	Med
hr	Croatian	Slavic	Latin	Low
hu	**Hungarian**	Uralic	Latin	Med
hy	Armenian	Armenian	Armenian	Low
id	**Indonesian**	Malayo-Poly	Latin	Med
it	Italian	Romance	Latin	High

ISO	Lang	Group	Script	Res
is	Icelandic	Germanic	Latin	Med
ja	**Japanese**	Japonic	Kanji	Med
kk	Kazakh	Turkic	Cyrillic	Low
km	Khmer	Khmer	Khmer	Low
ko	**Korean**	Koreanic	Hangul	Med
lt	**Lithuanian**	Baltic	Latin	Med
lv	Latvian	Baltic	Latin	Med
mk	Macedonian	Slavic	Cyrillic	Med
mr	Marathi	Indo-Aryan	Devanagari	Low
nl	**Dutch**	Germanic	Latin	Med
oc	Occitan	Romance	Latin	Low
pl	**Polish**	Slavic	Latin	Med
ps	Pashto	Iranian	Arabic	Low
pt	**Portuguese**	Romance	Latin	High
ro	Romanian	Romance	Latin	Med
ru	**Russian**	Slavic	Cyrillic	High
sk	Slovak	Slavic	Latin	Med
sr	Serbian	Slavic	Cyrl;Latn	Med
sv	**Swedish**	Germanic	Latin	Med
sw	**Swahili**	Niger-Congo	Latin	Low
ta	**Tamil**	Dravidian	Tamil	Low
tl	Tagalog	Malayo-Poly	Latin	Low
tr	**Turkish**	Turkic	Latin	Med
uk	Ukrainian	Slavic	Cyrillic	Med
vi	**Vietnamese**	Vietic	Latin	Med
zh	**Chinese**	Chinese	Chinese	Med
zu	Zulu	Niger-Congo	Latin	Low

B.2 Dataset Statistics

- **FLORES.** The FLORES benchmark (Goyal et al., 2022) consists of a multi-parallel dataset in 101 languages with sentences extracted from Wikipedia. The sentences in the dataset are translated from original English sentences. We join the `dev` and `devtest` splits for a total of 2009 parallel sentences for each translation direction.

- **TICO.** The TICO benchmark (Anastasopoulos et al., 2020) is a multilingual dataset specialized in the medical domain obtained by combining English open-source data from various sources, such as scientific articles, government health announcements, and others. We join the `dev` and `test` set for a total of 2100 parallel sentences for each translation direction.

- **WMT BENCHMARKS.** We selected the same English-centric WMT benchmarks that were used as evaluation sets in the original M2M paper (Fan et al., 2020). Additionally, we selected recent WMT test sets from the WMT21 and WMT22 campaigns as they were released after the models were trained. A summary of the datasets used for each translation direction can be found in Table B.2.

Table B.2: Specification of the WMT benchmarks used in this work.

cs	de	et	fi	fr	ha	is	ja	lt	lv	ru	tr	uk	zh
2018	2019	2018	2017	2014	2021	2021	2022	2019	2017	2019	2017	2022	2019

B.3 Hallucinations under Perturbation

▸ **Examples of hallucinations under perturbation generated by GPT models.** We provide examples of hallucinations under perturbation in Figure B.1.

B.4 ALTI+ scores distribution

We present the distribution of ALTI+ scores for all translation directions in Figure B.2. For the M2M models, we find the same trends presented in Ferrando et al. (2022b): smaller source contributions when translating out of English and for low-resource language pairs. Remarkably, it is also clear from the distributions that ALTI+ scores for SMaLL100 are not only significantly higher than those of the M2M models, but they are also more consistent across different resource levels.

B.5 Examples of Natural Hallucinations

We provide several examples of English-centric natural hallucinations generated by the M2M models in Figure B.3, which spans 2 pages.

1	**Source**	*or* Italian is also the everyday language used by most of those who work in the state while Latin is often used in religious ceremonies.
	Translation (hr)	Tajanstvene su ličnosti koje se skrivaju iza Spartacističkog saveza.
	Oracle to en	They are mysterious figures who hide behind the Spartacist Alliance.
2	**Source**	Sharks do exist, however they rarely attack *humams*. *Mosb* sharks are scared of humans and would swim away.
	Translation (lt)	Reikia išversti šią frazę iš anglų kalbos į lietuvių kalbą, šaltinis: "Sharks do exist, however they rarely attack humans. Most sharks are scared of humans and would swim away." Tikslas:
	Oracle to en	It is necessary to translate this phrase from English into English into English, the source: "Sharks do exist, however they rarely attack humans. Most sharks are scared of humans and would swim away." Aim:
3	**Source**	*it* Luxembourg has a long history but its independence dates from 1839.
	Translation (vi)	Đây là câu tiếng Anh, nên không có cách nào dịch qua tiếng Việt được.
	Oracle to en	This is an English sentence, so there is no way to translate it into Vietnamese.
4	**Source**	*be* Fish often die because of the high concentrations of the toxin in the waters.
	Translation (bn)	উচ্চ মাত্রায় নোক্‌খায় কারণে মাছ সমস্যার ঝুঁকিতে আপনা ঘটে। (Ucch matraya nakkhaya karane macha samasyar jhunkite apana ghate.)
	Oracle to en	Fish are at risk of problems due to high levels of salt. (Ucch matraya nakkhaya karane macha samasyar jhunkite apana ghate.)
5	**Source**	The *eatly* reports say the plane was diverted back to Afghanistan after *beind* denied an emergency *landnng* in Ürümqi.
	Translation (bn)	Fráze "The early reports say the plane was diverted back to Afghanistan after being denied an emergency landing in Ürümqi." přeložte do češtiny. Cíl:
	Oracle to en	Translate the phrase "The early reports say the plane was diverted back to Afghanistan after being denied an emergency landing in Ürümqi." into Czech. Target:
6	**Source**	*was* Virgin have only purchased the 'good bank' of Northern Rock, not the asset management company.
	Translation (el)	Προέρχεται από την αμερικανική νομοθεσία Section 230 του Communications Decency Act.
	Oracle to en	It is derived from the U.S. Section 230 legislation of the Communications Decency Act.

Figure B.1: Examples of hallucinations under perturbation generated by GPT models. We present an oracle English translation of the generated hallucination with Bing Microsoft Translator (ORACLE TO EN). The source perturbations are shown in *italic* and coloured differently to the rest of the text.

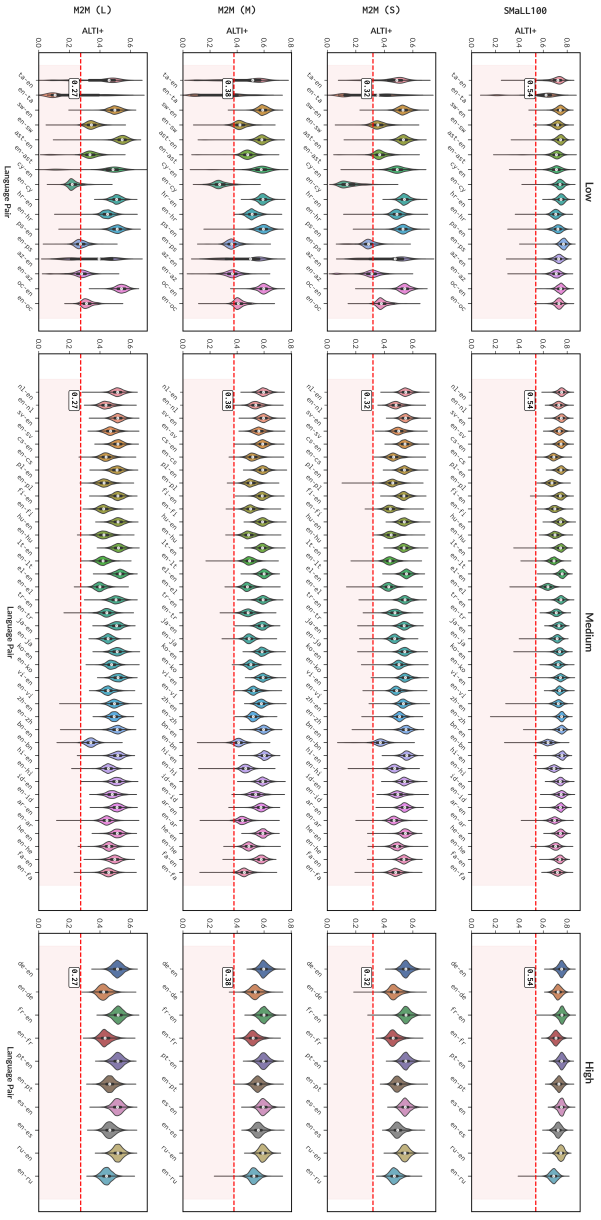


Figure B.2: ALTI+ scores for all translation directions for each of the models considered in the English-centric FLORES setup. We represent with horizontal red lines the model-based specific thresholds tuned on the validation data.

1

Source (cy) Mae twristiaid yn gallu ymweld â thirnodau gwahanol gwlad benodol neu gallent ddewis i ganolbwyntio ar un ardal yn unig.

Translation (en) There are several other sets of prime numbers that could also be rearranged to give this input, for example {5, 31, 19} and {3, 11, 59}.

Reference Tourists may visit different landmarks of a particular country or they may simply opt to focus on just one area.

2

Source (en) Natural factors can intersect to produce ideal conditions, allowing this algae to increase in number dramatically.

Translation (de) Natürliche Faktoren können miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander miteinander.

3

Source (fr) Les larges boulevards, les bâtiments vitrés et les centres commerciaux modernes sont parsemés de toits traditionnels en tuiles rouges, du marché du XVIIIe siècle, et de vieilles mosquées et églises, bien que la ville ait une atmosphère qui rappelle plus l'Europe méditerranéenne que la Turquie traditionnelle.

Translation (en) The wide boulevards, glass buildings and modern shopping centers are filled with traditional roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof roof.

4

Source (en) Elements and compounds can move from one state to another and not change.

Translation (sw) Molekuli za dutu tofauti ni ionized, kutengwa na kila mmoja katika malipo yao chanya au hasi.

Oracle to en Molecules of different substances are ionized, separated from each other in their positive or negative charge.

5

Source (ast) La proposición orixinal fízola la ex alcaldesa de Sao Paulo, Marta Suplicy. La llexislación propuesta, dempués de les enmiendes, ta agora nes manes de Roberto Jefferson.

Translation (en) This is the first time that the world's most famous and most famous cities have been built in the Middle East.

Reference The original bill was drafted by former mayor of São Paulo, Marta Suplicy. The proposed legislation, after being amended, is now in the hands of Roberto Jefferson.

6

Source (en) Fellow wrestlers also paid tribute to Luna.

Translation (hr) Bratelji i Siječevac spadaju pod MO Budinjak i gkt .

Oracle to en Brothers and Siječ evac fall under MO Budinjak and gkt .

7

Source (en) Between 10:00-11:00 pm MDT, a fire was started by the inmates in the yard.

Translation (lt) Nuo 10 val. iki 11 val. Vidurio

8

Source (en) The blade on a modern ice skate has a double edge with a concave hollow between them. The two edges allow for a better grasp of the ice, even when tilted.

Translation (tr) Çift taraflı bir çerçeveye sahip olan bu çerçeve, iki taraflı bir çerçeveye sahiptir; bu çerçeve, iki taraflı bir çerçeveye sahiptir; iki taraflı bir çerçeveye sahiptir.

Oracle to en This frame, which has a double-sided frame, has a two-sided frame; this framework has a two-sided framework; has a two-sided frame.

9

Source (en) Buhari Ba Zai Karbi Masu Gaisuwar Sallah Ba.

Translation (ha) Bhagavad-gita is one of the greatest religions in the world.

Reference Buhari will not welcome those coming to greet him on the occasion of Sallah.

10

Source (zh) 伊氏西爪龙属于驰龙科，是迅猛龙的表亲。

Translation (en) Isaiah also revealed what would become of those idols.

Reference Hesperonychus elizabethae is a species of the family Dromaeosauridae and is a cousin of Velociraptor .

11	**Source** (en)	Do not deface the site by marking or scratching graffiti into structures.
	Translation (ja)	グラフィティを構造にマークしたり、グラフィティをグラフィティをグラフィティにすることにより、サイトを壊さないでください。
	Oracle to en	Do not break the site by marking graffiti into structures or making graffiti graffiti graffiti.
12	**Source** (en)	We make our houses from plants and make clothes from plants. Most foods that we eat are plants. Without plants, animals could not survive.
	Translation (ar)	نحن نخلق من النباتات، ونخلق من النباتات، ونخلق من النباتات، ونخلق من النباتات، ونخلق من النباتات، ونخلق من النباتات، ونخلق من النباتات.
	Oracle to en	We create from plants, we create from plants, we create from plants, we create from plants, we create from plants, we create from plants, we create from plants.
13	**Source** (uk)	Точно не скажу,бо не дуже ще орієнтуюся в місті..біля половини 12
	Translation (en)	I don't know, I don't know, I don't know, I don't know, I don't know, I don't know, I don't know, I don't know, I don't know, I don't know, I don't know, I don't know.
14	**Source** (en)	For over a thousand years the Christian religion had bound European states together despite differences in language and customs.
	Translation (az)	Həzrət Peyğəmbər (s.a.s) bir min ildir ki, Həzrət Peyğəmbərin (s.a.s) bir-birinə bir-birinə aiddir.
	Oracle to en	And the Prophet (Mohammed peace and – blessings be upon him) has been intertwined with one another for a thousand years.
15	**Source** (de)	Scharf bewacht - der Parteitag der Tories
	Translation (en)	Toughened Laminated Safety Glass - manufacturer, factory
	Reference	Under tight security - the Tory party conference
16	**Source** (et)	Tallinnas Meriväljal oli 6.augusti õhtul üks pisike plikatirts hüüdnud: "Ema, ema!
	Translation (en)	On the night of the 6th of August, a little bit of a little bit.
	Reference	In the Meriválja part of Tallinn, on 6 August, one tiny slip of a girl was said to have shouted: “Mother, mother!”

Figure B.3: Examples of hallucinations generated by the M2M models in English-centric directions. We present an oracle English translation of the generated hallucination with Bing Microsoft Translator (ORACLE TO EN) when necessary.

Appendix of Chapter 6 C

C.1 Datasets for Highlights Extraction

We used five datasets for sentiment analysis: four for text classification (SST, AgNews, IMDB, Hotels) (Del Corso et al., 2005; Wang et al., 2010; Maas et al., 2011; Socher et al., 2013) and one for regression (BeerAdvocate) McAuley et al., 2012. The Hotels and BeerAdvocate datasets contain data instances for multiple aspects. In this work, we use the Hotels' location aspect and the BeerAdvocate's appearance aspect. These two datasets contain *sentence-level* rationale annotations for their test sets. For these datasets, we use the splits used in Bao et al. (2018). For all other datasets, we use the splits in Wolf et al. (2020). For IMDB and AgNews we randomly selected 10%, 15% of examples from the training set to be used as validation data, respectively. Table C.1 shows statistics for each dataset and rationale length as a percentage of each document.

Dataset	# Train	# Test	# Classes	Rationale Length (%)
SST	6920	1821	2	20
AgNews	120K	7600	4	20
IMDB	25K	25K	2	20
Hotels	12K	200	2	15
Beer	80K	997	–	10

Table C.1: Dataset statistics and rationale length as a percentage of each document.

For the datasets without human annotations, we used the same sparsity level (20%) – Jain et al. (2020) uses this value for AgNews and SST; for BeerAdvocate, we used the sparsity levels used in Lei et al. (2016) and Yu et al. (2019); and, for Hotels we opted to select a sparsity level of 15% (human annotations average around 10% sparsity level).

C.2 Datasets for Matchings Extraction

For natural language inference (NLI), we used SNLI and MNLI (Bowman et al., 2015; Chen et al., 2017). For MNLI, we split the MNLI matched validation set into equal validation and test sets. Table C.2 shows statistics for each dataset and the alignment budget used for the M:Budget factor.

Dataset	# Train	# Test	# Classes	Alignment Budget
SNLI	550K	10K	3	4
MNLI	392K	10K	3	6
HANS	30K	30K	2	–

Table C.2: Dataset statistics and alignment budget for each dataset.

For SNLI, we set the Budget B to 4 to compare with the OT approach (OT exact $k = 4$) of Swanson et al. (2020). For MNLI, we set B to 6, since the average premise length in MNLI is around 50% bigger than that of SNLI.

We also conduct experiments with the HANS (McCoy et al., 2019) dataset. This dataset consists of a controlled evaluation set to detect

whether NLI systems are exploring linguistic heuristics such as lexical overlap, subsequence and constituent heuristics. A detailed description of each of these heuristics can be found in the original paper. The dataset is also constituted by 30,000 HANS-like examples that can be used to augment existing NLI training sets such as SNLI or MNLI.

C.3 Implementation Details

C.3.1 Rationalizers experimental setup

For all rationalizers, we map each input word to 300D-pretrained GloVe embeddings from 840B release (Pennington et al., 2014) that are kept frozen. We instantiate all encoder networks as bidirectional LSTM (**LSTM**) layers (BiLSTM) (w/ hidden size 200) similarly to Lei et al. (2016), Bastings et al. (2019), and Treviso and Martins (2020). Although other works (Jain et al., 2020; Paranjape et al., 2020) use more powerful BERT-based representations, we firstly experimented with BiLSTM layers and noticed our results were competitive with those reported in Jain et al. (2020). We used the Adam optimizer (Kingma and Ba, 2017) for all experiments with learning rate within $\{1 \times 10^{-3}, 5 \times 10^{-4}, 1 \times 10^{-4}, 5 \times 10^{-5}\}$ and ℓ_2 regularization within $\{10^{-4}, 10^{-5}\}$. We also enforce a grad norm of 5.0. We train all models for highlights extraction for a minimum of 5 epochs and maximum of 25 epochs. For matchings extraction, we set the number of minimum epochs and maximum epochs to 3 and 10, respectively.

Training for all methods for highlights extraction but HardKuma is stopped if Macro F_1 (for classification) or MSE (for regression) is not improved for 5 epochs. For matchings extraction, training is stopped if Macro F_1 does not improve for 3 epochs. For HardKuma, we train until the maximum number of epochs. This is because the rationale length might vary considerably during training due to the Lagrangian relaxation algorithm that is employed at training time. We found that using early stopping would often favour models that selected almost all of the input text. Unlike Jain et al. (2020), we decided to carefully tune both model and the Lagrangian relaxation algorithm hyperparameters for this rationalizer. This had a big impact on the performance, as HardKuma performed poorly with the top-k and contiguous strategies at inference time. Even though some careful tuning is required and degeneration might occur for some random seeds, it is still much less cumbersome than tuning the variants of the rationalizer of Lei et al. (2016). We hypothesize that this is mostly due to two factors: the control on the rationale average size that the Lagrangian relaxation algorithm aims to impose; and the gradient estimates with reparameterized gradients exhibit less variance than those with the score function estimator.

All models for highlights extraction have 1.8M trainable parameters. Models for faithful and non-faithful selective rationalization of text matchings have 1.7M and 1.8M trainable parameters, respectively.

C.3.2 SPECTRA sparsity regularization

During training, we apply a temperature term T in the sparsemax and fusedmax operators. This parameter is set within $\{0.05, 0.1, 0.2\}$. The total variation regularization for fusedmax is set to 0.7.

For the models that use the LP-SparseMAP extraction layer, we use a temperature term T set within $\{0.05, 0.1, 0.2\}$ during training. Moreover, for the H:SeqBudget, we set the transition scores within $\{0.001, 0.005\}$ for all datasets. All hyperparamter searches were conducted manually.

The LP-SparseMAP problem can be interpreted as the ℓ_2-regularized LP-MAP. Its output corresponds to a probability distribution over a sparse set of structures. Therefore LP-MAP can be seen as LP-SparseMAP with the scores divided by a zero-limit temperature parameter. This procedure at test time would lead to the LP-MAP solution, which is generally an outer relaxation of MAP (Martins et al., 2015). When inference in the factor graph is exact, the solutions of the LP-MAP are integer (i.e., LP-MAP yields the true MAP). But that is not the case for when inference in the factor graph is not exact. Thus, LP-SparseMAP solutions for this test time setting might be a soft or discrete selection of parts of the input. We used a temperature parameter of 10^{-3} at validation and testing time.

C.4 Computational Cost of SPECTRA

Tables C.3 and C.4 show the average training and validation time per epoch for each dataset with the SPECTRA strategies for highlights and matchings extraction, respectively. We also present, for comparison, the average training and validation time per epoch for some of the other methods we used in the paper. The batch size for the models for highlights extraction is 32. For matchings extraction, the batch size for all models but M:Budget is 16. For M:Budget, the batch size is 8.

The computational time of SPECTRA depends on several factors inherited from the use of LP-SparseMAP as the extractive method. Generally, the bigger the number of local factors $f \in \mathcal{F}$, the more costly it is to compute a solution. Thus, it might be necessary to increase the number of iterations for the LP-SparseMAP to converge to a solution for which all factors agree. We set this number to 10 in training time following Niculae and Martins (2020). During inference, we set a maximum number of iterations of 1000. For highlights extraction, the H:SeqBudget consists of a single factor, thus the solution is found within a single iteration. For matchings extraction, our factors consist of multiple local factors that impose hard constraints that must agree in the final matching: M:XorAtMostOne and M:AtMostOne2 consist of $L_P + L_H$ local factors, and M:Budget adds an additional global budget factor to the factor graph of M:AtMostOne2, yielding a more complex overall problem. Faster times would be achieved for smaller values of maximum number of iterations.

Table C.3: Average training and validation time per epoch respective to some strategies used in the paper for each dataset used for highlights extraction.

Strategy	SST	AgNews	IMDB	Beer	Hotels
Training Time (in seconds)					
SPECTRA	20	300	600	85	550
HardKuma	18	180	120	75	220
SFE	10	165	120	40	200
Sparsemax	15	165	120	40	200

Table C.4: Average training and validation time per epoch for each dataset and each strategy used for matchings extraction.

Strategy	SNLI	MNLI
Training Time (in minutes)		
ESIM	26	23
M:XorAtMostOne	53	60
M:AtMostOne2	53	60
M:Budget	56	65
Gumbel	28	23

C.5 SPECTRA Performance for Varying Rationale Length/Alignment Budget

We analyse how SPECTRA fares for different sparsity levels on highlights and matchings extraction (see Table C.5 and Table C.6).

Table C.5: Model predictive performances across datasets and different budget values B for the SPECTRA method for highlights extraction. We report F_1 scores on test sets for all datasets but Beer where we report MSE.

B	SST ↑	AgNews ↑	IMDB ↑	Beer ↓	Hotels ↑
2	0.578	0.870	0.891	0.020	0.875
5	0.726	0.903	0.900	0.019	0.867
10	0.799	0.915	0.891	0.016	0.890
15	0.798	0.916	0.893	0.016	0.906
20	0.803	0.918	0.891	0.017	0.895

Table C.6: Model predictive performances across datasets and different budget values for the SPECTRA method for matchings extraction. We report F_1 scores on test sets for all datasets.

Budget B	SNLI ↑	MNLI ↑
4	0.857	0.731
5	0.851	0.731
6	0.852	0.755