CARNEGIE MELLON UNIVERSITY
LANGUAGE TECHNOLOGIES INSTITUTE

UNIVERSIDADE DE LISBOA
INSTITUTO SUPERIOR TÉCNICO

Understanding, Explaining, and Aligning Neural Models for Machine Translation

Patrick Santos Fernandes

CMU-LTI-25-010
October 2025

Thesis Committee:
Graham Neubig, Chair (Carnegie Mellon University)
André F. T. Martins (Instituto Superior Técnico)
Zachary Lipton (Carnegie Mellon University)
Mário Figueiredo (Instituto Superior Técnico)
Yoon Kim (Massachusetts Institute of Technology)
Chris Dyer (Google DeepMind)

Submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy.

Abstract

The recent success of large neural network models trained on massive amounts of unlabeled data has, yet again, highlighted the power of *scaling*, in terms of both model capacity and data quantity. However, despite its successes, scaling up also adds new additional challenges: (i) the large size and complexity of these models make it difficult to **understand** their learning dynamics and global behavior, and how these change with the data distribution; (ii) their complexity also make it difficult to extract **explanations** of how they arrive at individual predictions, and to know how to leverage these explanations; (iii) this, coupled with the fact that the data used to train these models is often noisy, can lead to models that do not **align** with their intended use case.

This thesis tackles all of the challenges above, with a particular focus on models trained for **machine translation** (MT). In the first part, we study the scaling behaviour of MT models trained to translate between multiple high-resource language pairs (LPs), showing that the way performance for individual LPs scales with model capacity is only dependent on the properties of that *"task"* and is invariant to cross-lingual interactions. We also investigate how much context information MT models leverage in document-level MT, proposing ways to increase it. In the second part, we start by proposing a novel method for extracting *explanations* of how neural models arrive at their predictions, relying on the assumption that good explanations should help other models learn, and leveraging bilevel optimization/meta-learning to learn explainers that teach well. Then we show that not only are *reasoning* models state-of-the-art systems for low-resource MT, but that their *chain-of-thought* rationales are good explanations to help teach smaller LLMs the same task. In the third and final part, we start by showing that, by including metrics of *translation quality* (trained on human feedback annotations) in the inference stage of a trained MT model, we can improve the quality of the translation and align the system towards *human-like* translations. We then also show that we can almost completely remove the dependency on human annotations by leveraging *large language models* to provide (fine-grained) translation quality feedback.

The code for this thesis is available open-source at:

github.com/CoderPat/thesis

Keywords: machine learning, deep learning, natural language processing, machine translation, large language models

Resumo

O recente sucesso de grandes modelos de redes neuronais treinados com enormes quantidades de dados não rotulados destacou, mais uma vez, o poder da *escalabilidade*, tanto em termos de capacidade do modelo quanto de quantidade de dados. No entanto, apesar desses sucessos, a escalabilidade também traz novos desafios adicionais: (i) o grande tamanho e complexidade desses modelos dificultam a **compreensão** das suas dinâmicas de aprendizagem e comportamento global, e como este mudam com a distribuição de dados; (ii) a sua complexidade também dificulta a extração de **explicações** sobre como eles chegam a previsões individuais, e saber como aproveitar essas explicações; (iii) isso, aliado ao facto de que os dados usados para treinar esses modelos são frequentemente ruidosos, pode levar a modelos que não se **alinham** com seu uso pretendido.

Esta tese aborda todos esses desafios, com foco especial em modelos treinados para **tradução automática** (TA). Na primeira parte, estudamos o comportamento de escalabilidade de modelos de TA treinados para traduzir entre múltiplos pares de línguas (PLs) com muitos recursos, mostrando que a forma como o desempenho para PLs individuais escala com a capacidade do modelo depende apenas das propriedades dessa "tarefa" e é invariante às interações entre línguas. Também investigamos quanta informação de contexto os modelos de TA aproveitam na tradução ao nível do documento, propondo maneiras de aumentá-la. Na segunda parte, começamos por propor um método inovador para extrair *explicações* de como os modelos neuronais chegam às suas previsões, baseando-nos na suposição de que boas explicações devem ajudar outros modelos a aprender, e utilizando otimização em dois níveis/meta-aprendizagem para treinar explicadores que ensinem bem. Mostramos também que os modelos de *raciocínio* não só são sistemas de última geração para TA com poucos recursos, mas que as suas *cadeias de pensamento* são boas explicações para ajudar a ensinar LLMs menores para a mesma tarefa. Na terceira e última parte, começamos por mostrar que, ao incluir métricas de *qualidade da tradução* (treinadas com anotações de *feedback* humano) na fase de inferência de um modelo de TA treinado, podemos melhorar a qualidade da tradução e alinhar o sistema com traduções *semelhantes às humanas*. Mostramos também que podemos eliminar quase completamente a dependência em anotações humanas usando *grandes modelos de linguagem* para fornecer feedback sobre a qualidade da tradução.

O código desta tese está disponível como código aberto em:

github.com/CoderPat/thesis

Palavras-chave: aprendizagem automática, aprendizagem profunda, processamento de linguagem natural, tradução automática, grandes modelos de linguagem

Acknowledgments

When I started my PhD more than 5 years ago, I had a vague notion that it would be a challenging and formative experience, but I had *no idea* just how much this quest would push my limits and force me to grow, not only as a researcher, but as a human being. This tale of two continents was only possible thanks to the kindness and support of countless people who, directly or indirectly, contributed to my journey.

A special thanks to both my advisors, Graham and André, whose guidance, kindness, and unwavering belief were instrumental in helping me find my path, especially when it felt very hard to see the road ahead. Graham, your attention to detail, ability to encourage deep questioning, and capacity to see potential where I saw obstacles made me rethink the research process and made me a more thoughtful researcher. André, your spontaneous ingenuity, ambitious planning, and relaxed optimism inspired a greater sense of possibility and made me a more daring researcher. Together, you encouraged me to take intellectual risks and showed trust when sharing responsibility. I'm truly lucky to have had you as a pair and, between conferences and car rides together, to call you friends.

I'll forever remember my time in Pittsburgh. To all the people I met at CMU, students, professors and administrators, and especially the other NeuLab members. Thank you for making me feel welcome and providing such a world-class research environment, and for the friendships I carried across the Atlantic. And to fellow residents of *Consulado*. Between house parties, brunches, and couch hangouts, we shared a connection in that *home* I'll never forget, and allowed my time in the States to be about so much more than research.

My time at Google and California allowed me to see the epicenter of action and large-scale research. I'm grateful to all my collaborators and friends there, who shaped me and my research, and a special thanks to Behrooz and Markus for guiding and allowing me to fully leverage the opportunities such an environment created.

I never expected to still be at IST more than ten years after first entering it, as practically a child. A thank you to all my SARDINE and Unbabel friends, who showed me I could come back home without sacrificing the quality of the research I could do, or the world-class talent of my collaborators. And the fun we had and brought abroad to conferences was something special, and only fuelled my growth as a researcher.

To all my family, and especially my parents, who provided unconditional support even when they didn't fully understand what I was doing or why it was taking so long, and my brother, who has always shared my enthusiasm for trying to *know*. And to all my friends *da zona*, who have accompanied and supported me even before I joined university.

Finally and especially, a thank you to Rafaela, my girlfriend and love of my life. You were always the one closest through the thick and thin, even if there was an ocean separating us. I wouldn't have been able to finish or even start this journey without your support, and you share in this achievement with me.

Contents

Prologue

Chapter 1

Introduction

The field of natural language processing (NLP) has seen explosive growth in interest and successful applications in recent years, largely driven by the efficacy of parametric models of language. In particular, massive neural network models (often with billions of parameters) leveraging massive amounts of textual data from the Internet (frequently surpassing trillions of words) have been able to match or even surpass human performance on a variety of natural language tasks. In machine translation (MT), for example, state-of-the-art systems are now able to translate between *high-resource* languages (e.g. English, German, etc.) with near-human quality (Kocmi et al., 2022).

However, despite the overall success of *scaling* the compute used (either by increasing the model capacity or by training on more data) at improving the general capabilities of these models, it also adds new additional challenges to their widespread adoption (besides the computational ones):

- Although we know that scaling works for most tasks[1], we lack principled understanding of *how* performance scales with model size, data quantity and distribution of domains/tasks in the data, and of *if* models actually leverage (contextual) information they have access to (during training and inference). This makes it difficult to predict and optimize model behaviour.
- Their size, coupled with the nature of neural network computations, makes it near impossible to get a mechanistic, causal understanding of how they work, and getting interpretable and *"faithful"* non-mechanistic explanations of their decisions is still an open problem. This adds to difficulty in understanding how to improve them, and in trusting them in safety-critical applications, like humanitarian and medical emergencies (Mehandru et al., 2023).
- The amount of data required to train these models is so large that manually-curated data is insufficient, and they have to be trained on data *"scraped"* from the inter-

[1] There are notable exceptions where performance seems can decrease with scale, a phenomenon known as *"inverse scaling"* (McKenzie et al., 2024).

net, loosely filtered with automated procedures. This leads to models that are often trained on *"noisy"* data (e.g. containing sentence pairs that are not actually translations of each other, or inappropriate harmful stereotypes) and that don't completely align with their intended use case.

In this thesis, we aim to tackle all these challenges by studying and training neural network models primarily for machine translation, a task that has been extensively studied for decades and that has contributed a number of technical innovations there were key to success of the broader NLP landscape (Bahdanau et al., 2015; Vaswani et al., 2017a). But while we focus on MT, most of the techniques and insights we develop are broadly applicable to other NLP tasks leverage neural networks, and can inform the development of more reliable, interpretable, and efficient general-purpose language models.

1.1 Thesis Overview

We start by providing a brief overview of the preliminary concepts, background and related work for the topics covered in this thesis (Chapter 2). The rest of the thesis is then divided into three parts, each tackling one of the challenges mentioned above.

Part I: Understanding the Behaviour In the first part, we focus on improving our understanding of the *global behaviour* of neural networks: statistical patterns and regularities that describe how these models behave during training and inference, and how these relate to various design choices and data used. First (Chapter 3), we study the scaling behavior of *multilingual* neural machine translation models, trained to translate between multiple language pairs, and how it changes based how their data (which been seen as different *tasks*) is weighted during training. By analysing dozens of *scaling laws* fitted on different distributions, we find that performance on individual translation tasks improves with model capacity at ratios that are independent of how the different tasks are weighted in the training. This allows us to propose a *joint* scaling law that can predict the full Pareto frontier of translation quality as we scale the mode. Then (Chapter 4), shifting our focus to document-level machine translation, we investigate how much contextual information beyond the sentence level these models actually leverage. We propose an information-theoretic metric to quantify context usage and find that models overall leverage the *target* language context more than the *source* one and propose a method to increase the usage of extra-sentential context.

Part II: Explaining as Teaching We then move onto the problem of extracting *local explanations* from neural network models: interpretable descriptions of how/why a model arrived at a specific decision/output given a particular input. Although many definitions of what makes a good explanation have been proposed, we focus on a quantitatively measurable definition based on the assumption that good explanations should help other models *learn*. Based on it, we propose a novel method for extracting explanations from trained neural network models, leveraging bilevel optimization/meta-learning to learn (parametrized) explainers that help teaching student models (Chapter 5). We show not only that our method is able to extract explanations with higher *simulability* (i.e. that help students simulate the teacher) than other methods, but that this explanations are more *plausible*, aligning with human explanations. Then (Chapter 6), we investigate if *reasoning* models, optimized to solve complex tasks by generating intermediate reasoning steps (*chains-of-thought*), generalize to machine translation, finding that not only are these models state-of-the-art systems for low-resource MT, but that their *chain-of-thought* rationales are good explanations (according to the aforementioned definition) and that we can leverage them to improve the quality of smaller LLMs at the same task without requiring additional *test-time compute*.

Part III: Aligning with Feedback In the final part, we then tackle the problem of misaligned behaviour in machine translation models trained on large corpora of (noisy) translation-pairs. We propose a *quality-aware* decoding algorithm that not only consider model-likelihood but also metrics of translation quality, effectively aligning the model with the goal of *human-like translation* (Chapter 7). To do so, we propose incorporating recent reference-based and reference-free *neural, trained* metrics (trained to align with human judgement of translation quality) into the decoding algorithm through techniques such as N-best reranking and *minimun Bayes risk* decoding. We show that models that leverage these metrics can produce more human-like translations (even if traditional automatic metrics would disagree). After (Chapter 8), we also show that LLMs can be leveraged as state-of-the-art *fine-grained* translation quality metrics, with only a few human-annotated translation quality examples (in contrast to previous trained neural metrics which require many orders of magnitude more data), and hinting at the possibility of using LLMs to align machine translation models.

And while each part can be read largely independently, we hope that this thesis shows the interconnected nature of these fundamental challenges .

1.2 Summary of Publications

Parts of the background and related work in Chapter 2 appear in the survey:

> **Fernandes**, P., Madaan, A., Liu, E., Farinhas, A., Martins, P. H., Bertsch, A., de Souza, J. G. C., Zhou, S., Wu, T., Neubig, G., & Martins, A. F. T. (2023). Bridging the gap: A survey on integrating (human) feedback for natural language generation. *Transactions of the Association for Computational Linguistics*. https://arxiv.org/abs/2305.00955

The content of Chapter 3 appears in:

> **Fernandes**, P., Ghorbani, B., Garcia, X., Freitag, M., & Firat, O. (2023, 23–29 Jul). Scaling laws for multilingual neural machine translation. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th international conference on machine learning* (pp. 10053–10071, Vol. 202). PMLR. https://proceedings.mlr.press/v202/fernandes23a.html

The content of Chapter 4 appears in:

> **Fernandes**, P., Yin, K., Neubig, G., & Martins, A. F. T. (2021, August). Measuring and increasing context usage in context-aware machine translation. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *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)* (pp. 6467–6478). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.505

The content of Chapter 5 appears in:

> **Fernandes**, P., Treviso, M. V., Pruthi, D., Martins, A. F. T., & Neubig, G. (2022). Learning to scaffold: Optimizing model explanations for teaching. *Advances in Neural Information Processing Systems*. https://openreview.net/pdf?id=V5rlSPsHpkf

The content of Chapter 7 appears in:

> **Fernandes**, P., Farinhas, A., Rei, R., C. de Souza, J. G., Ogayo, P., Neubig, G., & Martins, A. (2022, July). Quality-aware decoding for neural machine translation. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Proceedings of the 2022 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 1396–1412). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.100

The content of Chapter 8 appears in:

> **Fernandes**, P., Deutsch, D., Finkelstein, M., Riley, P., Martins, A. F. T., Neubig, G., Garg, A., Clark, J. H., Freitag, M., & Firat, O. (2023). The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. *Proceedings of the Seventh Conference on Machine Translation (WMT)*

Non-thesis research I have also pursued other research directions, which have lead to the following first-author publications (these are excluded for the remainder of this thesis):

> **Fernandes**, P., Yin, K., Liu, E., Martins, A., & Neubig, G. (2023, July). When does translation require context? a data-driven, multilingual exploration [Awarded **Best Resource Paper**]. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.36
>
> **Fernandes**, P., Agrawal, S., Zaranis, E., Martins, A. F. T., & Neubig, G. (2025). Do llms understand your translations? evaluating paragraph-level mt with question answering. *Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2504.07583

Additionally, I have also contributed to the following publications as a non-primary author:

> Yin, K., **Fernandes**, P., Pruthi, D., Chaudhary, A., Martins, A. F. T., & Neubig, G. (2021, August). Do context-aware translation models pay the right attention? In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *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)* (pp. 788–801). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.65
>
> Treviso, M., Góis, A., **Fernandes**, P., Fonseca, E., & Martins, A. (2022, May). Predicting attention sparsity in transformers. In A. Vlachos, P. Agrawal, A. Martins, G. Lampouras, & C. Lyu (Eds.), *Proceedings of the sixth workshop on structured prediction for nlp* (pp. 67–81). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.spnlp-1.7
>
> Yan, B., **Fernandes**, P., Dalmia, S., Shi, J., Peng, Y., Berrebbi, D., Wang, X., Neubig, G., & Watanabe, S. (2022, May). CMU's IWSLT 2022 dialect speech translation system. In E. Salesky, M. Federico, & M. Costa-jussà (Eds.), *Proceedings of the 19th international conference on spoken language translation (iwslt 2022)* (pp. 298–307). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.iwslt-1.27
>
> Sun, J., **Fernandes**, P., Wang, X., & Neubig, G. (2023, May). A multi-dimensional evaluation of tokenizer-free multilingual pretrained models. In A. Vlachos & I. Augenstein (Eds.), *Findings of the association for computational linguistics: Eacl 2023* (pp. 1725–1735). Association for Computational Lin-

guistics. https://doi.org/10.18653/v1/2023.findings-eacl.128

Farinha, A. C., Farajian, M. A., Buchicchio, M., **Fernandes**, P., C. de Souza, J. G., Moniz, H., & Martins, A. F. T. (2022, December). Findings of the WMT 2022 shared task on chat translation. In P. Koehn, L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussà, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. Jimeno Yepes, T. Kocmi, A. Martins, M. Morishita, … M. Zampieri (Eds.), *Proceedings of the seventh conference on machine translation (wmt)* (pp. 724–743). Association for Computational Linguistics. https://aclanthology.org/2022.wmt-1.70

Yan, B., Shi, J., Tang, Y., Inaguma, H., Peng, Y., Dalmia, S., Polák, P., **Fernandes**, P., Berrebbi, D., Hayashi, T., Zhang, X., Ni, Z., Hira, M., Maiti, S., Pino, J., & Watanabe, S. (2023, July). ESPnet-ST-v2: Multipurpose spoken language translation toolkit. In D. Bollegala, R. Huang, & A. Ritter (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 3: System demonstrations)* (pp. 400–411). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-demo.38

Jain, S., Keshava, V., Mysore Sathyendra, S., **Fernandes**, P., Liu, P., Neubig, G., & Zhou, C. (2023, July). Multi-dimensional evaluation of text summarization with in-context learning. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Findings of the association for computational linguistics: Acl 2023* (pp. 8487–8495). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.findings-acl.537

Ramos, M., **Fernandes**, P., Farinhas, A., & Martins, A. (2024, June). Aligning neural machine translation models: Human feedback in training and inference. In C. Scarton, C. Prescott, C. Bayliss, C. Oakley, J. Wright, S. Wrigley, X. Song, E. Gow-Smith, R. Bawden, V. M. Sánchez-Cartagena, P. Cadwell, E. Lapshinova-Koltunski, V. Cabarrão, K. Chatzitheodorou, M. Nurminen, D. Kanojia, & H. Moniz (Eds.), *Proceedings of the 25th annual conference of the european association for machine translation (volume 1)* (pp. 258–274). European Association for Machine Translation (EAMT). https://aclanthology.org/2024.eamt-1.22/

Alves, D. M., Pombal, J., Guerreiro, N. M., Martins, P. H., Alves, J., Farajian, A., Peters, B., Rei, R., **Fernandes**, P., Agrawal, S., Colombo, P., de Souza, J. G. C., & Martins, A. F. T. (2024). Tower: An open multilingual large language model for translation-related tasks. *Conference on Language*

Modeling COLM. https://openreview.net/forum?id=EHPns3hVkj

Agrawal, S., Farajian, A., **Fernandes**, P., Rei, R., & Martins, A. F. T. (2024). Assessing the role of context in chat translation evaluation: Is context helpful and under what conditions? *Transactions of the Association for Computational Linguistics*, *12*, 1250–1267. https://doi.org/10.1162/tacl_a_00700

Agrawal, S., De Souza, J. G. C., Rei, R., Farinhas, A., Faria, G., **Fernandes**, P., Guerreiro, N. M., & Martins, A. (2024, November). Modeling user preferences with automatic metrics: Creating a high-quality preference dataset for machine translation. In Y. Al-Onaizan, M. Bansal, & Y.-N. Chen (Eds.), *Proceedings of the 2024 conference on empirical methods in natural language processing* (pp. 14503–14519). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.emnlp-main.803

Rei, R., Pombal, J., Guerreiro, N. M., Alves, J., Martins, P. H., **Fernandes**, P., Wu, H., Vaz, T., Alves, D., Farajian, A., Agrawal, S., Farinhas, A., C. De Souza, J. G., & Martins, A. (2024, November). Tower v2: Unbabel-IST 2024 submission for the general MT shared task. In B. Haddow, T. Kocmi, P. Koehn, & C. Monz (Eds.), *Proceedings of the ninth conference on machine translation* (pp. 185–204). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.wmt-1.12

Faysse, M., **Fernandes**, P., Guerreiro, N. M., Loison, A., Alves, D. M., Corro, C., Boizard, N., Alves, J., Rei, R., Martins, P. H., Casademunt, A. B., Yvon, F., Martins, A. F. T., Viaud, G., Hudelot, C., & Colombo, P. (2025). Croissantllm: A truly bilingual french-english language model. *Transactions on Machine Learning Research*. https://openreview.net/forum?id=uA19Xo1o31

Wu, I., **Fernandes**, P., Bertsch, A., Kim, S., Pakazad, S., & Neubig, G. (2025). Better instruction-following through minimum bayes risk. *International Conference on Learning Representations ICLR*. https://openreview.net/forum?id=7xCSK9BLPy

Martins, P., **Fernandes**, P., Alves, J., Guerreiro, N., Rei, R., Alves, D., Pombal, J., Farajian, A., Faysse, M., Klimaszewski, M., Colombo, P., Haddow, B., De Souza, J., Birch, A., & Martins, A. (2025). Eurollm: Multilingual language models for europe [Published by Elsevier B.V.; 2nd EuroHPC user day, EuroHPC 2024]. *Procedia Computer Science*, *255*, 53–62. https://doi.org/10.1016/j.procs.2025.02.260

Boizard, N., Gisserot-Boukhlef, H., Alves, D. M., Martins, A., Hammal, A., Corro, C., Hudelot, C., Malherbe, E., Malaboeuf, E., Jourdan, F., Hautreux, G., Alves, J., El-Haddad, K., Faysse, M., Peyrard, M., Guerreiro, N. M., **Fernandes**, P., Rei, R., & Colombo, P. (2025). Eurobert: Scaling multilingual encoders for european languages. *Proceedings of the Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2503.05500

Ramos, M., **Fernandes**, P., Agrawal, S., & Martins, A. F. T. (2025). Multilingual contextualization of large language models for document-level machine translation. *Proceedings of the Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2504.12140

Pombal, J., Yoon, D., **Fernandes**, P., Wu, I., Kim, S., Rei, R., Neubig, G., & Martins, A. F. T. (2025). M-prometheus: A suite of open multilingual llm judges. *Proceedings of the Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2504.04953

Liu, E., Bertsch, A., Sutawika, L., Tjuatja, L., **Fernandes**, P., Marinov, L., Chen, M., Singhal, S., Lawrence, C., Raghunathan, A., Gashteovski, K., & Neubig, G. (2025). Not-just-scaling laws: Towards a better understanding of the downstream impact of language model design decisions. *The 2025 Conference on Empirical Methods in Natural Language Processing*. https://openreview.net/forum?id=XgGX19LJ2e

Chapter 2

Preliminaries

In this chapter, we provide an overview of the main concepts necessary to understand and motivate the remainder of this thesis. We start by introducing the (conditional) language modeling task and neural language models (looking at neural machine translation as a specific case), discussing how *scaling* (of model size and data) and *transfer* (of learned representations across tasks) have underpinned much of the recent success in NLP (Section 2.1). We then discuss the *interpretability* problem with neural models (Section 2.2), common approaches to extract explanations from them, and one approach to define and quantitatively measure how *good* an explanation of a model's decision is. Finally, we discuss how *feedback* can be used to align and further improve the performance of neural language models (Section 2.3).

2.1 Learning Parametric Models of Language

Language modelling is the task of estimating the probability of sequences of words or symbols occurring in a language of interest. We might want, for example, to estimate the probability of a sentence appearing in the English language, or the probability of a sequence of notes appearing in a piece of music.

Formally, given a sequence of tokens $y = (y_1, \ldots, y_T)$ from some vocabulary $y_i \in \mathcal{V}$, the goal is to estimate the probability of the sequence $p(y) = p(y_1, \ldots, y_T)$. To simplify the task, it is common to *autoregressively* factorise the probability of a sequence as the product of the probabilities of each token in the sequence *conditioned on the previous ones*, i.e.

$$p(y_1, \cdots, y_T) = \prod_{i=1}^{T} p(y_i \mid y_{i-1}, \cdots, y_1) \tag{2.1}$$

To tractably solve this task, one can estimate each conditional probability using a *parametric* model $p_\theta(y_i \mid y_{i-1}, \cdots, y_1)$, with parameters θ. The problem then reduces to choosing a parametric model that is expressive enough to capture the distribution of interest, and to estimate its parameters θ *from data*.

The latter is often tackled through an optimisation perspective: given some training samples from language $\mathcal{D} = \{y^{(i)}\}_{i=1}^{N}$, the goal is to find the parameters $\theta^\star$ that maximise the likelihood of language samples or, equivalently, minimise the *negative log-likelihood* of the data:

$$\theta^\star = \operatorname*{argmin}_{\theta} \mathcal{L}(\theta) \tag{2.2}$$

$$\mathcal{L}(\theta) = \mathbb{E}_{y \sim \mathcal{D}} \left[-\sum_{i=1}^{T} \log p_\theta(y_i | y_{<i}) \right] \tag{2.3}$$

Once the parameters $\theta^\star$ are estimated, language models can be used to, not only estimate the probability of a sequence but also generate new sequences in the language by, for example, sampling from the model distribution $\hat{y}_i \sim p_{\theta^\star}(y_i|\hat{y}_{i-1}, \cdots, \hat{y})$. Additionally, language modelling can be extended to the *conditional* case, where the goal is to estimate the probability $p(y \mid x)$ of a sequence y given another sequence x (from the same or another language).

While conceptually a simple task, most problems in NLP can be framed as (conditional) language modelling (Raffel et al., 2020a). Language models can be evaluated through their *perplexity* of some unseen data but, in practice, the goal is often to use language models to solve other tasks and, as such, task-specific metrics are generally used to evaluate their performance.

2.1.1 Neural Language Models

The choice of the parametric model plays an important role in the difficulty and approach to solving Equation 2.2. While early approaches to language modelling relied on *count-based* models, such as n-gram models, the current dominant approach is to use *neural networks* to parameterise the conditional (token) probabilities (Bengio et al., 2003).

Formally, a neural language model parameterises the conditional probability of tokens at each step as

$$p_\theta(y_i \mid y_{i-1}, \cdots, y_1) = \text{NORMALISE}(f_\theta(y_1, \cdots, y_{i-1})), \tag{2.4}$$

where $f_\theta : \mathcal{V}^* \to \mathbb{R}^{|\mathcal{V}|}$ denotes a neural network model parameterised by θ that returns scores or *logits* for each next possible tokens, and NORMALIZE : $\mathbb{R}^{|\mathcal{V}|} \to \Delta^{|\mathcal{V}|-1}$ normalises these scores into a probability distribution, typically using a SOFTMAX function:

$$\text{SOFTMAX}(z)_i = \frac{\exp(z_i)}{\sum_{j=1}^{|\mathcal{V}|} \exp(z_j)}. \tag{2.5}$$

The core advantage of using neural networks is that, by defining a suitable loss function,

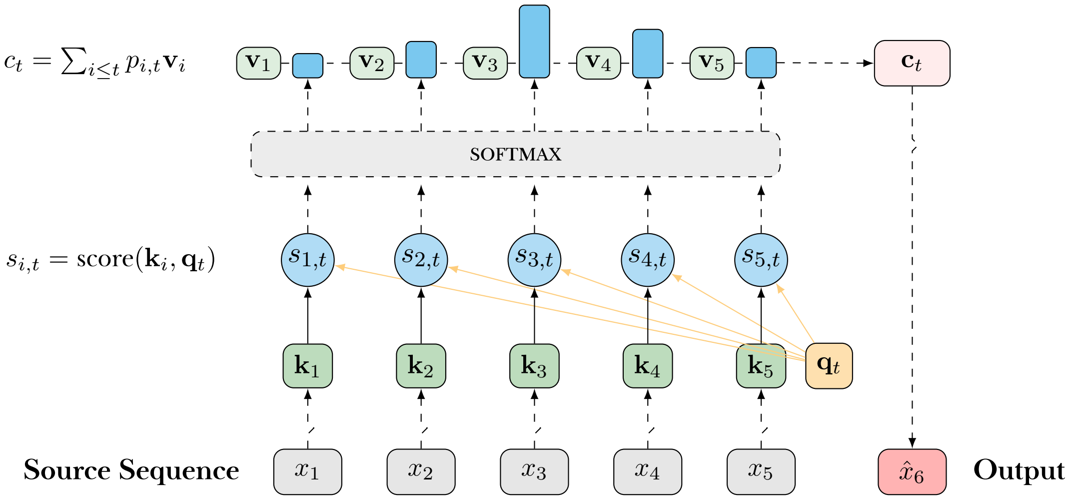

Figure 2.1: Simplified illustration of the attention mechanism, the main component of the *Transformer* architecture.

the parameters θ can be tractably estimated with *stochastic gradient descent*, with gradients computed using the *backpropagation* algorithm (Rumelhart et al., 1986). In the case of neural language models, this loss is precisely the negative log-likelihood of the data, as defined in Equation 2.3.

Notice that, since sentences in a language can have different lengths, the neural network model f_θ must handle variable-length sequences, something standard *feedforward* neural networks are not capable of. While *recurrent* neural networks (RNNs) (Hochreiter & Schmidhuber, 1997) have traditionally been used for this purpose, the current state-of-the-art neural language models generally use attention-based models, such as Transformers (Vaswani et al., 2017a). Figure 2.1 shows an simplified illustration of the mathematical operations behind the attention mechanism.

2.1.2 Neural Machine Translation

Machine Translation (MT) is the task of translating a sentence or document from a source language to a target language, intending to convey the same meaning as closely as possible. While earlier approaches to MT used rule-based systems and statistical models, the currently dominant approach is to frame MT as a *conditional language modelling* task, and use neural language models to solve the task (Sutskever et al., 2014). This is known as neural machine translation (NMT).

Formally, given a dataset of sentence pairs $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^N$, where $x^{(i)}$ and $y^{(i)}$ are sentences in the source and target languages respectively, NMT models are trained to minimise the *conditional* negative log-likelihood of the target sentences given the source sentences:

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,y)\sim\mathcal{D}} \left[-\sum_i \log p_\theta(y_i | y_{<i}, x) \right] \tag{2.6}$$

Once an NMT model is trained, one typically obtains translations from it by attempting to find the *most likely* translation given a source sentence x, i.e. $\hat{y} = \mathrm{argmax}_y\, p_\theta(y|x)$[1]. Since this problem is intractable, one typically uses approximate search algorithms, such as *beam search* (Graves, 2012b).

Traditionally, NMT systems used *encoder-decoder* architectures, where the neural network was "partitioned" into two distinct components: an *encoder* that processes the source sentence x into a contextualized representation, and a *decoder* that generates the target translation y conditioned on these source representations (Cho et al., 2014; Vaswani et al., 2017a). However, more recently, simpler *decoder-only* architectures, where both source and target sequences are processed sequentially, have started to dominate, as they allow an easier transfer of the universal representations learned from *pretrained* large language models to the translation task (Alves et al., 2024) (see the discussion in the next subsection).

The gold standard to evaluate the quality of NMT systems is to use *human evaluation*, where bilingual (professional) translators compare and score the system's translated sentences according to how well they convey the same meaning as the original source sentence (Freitag, Foster, et al., 2021). However, this is impractical for routine evaluation, and automatic metrics are the most common approach to evaluate machine translation. While early MT evaluation relied predominantly on *lexical* metrics, such as BLEU (Papineni et al., 2002a), recent work has shown that *neural* metrics trained to correlate with human judgments of translation quality, such as COMET (Rei et al., 2020a) and BLEURT (Sellam et al., 2020), are more effective at capturing the quality of translations (Freitag et al., 2022), and are now the standard for automatic evaluation.

2.1.3 Scaling & Transfer

For much of NLP's history, the typical paradigm was to treat each task as a separate problem, with dedicated datasets and specific models designed, or at least separately trained, for each task. Moreover, these models were relatively small (very rarely surpassing 1 billion parameters), as increasing the model size was generally believed to lead to *overfitting* to the task-specific training dataset.

However, in recent years the paradigm has shifted towards training large *pretrained* models with a language modelling objective on a large corpus of text (generally from the internet) since these models tend to perform well on a variety of tasks after a subsequent fine-tuning stage on small amounts of task-specific training data (Radford & Narasimhan, 2018), or even without any additional training or data (Brown et al., 2020a)!

Underpinning this shift are two empirical observations that have proven key in the success of pretrained models:

[1]While in MT this decoding objective tends to obtain higher-quality translations, it also possible decode sampling from the distribution of possible translations, like in other language modelling tasks (§7)

- **Scaling** the amount of compute use to train these models, either by increasing the model size or the amount of training data, is one of the most successful approaches to improve the performance of neural (language) models. In fact, the relationship between scale and performance follows predictable patterns known as *scaling laws*, which tend to hold for a wide range of models and tasks (Kaplan, McCandlish, Henighan, et al., 2020). For example, for a fixed amount of training data, the perplexity of a language model is expected to decrease as its size increases following

$$\mathcal{L}(N) = \beta N^{-\alpha} + L_{\infty}, \tag{2.7}$$

  where β and α are constants and L_{∞} are parameters that can be estimated by training (smaller) models on that data (see $3.2.1 for more details).

- **Transfer**, where neural language models that achieve good performance (low perplexity) on modelling a specific language (i.e English) tend to learn "universal representations" of that language. When these representations are leveraged through *transfer learning* to solve other tasks, they lead to models that are more sample-efficient and perform better than models trained from scratch (Raffel et al., 2020b), and they can even be used directly through *zero-shot prompting* or *in-context learning* (Brown et al., 2020a). Importantly, this transfer capability is significantly enhanced when models are trained on diverse data distributions or even across multiple languages .

These observations have recently culminated in the development *large language models* (LLMs), models with billions of parameters trained on internet-scale data, which have shown remarkable performance across a wide range of NLP tasks (OpenAI, 2023), including generating high-quality translations (Vilar et al., 2022).

2.2 Interpreting Neural Network's Decisions

The widespread adoption of neural networks across fields has been accompanied by an increased demand for *understanding* the behaviour of these models work, and to provide *explanations* for particular decisions and generations they make (Lipton, 2016b). However, the complex and highly nonlinear nature of neural network models makes it almost impossible to get a "causal" understanding of how inputs to the model lead to outputs, and the challenge is exacerbated with large models.[2]

Therefore, researchers have developed various methods for extracting *post-hoc* explanations from trained neural network models, which can be used to provide insights into how the model works and to provide explanations for particular decisions.

[2] *Mechanistic interpretability* is an ambitious attempt to get such causal understanding by reverse-engineering the internal circuits and algorithms of neural networks, but faces significant scalability challenges.

2.2.1 Extracting Local Explanations

What exactly is an "explanation" for a particular depends on the specific task, application and audience. However, in general, we are interested in understanding how the model uses its input to make a decision or generate output. In particular, *local* explanations aim to understand and interpret individual predictions or outputs, as opposed to *global* explanations which aim to reveal insights about the model's overall behaviour.

Feature Attribution One common approach to extracting local explanations from neural models is through *feature attribution* methods. These methods aim to distribute the importance of a prediction over the input features (for example, words in a sentence or pixels in an image) to get an intuition of which parts of the input have contributed to the decision or output. Table 2.1 shows an example of a feature attribution method applied to a text classification task.

Input	*The thesis is original and has interesting insights, but could be better written.*
Output	Distinction.
Explanation	The thesis is original and has interesting insights, but could be better written.

Table 2.1: Example of a thesis review classification task, showing *per-word attribution*.

A popular method for feature attribution is the *gradient-based* method which attributes the importance of each input feature based on the gradient of the output concerning that input feature. Formally, given a model f_θ that for an input $x = (x_1, \cdots, x_n)$ outputs a prediction $f_\theta(x)$ the gradient-based *score*

$$\mathbf{g}(x_i) = \|\nabla_{x_i} f_\theta(x)\|_2 \tag{2.8}$$

is a measure of how changing the feature x_i affects the output $f_\theta(x)$, and can be interpreted as the importance of that feature to the final prediction. Many variations of these gradient-based methods have been proposed (Denil et al., 2014; Sundararajan et al., 2017a), and they are commonly used to provide explanations since they can be used with any differentiable (neural network) model.

Alternatively, the prevalence of transformer-based models in NLP has led to feature attribution methods that leverage the attention mechanism in these models to provide explanations. Each attention mechanism naturally relies on computing, for each input/output feature x_i, distribution of scores over all input features that input "attends" to

$$\alpha_i = \textsc{softmax}\left(q_i K\right) \tag{2.9}$$

where $q_i \in \mathbb{R}^d$ is a query vector for the output feature x_i, and $K \in \mathbb{R}^{n\times d}$ is a matrix of key vectors for the input features. These attention scores can then be aggregated to provide an importance score for each input feature, for example, by picking a relevant input feature (for example, the first or last token in a sentence) and summing or averaging the attention scores produced in all the attention heads or from a given layer (Vashishth et al., 2019; Wiegreffe & Pinter, 2019).

Chain-of-Thought One interesting property of LLMs is that their performance on downstream tasks can generally be improved by prompting these models to generate a series of intermediate reasoning steps (*"Let's think step-by-step"*) (Wei et al., 2022). While this *chain-of-thought* phenomenon is not fully understood and has been primarily used to improve model performance, the intermediate generations can be interpreted as *free-form* explanations of the model's decision and can be representative of the model's underlying decision-making process (Lanham et al., 2023). Table 2.2 shows an example of a chain-of-thought explanation applied to a review classification task.

Input	*The thesis is original and has interesting insights, but could be better written.*
Output	Distinction.
Explanation	Let's think step-by-step:
	(1) Described as original and containing interesting insights, strong positive indicators.
	(2) The reviewer notes that it could be better written, which suggests a flaw in clarity.
	(3) Overall, writing is significantly less important than contributions.
	Therefore, it probably passes the bar for Distinction.

Table 2.2: Example classification task, showing the *chain-of-thought* explanation.

2.2.2 Evaluating Explanation Quality

While there are many methods for extracting explanations from neural models, the question of what method provides better or worse explanations and which one to use is not straightforward. This is exacerbated by the fact that, often, different methods produce explanations that disagree with each other for the same model and input (Neely et al., 2021).

Commonly, the quality of an explanation is evaluated according to its *plausibility*, by comparing it to some *ground truth* explanation provided by a human (given the same input and output). However, this assumes that the model's decision process should always match the human process, an assumption that can lead to *unfaithful* explanations that don't represent the model's actual decision process (Pruthi, Gupta, et al., 2020). Instead, alternative quantitative measures of explanation quality that don't rely on human-provided explanations have been proposed (DeYoung et al., 2020; Treviso et al., 2022).

In particular, Pruthi et al. (2022) proposed a *simulability*-based framework to evaluate the quality of an explanation: given a *teacher* model and explanations for its decisions, a *student* model S with parameters θ_S is trained to predict the teacher's decisions using *and* its explanations

$$\mathcal{L}_S(\theta_S) = \mathbb{E}_{(x,y,e)\sim\mathcal{D}}\left[-\log p_{\theta_S}(e_T, y_T \mid x)\right], \quad e_t, y_t = T(x) \tag{2.10}$$

where $\mathcal{D}$ is a dataset of input and decisions and explanations from a teacher model T, and ϕ_S are the parameters of the student model. The quality of the explanation is then evaluated by the performance of the student model at predicting the teacher's decisions on some held-out data, and compared to a student model that is trained without explanations.

Intuitively, this framework assumes that a *good* explanation of the model's underlying decision process should help the student model learn to simulate the teacher's decisions better than a model that is trained without explanations. This instrumentalist assumption also implies that good explanations might provide a valuable tool to improve new models' performance.

However, under this instrumental definition, any auxiliary information that improves the student's ability to reproduce the teacher's outputs could be regarded as an explanation, including signals that aren't inherently human-interpretable. For example, in model distillation, student models often improve performance when trained to match the teacher's full output distribution (via KL divergence). From a human-centered perspective, treating the teacher's logits as explanations might seem counter-intuitive: while they are highly informative for another model, they are unlikely to convey meaningful reasons that a human could understand or act upon. This apparent oddity underscores a fundamental corollary of the instrumental approach: there is no universal notion of what constitutes a *"good"* explanation, as this always depends on the intended consumer of that explanation. Nevertheless, while in principle we should measure explanation quality using human students (Arora et al., 2022), we will later show that even using machine students can provide informative signals about human interpretability (Section 5.5).

2.3 Aligning Neural Models with Feedback

The interpretability challenges discussed above are compounded by another fundamental issue with the scaling approach: the massive datasets required to train these models are generally scraped from the internet, and inevitably contain biases, errors, and harmful content. This, coupled with optimizing an objective function that simply maximizes the likelihood of this data, can lead to a *misspecification* of target behaviour (Kenton et al., 2021), and to models that generate toxic, inaccurate, and unhelpful content (Bender et al., 2021; Sheng et al., 2019), issues that might persist even after task-specific fine-tuning of pre-trained models. To address this, recent works have proposed moving from the traditional paradigm of static supervised learning, and instead also incorporating **human feedback** on model-generated outputs during the training or inference process. This move has led to the emergence of the first widely-used general-purpose language assistants (OpenAI, 2023).

2.3.1 Defining Human Feedback

Formally, we consider **human feedback** to be a family of functions $\mathcal{H}$ such that $h \in \mathcal{H}$ takes an input $x \in \mathcal{X}$ and one or more outputs $y_1, \cdots, y_n \in \mathcal{Y}$ (for example, by sampling from a trained language model) and returns some *feedback* $f \in \mathcal{F}$:

$$h : \mathcal{X} \times \underbrace{\mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n}_{n} \to \mathcal{F}. \tag{2.11}$$

A simple example of a feedback function is asking humans to say if, given an input, a particular output is good or bad ($h : \mathcal{X} \times \mathcal{Y} \to \{0, 1\}$). However, more complex feedback functions, such as rankings or natural language feedback, are also commonly used (see Table 2.3 for examples). The choice of format has implications on the expressivity of the feedback, the ease of its collection, and how we can use it to improve systems, and the level of *rationality* of said feedback is heavily impacted by this choice (Ghosal et al., 2023).

This feedback is collected to *align* the model's behaviour with some (often ill-defined) *goal* behaviour: for example, we might want our summarization model to generate summaries that contain all core information, even if it means they are longer; or in *commercial* machine translation, where we want to ensure that models do not mistranslate business-critical information and have good *translation quality*. This **alignment objective** has been studied extensively in the *AI safety and alignment* literature (Amodei et al., 2016; Bostrom, 2014; Kenton et al., 2021) , but can be broadly framed and studied according to two sub-objectives and how they interact: increasing the model's *helpfulness* and/or its *harmlessness* (Bai, Jones, et al., 2022).

Input	Output(s)	Feedback	Type
A melhor comida do mundo é a portuguesa.	*The worst food in the world are Portuguese.*	0.7	Score
		'worst': `major/accuracy` 'are': `minor/fluency`	MQM
		'worst' → 'best', 'are' → 'is'	Post-Edition
Artificial intelligence has the potential to revolutionize industries (...) but ethical concerns need to be handled.	*AI can change industries.*	`Fluency:` 1 `Relevance:` 0.7	Multi-Aspect
		"Misses the ethical concerns."	Natural Language
Explain the moon landing to a 6 year old	A: *People went to the ...* B: *The moon is a satellite...*	A > B	Ranking

Table 2.3: Example input and output for three tasks (machine translation, summarization, and instruction following) and possible different (example) feedback that can be given.

2.3.2 Leveraging Feedback to Improve Generation

In an ideal scenario, we would directly leverage human feedback to improve generation. One way to use it is by optimizing the model parameters directly. This requires the feedback to be "optimizable", *i.e.*, possibly formulated as an optimization problem based on which we can obtain an improved model. For instance, if the feedback is a numerical preference score ($f \in \mathbb{R}$), we can create the following optimization problem:

$$\theta^{\star} = \underset{\theta}{\operatorname{argmax}} \, \mathbb{E}_{x \sim \mathcal{D}}[h(x, \hat{y})] \,, \quad y = m_{\theta}(x) \tag{2.12}$$

where $\mathcal{D}$ is the distribution of possible inputs and $\hat{y}$ is an output generated by the model.

Various techniques have been suggested to optimize θ using the collected human feedback. The most popular approach is to use *reinforcement learning* (RL), where the feedback is used as a reward signal, and use common RL algorithms such as REINFORCE (Williams, 1992), which updates the parameters with the gradients. However, alternatives to RL exist, such as jointly modelling the feedback and language from a dataset $\mathcal{D} = \{(x^{(i)}, y^{(i)}, f^{(i)})\}_{i=1}^{N}$, with $f^{(i)} = h(x^{(i)}, y^{(i)})$:

$$\mathcal{L}(\theta) = \mathbb{E}_{(x,y,f) \sim \mathcal{D}} \left[-\log p_{\theta} \left(f, y \mid x \right) \right] \tag{2.13}$$

and factorizing the joint probability by either predicting the feedback after generating an output (J. Li et al., 2017) or predicting an output for a given specific feedback (Korbak et al., 2023).

2.3.3 Learning Models of Human Feedback

Directly using human feedback to improve model behaviour is not feasible in the general case: asking humans to provide feedback for *every* model output is both expensive and time-consuming. An alternate approach to obtaining human feedback is to develop models that can predict or approximate it. Although they may not be perfect, they can provide feedback at a low cost after training, enabling feedback-dependent techniques at scale.

More formally, given a feedback function $h : \mathcal{X} \times \mathcal{Y}_1 \times \cdots \times \mathcal{Y}_n \to \mathcal{F}$, we want to learn a *parametric* (numerical) feedback model $\hat{h}_{\phi} : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ (with parameters ϕ) that "agrees" with human feedback. This agreement is expressed through a loss which the model is trained to minimize.

$$\phi_{\star} = \underset{\phi}{\operatorname{argmin}} \, \mathbb{E}_{x, y_1, \cdots, y_n \sim \mathcal{D}_f} \left[\mathcal{L}(\phi) \right] \tag{2.14}$$

$$\mathcal{L}(\phi) = \operatorname{loss} \left(\hat{h}_{\phi}(x, y_1), \cdots, h(x, y_{1:n}) \right) \tag{2.15}$$

For example, if the feedback function we are trying to model is also numerical ($h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$), then this loss can just be any standard regression loss, such as the squared difference between the human feedback and model feedback

$$\mathcal{L}(\phi) = \left(\hat{h}_\phi(x, y) - h(x, y)\right)^2 \tag{2.16}$$

Feedback modelling has been studied extensively in the context of *learning evaluation metrics* for NLP. For example, in MT, BLEURT (Sellam et al., 2020) and COMET (Rei et al., 2020a) were trained to regress on human translation quality assessments (using a loss similar to Equation 2.16) and, as mentioned, this has led them to be much more effective than traditional lexical metrics[3]. Notably, these metrics were not trained with the intent of improving generation directly, though they can be used for this purpose (as we will show in Chapter 7). More recently, there has been more interest in developing feedback models directly for improving generation (Böhm et al., 2019; Ziegler et al., 2019).

After training a feedback model, we can use it almost exactly as we would use human feedback, for example, by optimizing the model parameters with feedback it gives. Additionally, using a feedback model also opens the door to leveraging feedback during the decoding process, which can be used to improve the performance of an already trained model (as we will show in Chapter 7). Since feedback models are imperfect proxies of human feedback, generally regularization is added when training with them to avoid *"reward overfitting."* (Gao, Schulman, & Hilton, 2022). Nevertheless, feedback models suffer from distribution shifts (Zouhar et al., 2024) which makes successfully optimizing for them challenging.

2.3.4 AI Feedback

Feedback models have been crucial in advancing generation techniques by effectively leveraging feedback. However, they are heavily reliant on human input: for example, Gao, Schulman, and Hilton (2022) found that across various preference model sizes, utilizing fewer than 1,000 comparisons resulted in only chance improvements. Moreover, employing static feedback can make consistency challenging, causing changes in the model's output distribution.

AI-generated feedback, an emerging research area, focuses on harnessing the LLM's own abilities to enhance the model without human intervention. For example, LLMs can be prompted to assess the harmfulness of their outputs and revise them accordingly (Bai, Kadavath, et al., 2022), or an external LLM can be used to critique the task model's outputs (Akyürek et al., 2023; Madaan et al., 2021). Generally, larger LLMs provide feedback closer to humans and are particularly effective at generating feedback when they themselves have been trained to adhere to align with human preferences.

While AI-generated feedback still hasn't replaced human feedback (which is implicitly

[3] *Reference-based* MT metrics can be seen feedback models which take as an additional input a *reference* translation, which eases the difficulty in approximating translation feedback

used to train the LLM and design the *prompt* that generates feedback), it has the potential to significantly reduce the need for human feedback when training and aligning *new* models, and generality of LLMs allow us to explore a wide range of feedback types and formats, providing a good model to study the impact of these choices on alignment at scale.

Part I

Understanding the Behaviour

Chapter 3

Scaling Laws for Multilingual Neural MT

In this chapter, we provide a large-scale empirical study of the scaling behaviour and properties of multilingual neural machine translation models. We examine how increases in the model size affect the model performance and investigate the role of the training mixture composition on the scaling behavior. We find that changing the weightings of the individual language pairs in the mixtures only affects the multiplicative factor of the scaling law. In particular, we observe that multilingual models trained using different mixing rates all exhibit the same scaling exponent. Through a novel joint scaling law formulation, we compute the *effective number of parameters* allocated to each language pair and examine the role of language similarity in the scaling behavior of our models. We find little evidence that language similarity has any impact. In contrast, the direction of the multilinguality plays a significant role, with models translating from multiple languages *into* English exhibiting a larger number of effective parameters than their reversed counterparts. Finally, we leverage our observations to predict the performance of models trained with *any* language weighting at *any* scale, reducing efforts required for language balancing in large multilingual models. Our findings generalize to out-of-domain test sets and to multiple evaluation metrics, such as ChrF and BLEURT.

This chapter is based on Fernandes, Ghorbani, et al. (2023).

3.1 Introduction

Over the past few years, scaling has emerged as a popular and effective way to improve the performance of neural networks (Brown et al., 2020b; Chowdhery et al., 2022a; Lepikhin et al., 2020). Given the costs associated with training large neural models, much work has gone into understanding their scaling properties and predicting the evolution of their performance with scale through **scaling laws**. Such scaling laws have been instrumental in guiding the model development efforts across a variety of domains such as computer vision (Zhai et al., 2022), language modeling (Hoffmann et al., 2022; Kaplan, McCandlish, Henighan, et al., 2020), and neural machine translation (Ghorbani et al., 2022).

Despite these impressive developments, most of the scaling laws studies available in the literature only focus on single-task, single-language models. On the contrary, current massive neural models are often trained to solve more than one task across one or more modalities & languages (Chowdhery et al., 2022a; Reed et al., 2022; Sanh et al., 2022). This disconnect from the current research frontier limits the applicability of scaling laws in guiding model development decisions. In particular, currently available scaling laws studies are unable to inform the decision process on **balancing the different tasks effectively** at training time. Without such guidance, practitioners often have to rely on cumbersome and costly approaches such as approximate grid search to inform their decision-making; such approaches quickly become infeasible as the problem scale grows.

In this paper, we take the initial step towards developing a quantitative understanding of the scaling behavior for multitask models. We choose multilingual neural machine translation (MNMT) as the setup for this initial study. This choice is motivated by several reasons: (1) MNMT has been framed and studied as a multi-task optimization problem extensively in the past (Arivazhagan et al., 2019b; Dong et al., 2015; Luong et al., 2015; Z. Wang et al., 2021); (2) It provides a popular setup with mature benchmarks and substantial literature on scaling (Bapna et al., 2022; Costa-jussà et al., 2022; Huang et al., 2019; Lepikhin et al., 2020); (3) Moreover, recent results on scaling laws for single-task MT models provide a natural starting point for our study (Bansal et al., 2022; Ghorbani et al., 2022; Gordon et al., 2021; B. Zhang et al., 2022). (4) Finally, recent findings on the optimization dynamics of MNMT models greatly simplify our study by removing the need to examine the role of the optimization algorithm in our results (Xin et al., 2022).

For our analysis, we train over 200 MNMT models (ranging from 20M to 1B non-embedding parameters) and systematically examine their scaling behaviors. We focus our investigation on the **data-rich, compute-rich regime** where we have access to vast amounts of training data for all the language pairs (i.e. tasks)[1] and the models are trained to near convergence. Here, the main bottleneck in the model performance is the lack of model capacity. We establish the following observations:

- For each fixed training mixture, the evolution of the test cross-entropy loss for the i_{th} language pair ($\mathcal{L}_i$) with model size (N) follows a scaling law that resembles the scaling behavior of single-language-pair models:

$$\mathcal{L}_i(N; \boldsymbol{p}) \approx \beta_{\boldsymbol{p},i} N^{-\alpha_{\boldsymbol{p},i}} + L_{\infty}^{(\boldsymbol{p},i)}. \tag{3.1}$$

  Here, $\boldsymbol{p}$ is a vector of probabilities that determines the weight of each language pair in the training mixture. Furthermore, we find that changes in the language pair weightings only affect the multiplicative factor β; the scaling exponent α and the irreducible loss L_∞ are unaffected by these changes. As such, our results suggest that scaling mul-

[1]Using machine translation terminology, all language pairs are *high-resource*.

tilingual models improves the loss at rates that are independent of the weights of the individual language pairs in the training mixture.

- We leverage these findings to propose a scaling law that jointly predicts the performance for all language pairs and weightings considered, and use it to examine how the model splits its capacity in between the language pairs by computing the **effective number of parameters** allocated to each language pair (Section 3.3.3).
- We examine the popular belief that training multilingual models on similar languages is more effective than training models in unrelated languages. Surprisingly, for the high-resource language pairs considered, we do not observe any significant differences in the scaling behavior of models trained to translate from English into related languages (En→{De, Fr}) and models trained in unrelated languages (En→{De, Zh}). In contrast, we observe that models trained to translate from multiple languages into English (XX→En) benefit much more from multitasking compared to those trained on translation out of English (En→XX).
- By approximating the capacity splitting behavior of multilingual models, in Section 3.3.4, we provide a scaling law that predicts **the full task performance trade-off frontier** as a function of the model size N (See Figure 3.7). In Section 3.3.4, we describe how such predictions can be leveraged for efficiently guiding task balancing when training large multilingual models.

3.2 Background

3.2.1 Neural Scaling Laws

Recent research suggests that the performance of large neural models is well-predicted by a smooth function of the fundamental problem parameters: the model size N,[2] the size of the training data D, and the amount of compute used for training C (Hernandez et al., 2021; Hestness et al., 2017; Kaplan, McCandlish, Henighan, et al., 2020; Rosenfeld et al., 2019). The most relevant of these studies to ours is Ghorbani et al. (2022) where the authors study the effects of increasing the model size for single-task NMT models in the data-rich ($D \to \infty$), compute-rich ($C \to \infty$) regime. In this setting, the authors show that the following *bivariate* law describes the scaling behavior of encoder-decoder Transformers

$$\mathcal{L}(N_e, N_d) = \beta N_e^{-\alpha_e} N_d^{-\alpha_d} + L_\infty. \tag{3.2}$$

Here, N_e and N_d correspond to the number of parameters in the encoder and decoder respectively and L_∞ corresponds to the irreducible loss associated with the task. $\{\beta, \alpha_e, \alpha_d, L_\infty\}$ are the parameters of the scaling law that need to be empirically estimated from the data.

[2]Following the literature conventions, we only consider the non-embedding layers when computing N.

In addition, Ghorbani et al. (2022) examine the question of optimally allocating parameters between the encoder and the decoder. They show that in order to attain the optimal scaling behavior, one needs to proportionally scale the encoder and the decoder together. Under such scaling scheme, Equation 3.2 simplifies to

$$\mathcal{L}(N) = \beta N^{-\alpha} + L_{\infty}, \tag{3.3}$$

which is similar to the scaling behavior observed in other domains such as computer vision (Zhai et al., 2022) and autoregressive generative models (Henighan et al., 2020).

Based on these results, to achieve the optimal scaling behavior, we adopt the proportional encoder-decoder scaling scheme for our experiments.

3.2.2 Multitask Optimization

Multilingual NMT is commonly framed and studied as a multitask optimization problem (Arivazhagan et al., 2019b; Dong et al., 2015; Luong et al., 2015; Z. Wang et al., 2021).

We focus our investigation on the supervised learning setup where the model parameters $\boldsymbol{\theta} \in \mathbb{R}^N$ are trained on K different tasks simultaneously. In multilingual MT, each task corresponds to translation for a different language pair. We denote the loss associated with task i with $\mathcal{L}_i(\boldsymbol{\theta})$.

Multitask models are often trained by minimizing a convex combination of the per-task losses:

$$\hat{\boldsymbol{\theta}}(\boldsymbol{w}) = \operatorname{argmin} \sum_{i=1}^{K} \boldsymbol{w}_i \mathcal{L}_i(\boldsymbol{\theta}) \ ; \ \boldsymbol{w} > 0, \ \sum_{i=0}^{K} \boldsymbol{w}_i = 1 \tag{3.4}$$

Here, $\boldsymbol{w}$ is a fixed vector of the task weights, determined apriori by the practitioner to emphasize her preferences on the balancing of the tasks. This so-called **scalarization** approach is highly popular in the community due to its effectiveness and simplicity.[3] In fact, despite this simplicity, recent results on multitask optimization suggest that scalarization achieves performances on par or better than bespoke optimizers designed specifically for multitask models (Kurin et al., 2022; Xin et al., 2022).

In current large text models, such explicit scalarization is rare. Instead, scalarization is often implemented **implicitly**, by sampling observations from each task proportionally to that task's weight. Proportional sampling produces (in expectation) the same overall loss function as explicit scalarization but with much less engineering complexity.

[3] See Boyd and Vandenberghe (2004) for more a detailed discussion of scalarization.

Xin et al. (2022) demonstrate that there exists a smooth, well-defined performance trade-off frontier for multitask models in the data rich regime. This frontier represents the performance trade-off the model is able to achieve in between the tasks as a function of the task weights (see Figure 3.1 for a cartoon representation).

Naturally, finding an accurate characterization of the performance trade-off frontier is key in finding a systematic solution to the task balancing issue.

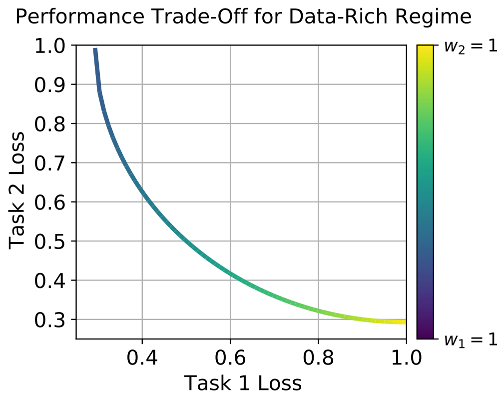

Figure 3.1: Cartoon representation of the performance trade-off (paretto) frontier for a hypothetical model.

3.3 Effects of Scale in Multilingual MT

3.3.1 Experimental Setup

We use the pre-LN encoder-decoder Transformer architecture in our models (Vaswani et al., 2017a; Xiong et al., 2020). We train models of up to 8 sizes, approximately ranging from 20M to 1B (non-embedding) parameters. When scaling encoder-decoder Transformers, to achieve the optimal scaling behavior, we scale the encoder and the decoder proportionally by increasing the model dimension and the number of layers in tandem.

For our experiments, we train two cohorts of models: En→XX and XX→En. For En→XX cohort, we train multilingual model for translation from English to {German (De), Chinese (Zh)} and {German (De), French (Fr)}. For XX→En cohort, we present results for {De, Zh}→En.

We use the *implicit* scalarization approach to train our models; each observation in the training batch is chosen from the first language pair with probability p and the second language pair with probability $1 - p$. For our experiments, we choose p from the set

$$p \in \{0, 0.05, 0.1, 0.3, 0.5, 0.7, 0.9, 0.95, 1\}. \tag{3.5}$$

For En→XX models, to avoid confusing the model, we prepend a language token to the source sentence specifying the target language (e.g. `<2de>`). The models are trained with per-token cross-entropy loss and Adafactor optimizer (Shazeer & Stern, 2018), using a fixed batch size of 500K tokens and inverse square root learning rate schedule. To mirror the compute-rich regime as closely as possible, we trained our models to near convergence. In practice, this translates to training our smaller models ($<$ 500M parameters) for 500K gradient steps and larger models for 1M steps.

To place our models in the data-rich regime, we use a massive in-house web-crawled dataset for training our models. We filter this data using an online data selection procedure (W. Wang et al., 2018) and high-quality web-domain reference sets, extracting 600M sentences for each language pair in the En→XX direction and 1.2B sentences for the XX→En language pairs. We tokenize this corpus by using a pretrained multilingual SentencePiece (Kudo, 2018) vocabulary, with a size of 128K sub-words.

We measure the performance of models on both *in-domain* and *out-of-domain* test sets. For the in-domain test set, we extract 2000 sentences from the same in-house datasets used to create the training (ensuring no overlap). For out-of-domain, we use *newstest2019* (Barrault et al., 2019), consisting of 2000 sentence-pairs extracted from aligned news documents.

3.3.2 Results & Analysis

Understanding Multilingual Scaling We start our analysis by independently examining the model scaling behavior for each individual language pair weighting p in (3.5). For each choice of p, we fit a scaling law of the form

$$\mathcal{L}_i(N;p) = \beta_{p,i} N^{-\alpha_{p,i}} + L_\infty^{(p,i)} \tag{3.6}$$

to the empirical (test) performance of models resulting from that language pair weighting.

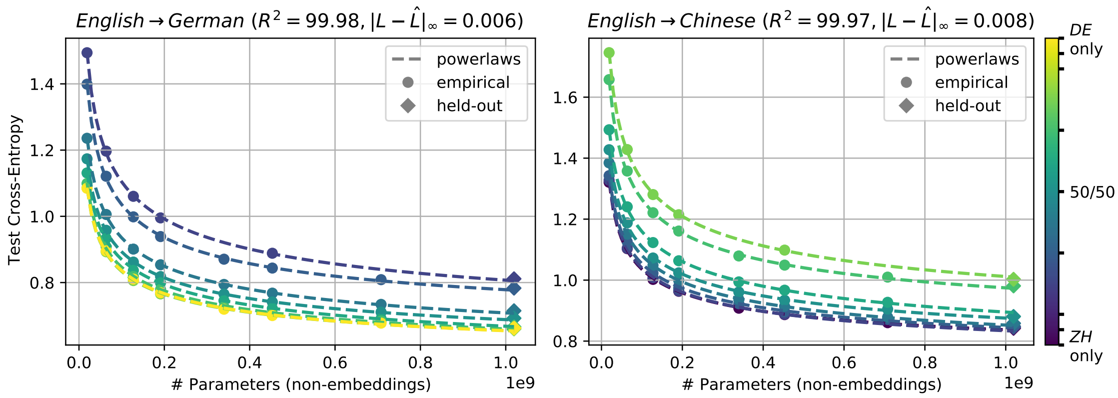

Figure 3.2: The evolution of the in-domain test cross-entropy loss with model size for En→{De, Zh} models, as well as the fitted scaling laws. These scaling laws are **fitted separately for each language pair weighting.** The color represents the weighting of the languages. The scaling laws are able to capture close to 100% of the variation in the data for both language pairs. Note that we don't show the *zero-shot* behavior.

Figure 3.2 presents our findings for En→{De, Zh} models. Each point on the graph corresponds to the empirical test-cross entropy performance of a model at the end of the training.[4] We observe that our per-weighting laws are able to capture the scaling behavior

[4] For low probability language pairs, we apply a convergence correction procedure to make up for slow

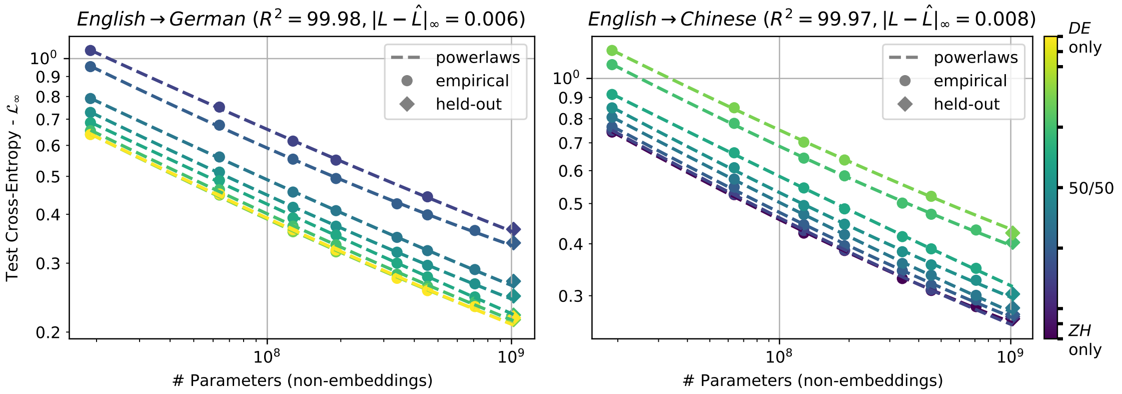

Figure 3.3: Log-log plot of the evolution of the (in-domain) test cross-entropy loss as we scale. We subtract a constant $L_\infty^{(i)}$, jointly fitted for all the weightings (Equation 3.7). All lines are nearly parallel, suggesting that the scaling exponent is unchanged for all p.

of our multilingual models on both language pairs. As expected, when the weight for one of the languages is decreased, the performance of the models on that language decreases for all scales. Our results suggest that the benefits of the increased model size for MNMT models are well-described by a power-law.

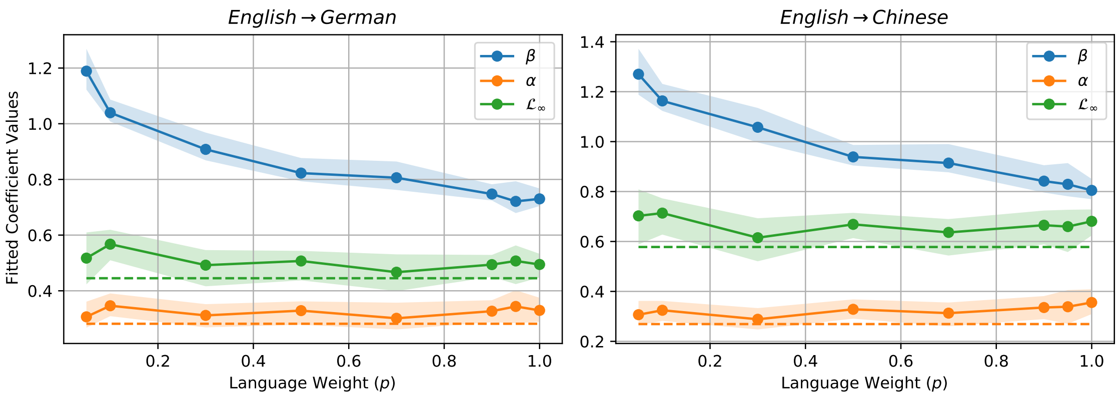

Figure 3.4: Coefficient values for German (left) and Chinese (right) as a function of the language weight, with the shaded region representing the standard deviation. The dashed lines represent the value of jointly fitted coefficients from Equation 3.7.

Figure 3.4 shows the fitted scaling law coefficients for different values of p. The shaded area marks the one standard deviation uncertainty interval of our estimates.[5] Interestingly, we find that, across all values of p, both the scaling exponent (α) and the irreducible loss ($\mathcal{L}_\infty$) seem to be relatively unchanged. In particular, all of our estimated α and $\mathcal{L}_\infty$ parameters are within two standard deviations of each other. In contrast, the multiplicative

convergence.

[5]We gauge the uncertainty in the coefficients by measuring the fluctuations in our estimates when our empirical datapoints are perturbed by $\epsilon \overset{\text{i.i.d}}{\sim} \mathcal{N}(0, \sigma^2)$. We choose a conservative σ of 1% of the observed empirical loss for each data point.

factor β seems to be highly sensitive to the choice of p.

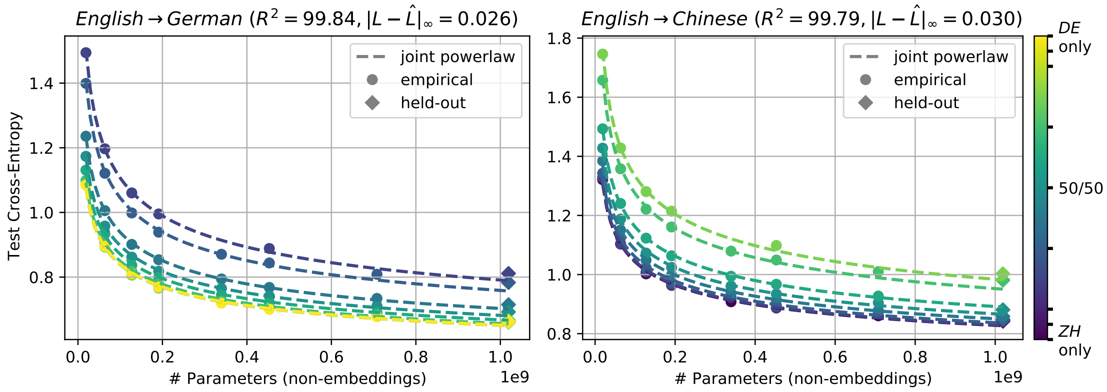

Figure 3.5: The **joint** scaling law of Equation 3.7 closely captures the scaling behavior of En→{De, Zh} models. Test loss here is evaluated on in-domain test sets.

Figure 3.3 visually confirms the assertion that for our models α_p and L_∞ are effectively constant. Here, we have subtracted a fixed constant $L_\infty^{(i)}$ from all the Figure 3.4 curves corresponding to the language pair i. We then plot results on log-log axes. As the figure suggests, the lines are all near parallel, suggesting that the scaling exponent is unchanged for all p. In practical terms this means that, for example, doubling the capacity of a multilingual model will reduce its loss by the same $\frac{1}{2^\alpha}$ factor, no matter how the training mixture looks like. This also means that single-language-pair scaling laws can be used to gauge the benefits of scaling multilingual models.

Jointly Modeling Multilingual Scaling Based on the findings above, we make the assumption that the scaling exponents and the irreducible losses are independent of the language pair weights, and propose a **joint** scaling law of the form

$$\mathcal{L}_i(N; p) \approx \beta_{p,i} N^{-\alpha_i} + L_\infty^{(i)}. \tag{3.7}$$

Figure 3.5 shows the fit of this joint scaling law for En→{De, Zh} models evaluated on the in-domain test sets. Note that here, we fit a total of 10 parameters for each language pair – 8 for $\beta_{p,i}$'s and two for α_i and $L_\infty^{(i)}$. In contrast, in Figure 3.2, we used 24 overall parameters to capture the scaling behavior for each language pair. Despite this significant decrease in the number of total fitted parameters, we observe that our joint laws are able to almost completely capture the scaling behavior.

3.3.3 Effective Network Capacity for Multilingual Models

We leverage our joint scaling law to examine how MNMT models split their capacity in between the different language pairs. We start by defining the notion of **the effective number of parameters**:

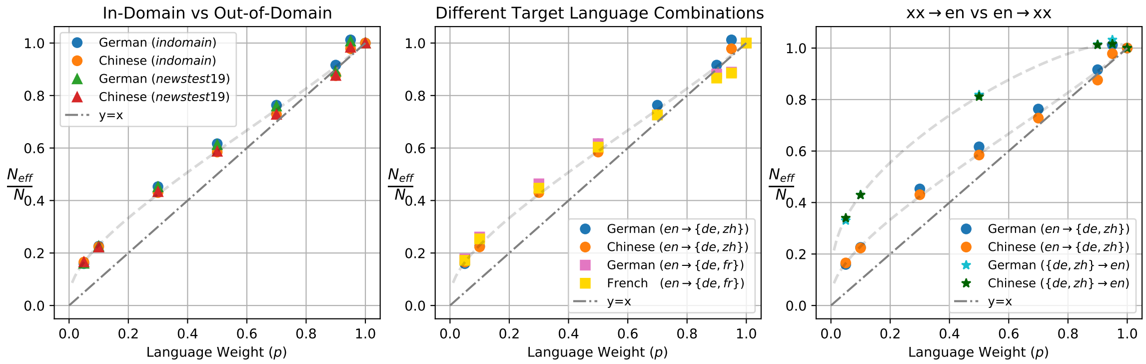

Figure 3.6: The effective fraction of parameters allocated to each language pair as estimated by our joint scaling laws. Gray dashed lines correspond to the fitted $\hat{f}_i$ described in Equation 3.12. *Left:* Comparison of the capacity splitting behavior of En→{De, Zh} models for in-domain and out-of-domain test sets. We observe minimal differences between the two. *Center:* Comparison of the capacity splitting behavior for En→{De, Zh} and En→{De, Fr} models. We don't observe any changes in the interaction between the language pairs based on language similarity. *Right:* Comparison of the capacity splitting behavior for translation to and from English. XX→En models exhibit more synergy among the language pairs.

Definition. Consider a multilingual model in which a language pair i has been trained with weight p. We define the effective number of parameters allocated to i, $N_{\text{eff}}^{(i,p)}$, to be equal to the number of parameters necessary for a single-language-pair model solely trained on i to reach the same (test loss) performance as the multilingual model.

Mathematically, $N_{\text{eff}}^{(i,p)}$ can be written as the solution of the equation

$$\mathcal{L}_i(N;p) = \mathcal{L}_i(N_{\text{eff}}^{(i,p)};1). \tag{3.8}$$

A simple derivation yields that

$$N_{\text{eff}}^{(i,p)} = \left(\frac{\beta_{1,i}}{\beta_{p,i}}\right)^{\frac{1}{\alpha_i}} N. \tag{3.9}$$

Crucially, our calculations suggest that the fraction of parameters allocated to language pair i, which we denote by $f_i(p)$, is independent of the model size:

$$f_i(p) \equiv N_{\text{eff}}^{(i,p)}/N = \left(\frac{\beta_{1,i}}{\beta_{p,i}}\right)^{\frac{1}{\alpha_i}}. \tag{3.10}$$

This observation yields a fundamental, scale-independent quantity that can be leveraged for understanding the interactions between the different language pairs in the model.

Figure 3.6 shows the empirically estimated effective parameter ratios for our models. Several observations are in order:

Consistency Across Domains: In Figure 3.6 (left), we compare the capacity splitting behavior of the models on in-domain and out-of-domain (newstest19) test sets. Even though the scaling laws coefficients for in-domain and out-of-domain test sets differ, we observe that the capacity splitting behavior is mostly unchanged with different test sets. These findings hint at some measure of universality across test domains on how MNMT models divide their capacity and share their parameters.

Consistency Across Languages Pairs: In Figure 3.6 (center), we compare the capacity splitting behavior of En→{De, Zh} and En→{De, Fr} models. The conventional wisdom in the MT literature suggests that the tasks in En→{De, Fr} should exhibit a more positive interaction with each other compared to En→{De, Zh}. This is often justified by the intuition that representations are more aligned in related languages and more aligned representations will encourage parameter sharing (Dabre et al., 2017). Surprisingly, our results suggest that the interaction dynamics in En→{De, Fr} and En→{De, Zh} models are not significantly different. In both settings, we observe a relatively neutral multilingual behavior – the performance of an MNMT model of size N trained on language pair i with (sampling) weight p is essentially similar to a single-language-pair model of size pN. In other words, there is minimal synergy among the languages in both setups. This finding, while surprising in light of prior evidence for cross-lingual transfer and the effects of shared scripts or subword vocabularies (Blum et al., 2025; Qi et al., 2023; Schäfer et al., 2024), likely reflects the specific nature of *out-of-English* translation, and language/task similarity may play a role capacity sharing profile in other multi-task settings (as hinted below).

En→XX vs XX→En: In Figure 3.6 (right), we compare the interaction between the language pairs when translating out of English vs when translating into English. In stark contrast to the En→XX setting, when translating into English, we observe significant positive synergy among the language pairs. This observation aligns well with recent results in the literature showing multilingual models achieving SOTA performance for translation to English (Chowdhery et al., 2022a; Lepikhin et al., 2020). It is unclear if this synergy arises as a specificity of having English as the target language or because multilingual encoding is intrinsically more amenable to parameter sharing than multilingual decoding. Understanding the exact dynamics giving rise to such positive interaction between the language pairs is an exciting open question.

Benefits for Massive Multilingual Models: While we observed minimal synergy between En→XX languages pairs, and therefor minimal gains in *absolute* effective capacity, if we look at *relative* effective capacity, we can see considerable benefits in using multilingual models for language pairs with small weight. For example, a model trained for En→{De, Zh} with 5% weight on German has an effective capacity of more than 3× a model trained with 5% capacity of this model for only German. These relative gains are even more evident when there is positive task synergy, such as for XX→En, where models train with 5% weight have more than 6× gain in (effective) parameters. This hints that, if these findings generalize beyond the two-task setup , then training large multilingual models for training mixtures with a large number of small weight language pairs is significantly more memory efficient than training separate small models for each language pair.

3.3.4 Guiding Language Balancing

As discussed earlier, one of the areas where multilingual scaling laws can be most impactful is in guiding language balancing/weighting when training large multilingual models, an open problem that has been studied extensively (Arivazhagan et al., 2019a; X. Wang et al., 2020). However, in its current form, our (joint) scaling law can only be use to decide between weightings that were used for fitting it and cannot be used to predict performance on new, unseen weightings, as $\beta_{p,i}$ needs to be estimated empirically.

To extend to unseen language pair weightings, we instead focus on estimating $f_i(\cdot)$. Given access to $f_i(p)$, accurate prediction of $\mathcal{L}_i(N)$ for **any weighting** can be achieved by using the **single-language-pair scaling law**:

$$\mathcal{L}_i(N; p) = \beta_{1,i}\big(\hat{f}_i(p)N\big)^{-\alpha_i} + L_\infty^{(i)}. \tag{3.11}$$

As observed in Section 3.3.3, $f_i(p)$ has a number of desirable properties that makes it easy to estimate: (i) it is invariant to test set and languages, (ii) it is smooth and generally well-behaved. As such, one can achieve an accurate approximation of f with just a few data points.

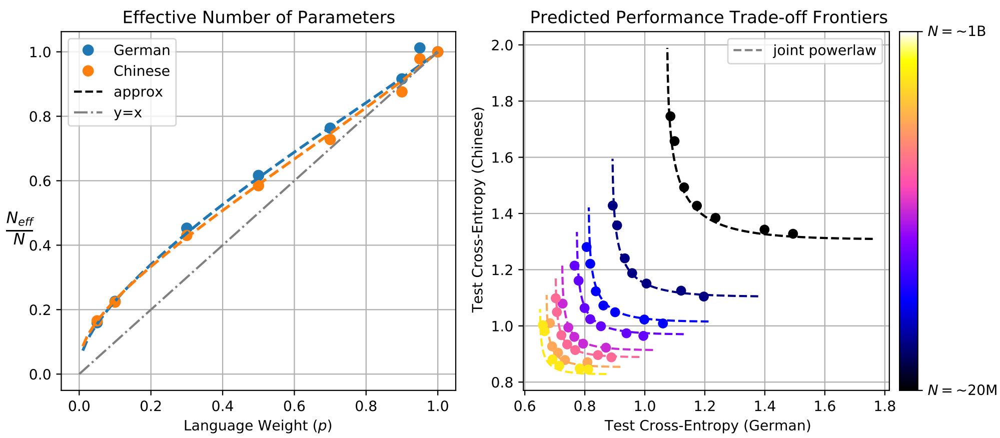

Figure 3.7: Approximate joint scaling laws described by equations (3.11) and (3.12) almost perfectly capture the language pair interactions across all scales. *Left:* The fitted approximation $\hat{f}$ described in Equation 3.12. *Right:* The predicted performance trade-off frontier (dashed lines) as well as the empirically observed trade-off values.

We utilize this methodology to estimate the full task performance trade-off frontier for En→{De, Zh} models. For estimating $f_i(\cdot)$, we fit an approximate joint scaling law of the form Equation 3.11, where $\hat{f}_i(\cdot)$ is parameterized as

$$\hat{f}_i(p) = p + c_1 p^{c_2}(1-p)^{c_3} \tag{3.12}$$

with c_1, c_2, c_3 being fitted coefficients. Figure 3.7 demonstrates our results; our procedure is able to almost perfectly capture the full task performance frontier across a variety of model scales. With access to such accurate predictions of the performance frontier, a practitioner can precisely determine how to weigh the individual language pairs during training based on her preferences and target model size.

We should note that the choice of function class to fit $f_i(\cdot)$ is highly dependent on the practitioner's computational budget. In our case, we prioritized accuracy and used a flexible function class of the form (3.12) for fitting. Such flexibility comes with the cost of needing to compute more empirical values to reliably estimate $f(\cdot)$. In the scenarios with more limited computational budget, we have observed that even rudimentary linear approximations of f are able to provide accurate representations of the performance frontier.

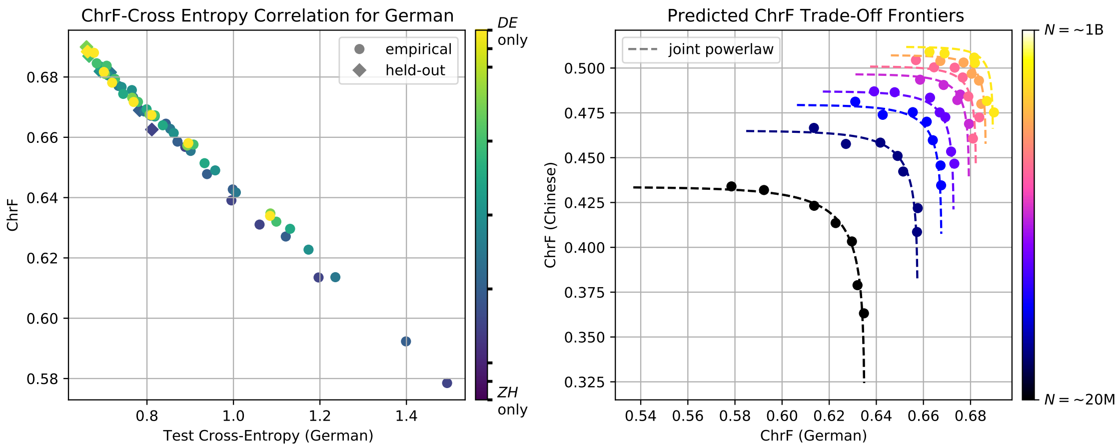

Figure 3.8: The generation quality behavior of our models as measured by ChrF. *Left:* We observe consistent positive correlations between ChrF and cross-entropy loss. *Right:* Our scaling laws can be used to generate accurate performance trade-off frontiers for ChrF.

Translation Quality Finally, we note that in the MT literature, quality is often measured via metrics such as BLEU (Papineni et al., 2002c), ChrF (Popović, 2015) and BLEURT (Sellam et al., 2020) as opposed to cross-entropy, since the latter doesn't account for the problem of *decoding* translations from the models and is sometimes found to not correlate with human preferences (Koehn & Knowles, 2017). As such, MT practitioners might be concerned regarding the applicability of these results for practical applications. To ensure that our findings also apply to the quality of translations, we decode translations from our trained models using beam search (Graves, 2012a) and evaluate how their quality changes as we scale the models, using ChrF and BLEURT.

Figure 3.8 (left) shows cross-entropy and ChrF scores for the En→De language pair of our En→{De, Fr} models, evaluated on the in-domain test set. We find that this automatic metric has an almost-linear relationship with cross-entropy, hinting that our observations also generalize from cross-entropy to generation quality. Figure 3.8 (right) also shows the predicted ChrF performance trade-off frontier obtained by fitting our joint scaling law (Equation 3.7) to the ChrF performance on the in-domain test set (parametrizing the effective parameter fraction function as in Equation 3.12). Our procedure is able to capture this trade-off frontier almost as well as the cross-entropy frontier.

3.4 Conclusions & Subsequent Work

Current state-of-the-art large neural models are moving towards using as much data from as many domains, modalities and languages as possible to unlock exciting new capabilities. Unfortunately, a clear understanding of the behavior of such multitask models at scale is missing. In this chapter, we attempted to take an initial step towards alleviating this problem by performing a large-scale study of the properties of multilingual machine translation models (which can be seen as doing multiple translation tasks). We showed that cross-lingual scaling interactions are surprisingly simple, with the per-task scaling exponent for model capacity only dependent on the properties of the individual translation tasks. In the end, we sketched a procedure to compute the *effective fraction of parameters* assigned to a language, and to estimate the task performance trade-off frontier for all model scales.

Our work had some limitations that hindered the generalizability of our findings: we focused only scaling with respect to *model capacity*, studying translation tasks on the *high*-resource scenario; and, to keep our investigation tractable, we focused mostly on two-task scenario. Since the publication of our work, others have studied the more realistic case where there is limited data (L. Chen et al., 2023) (incorporating, and for the more general case of language modeling with more than two tasks / languages (He et al., 2024), obtaining similar findings on the invariance of scaling with respect to cross-lingual and cross-task interactions.

Chapter 4

Measuring and Increasing Context Usage in Context-Aware MT

Studying model performance and its scaling behaviour can already be an effective tool to make informed choices about data composition and task distribution. But we still don't understand how models leverage information in the input, even when trained for a single task. For example, in *document-level* machine translation, it is still unclear how models use context outside of individual sentences, or even if its beneficial *at all*. Likelihood and other performance metrics do not provide this more fine-grained understanding.

In this chapter, we introduce a new metric, *conditional cross-mutual information*, to quantify the usage of context by these models. Using this metric, we measure how much document-level machine translation systems use particular varieties of context. We find that target context is referenced more than source context, and that conditioning on a longer context has a diminishing effect on results. We then introduce a new, simple training method, *context-aware word dropout*, to increase the usage of context by context-aware models. Experiments show that our method increases context usage and that this reflects on the translation quality according to metrics such as BLEU and COMET, as well as performance on anaphoric pronoun resolution and lexical cohesion contrastive datasets.

This chapter is based on Fernandes et al. (2021).

4.1 Introduction

While neural machine translation (NMT) is reported to have achieved human parity in some domains and language pairs (Hassan et al., 2018), these claims seem overly optimistic and no longer hold with document-level evaluation (Läubli et al., 2018; Toral et al., 2018). Recent work on *context-aware* NMT attempts to alleviate this discrepancy by incorporating the surrounding context sentences (in either or both the source and target sides) in the translation system. This can be done by, for example, feeding context sentences to standard NMT models (Tiedemann & Scherrer, 2017), using different encoders for context (J. Zhang et al., 2018), having cache-based memories (Tu et al., 2018a), or using models with hierarchical attention mechanisms (Maruf, Martins, & Haffari, 2019; Miculicich et al., 2018) — more details in §4.2. While such works report gains in translation quality compared to sentence-level baselines trained on small datasets, recent work has shown that, in more realistic high-resourced scenarios, these systems fail to outperform simpler baselines with respect to overall translation accuracy, pronoun translation, or lexical cohesion (Lopes et al., 2020).

Given the findings, we next consider how to encourage models to use more context. Specifically, we introduce a simple but effective variation of word dropout (Sennrich et al., 2016a) for context-aware machine translation, dubbed **CoWord dropout** (§4.4). Put simply, we randomly drop words from the *current* source sentence by replacing them with a placeholder token. Intuitively, this encourages the model to use extra-sentential information to compensate for the missing information in the current source sentence. We show that models trained with CoWord dropout not only increase context usage compared to models trained without it but also improve the quality of translation, both according to standard evaluation metrics (BLEU and COMET) and according to contrastive evaluation based on inter-sentential discourse phenomena such as anaphoric pronoun resolution and lexical cohesion (§7.4, Table 4.1).

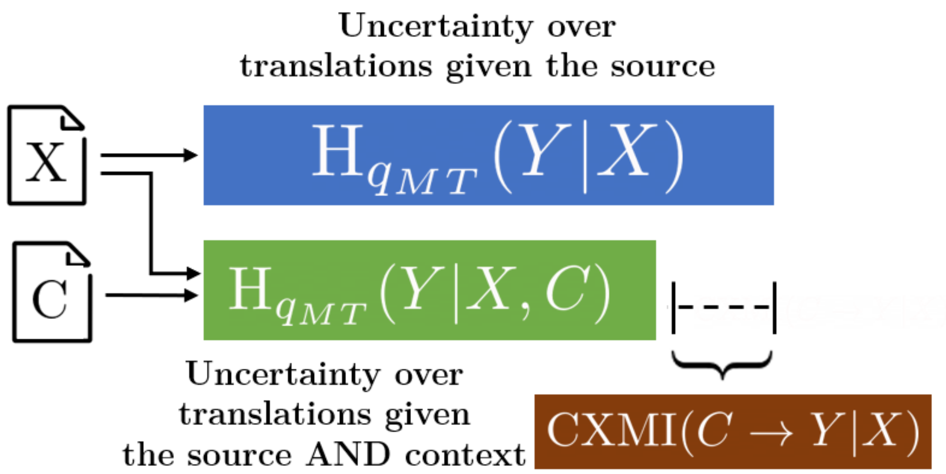

Figure 4.1: Illustration of how we can measure context usage by a model q_{MT} as the amount of information gained when a model is given the context C and source X vs when the model is only given the X.

Source:	*The Church is merciful...* It always welcomes the misguided lamb.
Target:	*Die Kirche ist barmherzig...*
Baseline	Es heisst die fehlgeleiteten Schäflein immer willkommen.
Context-Aware	Es heisst die fehlgeleiteten Schäflein immer willkommen.
+CoWord	Sie heisst die fehlgeleiteten Schäflein immer willkommen.

Table 4.1: Example where context (italic) is needed to correctly translate the pronoun "it". Both the sentence-level baseline and context-aware model fail to correctly translate it while the context-aware model trained with *CoWord dropout* correctly captures the context.

4.2 Context-Aware Neural Machine Translation

We are interested in learning a system that translates documents consisting of multiple sentences between two languages.[1] More formally, given a corpus of parallel documents in two languages, $\mathcal{D} = \{D_1, ..., D_N\}$, where each document is a sequence of source and target sentences, $D = \{(x^{(1)}, y^{(1)}), ..., (x^{(K)}, y^{(K)})\}$, we are interested in learning the mapping between the two languages.

We consider the typical (auto-regressive) neural machine translation system q_θ parameterized by θ. The probability of translating $x^{(i)}$ into $y^{(i)}$ given the context of the sentence $C^{(i)}$ is

$$q_\theta(y^{(i)}|x^{(i)}, C^{(i)}) = \prod_{t=1}^{T} q_\theta(y_t^{(i)}|x^{(i)}, y_{<t}^{(i)}, C^{(i)})$$

where $y_t^{(i)}$ represents the tth token of sentence $y^{(i)}$. This context can take various forms. On one end, we have the case where no context is passed, $C^{(i)} = \varnothing$, and the problem is reduced to sentence-level translation. On the other end, we have the case where all the source sentences and all the previous generated target sentences are passed as context $C^{(i)} = \{x^{(1)}, ..., x^{(K)}, y^{(1)}, ..., y^{(i-1)}\}$.

As mentioned, there are many architectural approaches to leveraging context (see §4.5 for a more complete review), and the methods that we present in this paper are compatible with most architectures because they do not specify how the model q_θ uses the context. In experiments, we focus mostly on the simpler approach of concatenating the context to the current sentences (Tiedemann & Scherrer, 2017). Recent work by Lopes et al. (2020) has shown that, given enough data (either through pre-training or larger contextual datasets), this simple approach tends to be competitive with or even outperform its more complex counterparts

[1] Here, a "document" could be an actual document but it could also represent other contextual collections of text, such as a sequence of dialogue utterances.

4.3 Measuring Context Usage

4.3.1 Conditional Cross-Mutual Information

While context-aware models *allow* use of context, they *do not ensure* contextual information is actually used: models could just be relying on the current source sentence and/or previously generated target words from the same sentence when generating the output.

Contrastive evaluation, where models are assessed based on the ability to distinguish correct translations from contrastive ones, is a common way to assess the ability of context-aware models to capture specific discourse phenomena that require inter-sentential context, such as anaphora resolution (Müller et al., 2018) and lexical cohesion (Bawden et al., 2018). However, these methods only provide an indirect measure of context usage with respect to a limited number of phenomena and can fail to capture other, unknown ways in which the model might be using context. Y. Kim et al. (2019) showed that most improvements to translation quality are due to non-interpretable usages of context, such as the introduction of noise that acts as a regularizer to the encoder/decoder. This problem is further exacerbated by the fact that there is no clear definition of what entails "context usage".

In a different context, Bugliarello et al. (2020) introduced *cross-mutual information* (XMI), to measure the "difficulty" of translating between different language pairs in sentence-level neural machine translation. Given a language model q_{LM} for a target sentence Y and a translation model q_{MT} for translating from X to Y, XMI is defined as:

$$\mathrm{XMI}(X \rightarrow Y) = \mathrm{H}_{q_{LM}}(Y) - \mathrm{H}_{q_{MT}}(Y|X),$$

where $\mathrm{H}_{q_{LM}}$ denotes the cross-entropy of the target sentence Y under the language model q_{LM} and $\mathrm{H}_{q_{MT}}$ the conditional cross-entropy of Y given X under the translation model q_{MT}. This allows us to measure how much information the source sentence gives us about the target sentence (an analogue of mutual information for cross-entropy). In the case where q_{LM} and q_{MT} perfectly model the underlying probabilities we would have $\mathrm{XMI}(X \rightarrow Y) = \mathrm{MI}(X, Y)$, the true mutual information.

Taking inspiration from the above, we propose **Conditional Cross-Mutual Information (CXMI)**, a new measure of the influence of context on a model's predictions. This is done by considering an additional variable for the context C and measuring how much information the context C provides about the target Y given the source X. This can then be formulated as

$$\mathrm{CXMI}(C \rightarrow Y|X) = \\ \mathrm{H}_{q_{MT_A}}(Y|X) - \mathrm{H}_{q_{MT_C}}(Y|X, C)$$

where $\mathrm{H}_{q_{MT_A}}$ is the entropy of a *context-agnostic* machine translation model, and $\mathrm{H}_{q_{MT_C}}$

refers to a *context-aware* machine translation model. This quantity can be estimated over an held-out test set with N sentence pairs and the respective context as:

$$\mathrm{CXMI}(C \rightarrow Y|X) \approx \\ -\frac{1}{N}\sum_{i=1}^{N}\log\frac{q_{MT_A}(y^{(i)}|x^{(i)})}{q_{MT_C}(y^{(i)}|x^{(i)},C^{(i)})}$$

While q_{MT_A} and q_{MT_C} can, in theory, be any models, we are interested in removing any confounding factors other than the context that might lead to instability in the estimates of the distributions. For example, if q_{MT_A} and q_{MT_C} use completely different models, it would not be clear if the difference in the probability estimates is due to the introduction of context or due to other extraneous factors such as differences in architectures, training regimens, or random seeds. To address this we consider a single model, q_{MT}, that is able to translate with and without context (more on how this achieved in §4.3.2). We can then set the context-agnostic model and the contextual model to be the same model $q_{MT_A} = q_{MT_C} = q_{MT}$. This way we attribute the information gain to the introduction of context. Throughout the rest of this work, when we reference "context usage" we will precisely mean this information gain (or loss).

4.3.2 Experiments

Data We experiment with a document-level translation task by training models on the IWSLT2017 (Cettolo et al., 2012) dataset for language pairs EN → DE and EN → FR (with approximately 200K sentences for both pairs). We use the test sets 2011-2014 as validation sets and the 2015 as test sets. To address the concerns pointed out by Lopes et al. (2020) that gains in performance are due to the use of small training corpora and weak baselines, we use Paracrawl (Esplà et al., 2019) and perform some data cleaning based on language identification tools, creating a pretraining dataset of around 82M and 104M sentence pairs for EN → DE and EN → FR respectively.

All data is encoded/vectorized with byte-pair encoding (Sennrich et al., 2016b) using the *SentencePiece* framework (Kudo & Richardson, 2018). For the non-pretrained case, we use 20K vocabulary size shared across source/target, while for the pretrained case we use a 32K vocabulary size.

Besides translation quality, we also evaluate our models on two contrastive datasets for different discourse phenomena to better assess the ability of our models to capture context (more on this in §4.4.2):

- For the EN → DE language pair, we evaluate on the *ContraPro* dataset (Müller et al., 2018), targeting anaphoric pronoun resolution. Source-side sentences contain the English anaphoric pronoun *it* while target-side sentences contain the corresponding German translations *er*, *sie* or *es*. Contrastive erroneous translations are automatically created by replacing the correct pronoun with one of the other two. The test set contains 4,000 examples for each target pronoun type and context is needed to correctly disambiguate. Context includes the four previous sentences

- For the EN $\rightarrow$ FR language pair, we evaluate on the dataset by Bawden et al. (2018) targeting anaphoric pronoun resolution and lexical cohesion. It contains 200 manually curated examples for each phenomenon. Anaphora examples include singular and plural personal and possessive pronouns that require context to be correctly inferred and the dataset is balanced such that a model that does not use context can only achieve 50% accuracy. Context includes the previous sentence

Models and Optimization For all our experiments, we consider an encoder-decoder Transformer architecture (Vaswani et al., 2017a). In particular, we train the *transformer small* (hidden size of 512, feedforward size of 1024, 6 layers, 8 attention heads). For the pretrained setup, we also pre-train a *transformer large* architecture (hidden size of 1024, feedforward size of 4096, 6 layers, 16 attention heads) and subsequently fine-tune on the IWSL2017 datasets.

As in Vaswani et al. (2017a), we train using the Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.98$ and use an inverse square root learning rate scheduler, with an initial value of 10^{-4} and 5×10^{-4} for pretrained and non-pretrained cases respectively, and with a linear warm-up in the first 4000 steps. We train the models with early stopping on the validation perplexity.

We train all our models on top of the *Fairseq* framework (Ott et al., 2019).

What Context Matters? To assess the relative importance of different context sizes on both the source and target side, we start by considering two models, one for the source-side context and one for the target-side context, that receive context of size k, $C^{(i)} = \{x^{(i-k)}, \ldots, x^{(i-1)}\}$ or $C^{(i)} = \{y^{(i-k)}, \ldots, y^{(i-1)}\}$. During training, k is selected randomly to be in $\{1, \ldots, 4\}$ for every example. This way the model is trained to translate the same source without and with different context sizes and is thus able to translate based on any context size in that interval.

Figure 4.2 shows the CXMI values computed over the test set as a function of the context size for both the source-side and target-side contextual models for both the non-pretrained and pretrained regimens for the EN $\rightarrow$ DE language pair.

For the non-pretrained case, for both the source and target context, the biggest jump in context usage is when we increase the context size from 0 to 1. After that, increasing the context size leads to diminishing increases in context usage and even *reduced* context usage for the source-side context. Interestingly, when the model is stronger, such as in the pretrained case, we can see that it can leverage target-side context even better than the non-pretrained case, with a similar trend of diminishing increases in context usage for both

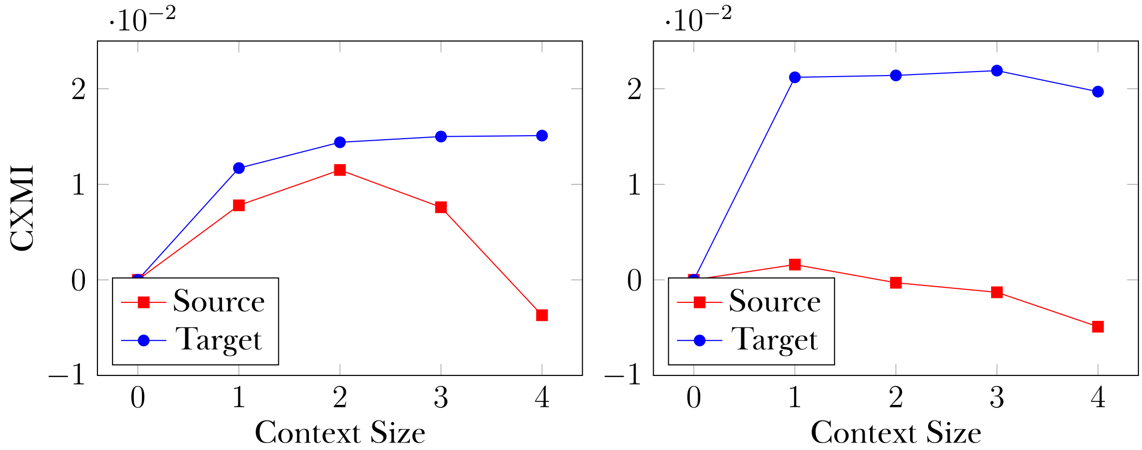

Figure 4.2: CXMI values for the EN $\rightarrow$ DE as a function of source and target context sizes for non-pretrained (left) and pretrained (right) models.

regimes. However, this is not the case for the source-side context, and it seems that the pretrained model is barely able to use the contextual information on this side.

Overall, for this regime, we can conclude that having a context size of one or two previous sentences on both sides is beneficial to the model, and that target-side context is *slightly* more used than source-side context. This appears to corroborate the findings of Bawden et al. (2018) that target-side context is more effective than the source context.

Does CXMI Really Measure Context Usage? To assert that CXMI correlates with interpretable measures of context usage, we perform a correlation analysis with the performance in the contrastive datasets mentioned. In these datasets, usage of context is evident where the model picks the right answer when it is passed the context and is not able to do so when no context is given. Thus Table 4.2 shows the point-biserial correlation coefficient[2] between the *per-sample* CXMI and binary random variable and a binary variable that takes the value 1 if the contextual model picks the correct translation and the non-contextual model picks the incorrect one, for different context sizes on the pretrained model. We can see that there is a statistically significant correlation between both values, which strengthens the notion that CXMI captures previous measures of context usage to some extent.

4.4 Increasing Context Usage

4.4.1 Context-aware Word Dropout

Motivated by the above results demonstrating the limited context usage of models trained using the standard MLE training paradigm, particularly with respect to more distant con-

[2]The Point-Biserial correlation coefficient is a special case of the *Pearson* correlation coefficient when one of the random variables is dichotomous.

	r_{pb}		
Context Size	(1)	(2)	(3)
1	**0.365**	**0.315**	**0.206**
2	**0.366**	-	-
3	**0.367**	-	-
4	**0.366**	-	-

Table 4.2: Point-Biserial correlation coefficients with pretrained models for different context sizes. Measured on *ContraPro* (1) and Bawden et al. (2018), both for pronoun resolution (2) and lexical cohesion (3). Bold values mean statistically significant correlation with $p < 0.01$.

text, we now ask the question: "Is it possible to modify the training methodology to increase context usage by the model?" As an answer, we extend a popular regularization technique used in sentence-level machine translation, word dropout (Sennrich et al., 2016a), to the context-aware setting. The idea behind context-aware word (CoWord) dropout is to model the translation probability between $x^{(i)}$ and $y^{(i)}$ as

$$p_\theta(y^{(i)}|x^{(i)}) = \prod_{t=1}^{T} p_\theta(y_t^{(i)}|\tilde{x}^{(i)}, y_{<t}^{(i)}, C^{(i)}),$$

where $\tilde{x}^{(i)}$ is a perturbed version of the current source sentence generated by randomly dropping tokens and replacing them with a mask token given a dropout probability p:

$$r_t^{(i)} \sim \text{Bernoulli}(p)$$
$$\tilde{x}_t^{(i)} = \begin{cases} \langle \text{MASK} \rangle & \text{if } r_t^{(i)} = 1 \\ x_t^{(i)} & \text{otherwise.} \end{cases}$$

In the case where no context is passed $C^{(i)} = \varnothing$, CoWord dropout reduces to word dropout. The intuition behind such a perturbation is that, by dropping information from the current source and *not* the context, we increase the relative reliability of context $C^{(i)}$, therefore providing the inductive bias that context is important for the translation. We will see in §7.4 that this inductive bias is beneficial and that CoWord dropout not only improves performance but also increases context usage.

4.4.2 Experiments

To assess if our proposed CoWord dropout increases context usage by models, we train a model using the same *dynamic* context size setting used in §4.3.2.

Setup As in §4.3.2, we consider *transformer* models trained on the IWSLT2017 for both EN → DE and EN → FR, both from scratch and pretrained. Due to findings in the previous section, we consider models with either only *target-side* context or both *source-side* and *target-side* context.

Context Usage Figure 4.3 plots the CXMI values on the test set as a function of the *target* context size as we increase the dropout value p. We see that increasing this value consistently increases context usage according to CXMI across different context sizes. Note that, at test time, CoWord dropout is disabled, which means that it provides inductive bias only during training and models *learn* to use more context by themselves.

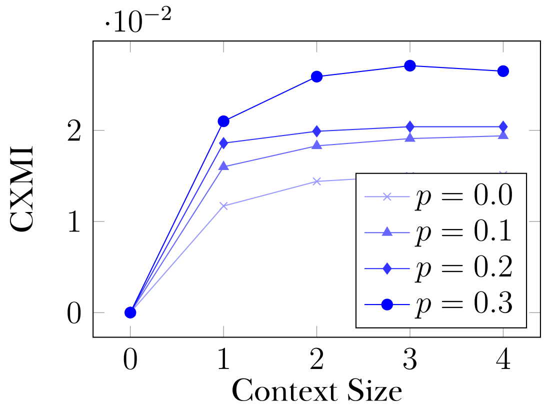

Figure 4.3: CXMI as a function of target context size for varying CoWord dropout

Source Context	Source	Target Context	Target	ΔCXMI
More people watched games because it was faster.	It was more entertaining	Mehr Menschen sahen zu, die Spiele wurden schneller	und unterhaltsamer.	0.53
The ball comes off track.	You don't know where it's going to land	Der Ball ist außer Kontrolle	Sie wissen nicht, wo er landet.	0.33
I really think that this lie that we've been sold about disability is the greatest injustice	It makes life hard for us	Meiner Meinung nach ist diese Luge über Behinderung eine schreiende Ungerechtigkeit	Sie macht uns das Leben schwer.	0.25

Table 4.3: Examples where models with CoWord dropout use the target context more than models trained without it. Word highlighted blue in the context are used to disambiguate translations while highlighted green in the target use context according to native speakers. Words underlined in the target are the ones with the highest *per-word* CXMI i.e. the ones that use the most context according to the model

Table 4.3 illustrates some examples where the CoWord dropout increased the *per-sample* CXMI significantly. While the model only has access to *target* context, we present the source context for clarity. In the first example, while the source is a complete sentence, the target is only a fragment of one so the context helps complete it. In the other two examples shown, we can see that context helps disambiguate the gender of the German translation of the English pronoun *it*. Interestingly, the words that use context the most according to CXMI match very closely to the ones that native speakers annotated.

Translation Quality To evaluate if the increased usage of context correlates with better machine translation quality, based on the previous experiments on context usage and values for CoWord dropout, we consider three models trained with *fixed-size* context:

- A **baseline** that has no context, reducing to sentence-level model ie: *i.e.*, $C^{(i)} = \varnothing$;
- a **one-to-two** model having as context the previous target sentence, *i.e.*, $C^{(i)} = \{y^{(i-1)}\}$;
- a **two-to-two** model having as context the previous source sentence and the previous target sentence, *i.e.*, $C^{(i)} = \{x^{(i-1)}, y^{(i-1)}\}$.

For all models with target context, when decoding, we use the previous decoded sentences as target context.

		EN → DE				EN → FR			
				w/ pretraining				w/ pretraining	
	p	BLEU	COMET	BLEU	COMET	BLEU	COMET	BLEU	COMET
baseline	0.0	26.36	0.083	35.10	0.521	37.62	0.450	42.98	0.679
	0.1	**27.26**	0.159	**35.15**	**0.525**	38.16	0.472	**43.28**	**0.679**
	0.2	26.97	**0.163**	35.13	0.524	**38.34**	**0.474**	42.99	0.678
1-to-2	0.0	26.60	0.087	**35.22**	**0.528**	37.59	0.450	42.89	0.672
	0.1	**27.36**	0.174	34.92	0.527	38.25	0.472	42.88	0.677
	0.2	27.33	**0.193**	34.75	0.524	**38.27**	**0.485**	**42.90**	**0.678**
2-to-2	0.0	26.85	0.090	34.47	0.471	37.54	0.453	**42.97**	0.674
	0.1	**27.72**	0.169	34.51	0.522	**38.30**	0.467	42.95	**0.676**
	0.2	27.21	**0.177**	**34.65**	**0.525**	38.15	**0.468**	42.88	0.675

Table 4.4: Results on IWSLT2017 with different probabilities for CoWord dropout. Averaged across three runs for each method.

Table 4.4 shows the performance across three different seeds of the baseline and contextual models for both the non-pretrained and pretrained setting, with increasing values of CoWord dropout p. We also run the baseline with CoWord dropout (which, as said previously, reduces to word dropout) to ensure that improvements were not only due to regularization effects on the *current* source/target. We report the standard BLEU score (Papineni et al., 2002b) calculated using sacreBLEU (Post, 2018) and COMET, a more accurate evaluation method using multilingual embeddings (Rei et al., 2020a).

For the non-pretrained case, we can see that a CoWord dropout value $p > 0$ consistently improves the performance of the contextual models when compared to models running with $p = 0$ and with the sentence-level baseline with the same values for word dropout. For the pretrained case, the improvements are not as noticeable, although models trained with CoWord dropout still always outperform models trained without it. This is perhaps a reflection of the general trend that better models are harder to improve.

Table 4.5 shows that CoWord dropout is also helpful for the *multi-encoder* model, with CoWord dropout helping significantly. This shows that this method could be helpful for context-aware architectures other than concatenation-based.

	p	EN → DE BLEU	EN → DE COMET	EN → FR BLEU	EN → FR COMET
baseline	0.0	26.36	0.083	37.62	0.450
	0.1	**27.26**	0.159	38.16	0.472
	0.2	26.97	**0.163**	**38.34**	**0.474**
multi	0.0	26.64	0.104	37.85	0.466
	0.1	**27.45**	0.190	37.98	0.460
	0.2	27.31	**0.190**	**38.30**	**0.484**

Table 4.5: Results on IWSLT2017 for a multi-encoder 1-to-2 model with different probabilities for CoWord dropout. Averaged across three runs for each method.

	p	EN → DE Pronouns	EN → DE w/ pretraining Pronouns	EN → FR Pronouns	EN → FR Cohesion	EN → FR w/ pretraining Pronouns	EN → FR w/ pretraining Cohesion
baseline	0.0	42.96	46.57	50.00	50.00	50.00	50.00
1-to-2	0.0	57.36	76.79	68.16	49.99	**86.83**	**56.83**
	0.1	58.70	76.28	72.33	51.49	86.49	56.66
	0.2	**60.72**	**77.52**	**72.99**	**52.16**	85.66	56.49
2-to-2	0.0	61.06	80.33	72.16	50.99	85.66	64.33
	0.1	**66.00**	**80.35**	**73.99**	**52.49**	87.16	**65.99**
	0.2	65.47	79.97	**73.99**	**52.49**	**88.49**	63.99

Table 4.6: Results on anaphoric pronoun resolution and lexical cohesion contrastive datasets with different probabilities for CoWord dropout. Averaged across three runs for each method.

Discourse Phenomena While automatic metrics such as BLEU and COMET allow us to measure translation quality, they mostly target *sentence-level* quality and do not specifically focus on phenomena that require context-awareness. Contrastive datasets, as described in §4.3.2, allow us to measure the performance of context-aware models in specific discourse phenomena by comparing the probability of *correct* translation against the *contrastive* translations. Models that capture the targeted discourse phenomena well will consistently rank the correct translation higher than the contrastive ones. While there is a disconnect between the translation (done via decoding) and contrastive evaluation, it is currently the best way to measure a model's performance on context-aware discourse phenomena.

Table 4.6 shows the average performance over the contrastive datasets of the baseline and contextual models for both the (non-)pretrained settings, with increasing values of CoWord dropout p. We can see that in general, increasing CoWord dropout leads to improved performance, particularly for the non-pretrained case. This gain is particularly clear for pronoun resolution and the EN → DE language pair. We hypothesise that this is due to the small size of the contrastive sets for the EN → FR language pair, which leads to high variance.

Table 4.7 similarly shows that CoWord dropout improves the performance of the multi-encoder model across all phenomena, which again shows that our proposed regularization method has benefits for multiple architectures for context-aware machine translation. Curiously, when these models are trained without CoWord dropout, they achieve performance similar to the sentence-level baseline, while when dropout is applied, they are able to effectively start using context.

		EN → DE	EN → FR	
	p	Pronouns	Pronouns	Cohesion
baseline	0.0	42.96	50.00	50.00
multi	0.0	42.85	49.74	49.99
	0.1	47.29	51.74	50.24
	0.2	**47.57**	**52.50**	**50.99**

Table 4.7: Results on anaphoric pronoun resolution and lexical cohesion contrastive datasets for the multi-encoder 1-to-2 model with different probabilities for CoWord dropout. Averaged across three runs for each method.

4.5 Related Work

Context-aware Machine Translation There have been many works in the literature that try to incorporate context into NMT systems. Tiedemann and Scherrer (2017) first proposed the simple approach of concatenating the previous sentences in both the source and target side to the input to the system; Jean et al. (2017), Bawden et al. (2018), and J. Zhang et al. (2018) used an additional context-specific encoder to extract contextual features from the previous sentences; Maruf and Haffari (2018) and Tu et al. (2018b) used cache-based memories to encode context; L. Wang et al. (2017) used a hierarchical RNN to encode the global context from all previous sentences; Miculicich et al. (2018) and Maruf, Martins, and Haffari (2019) used hierarchical attention networks to encode context; J. Chen et al. (2020) added document-level discourse structure information to the input; Z. Sun et al. (2020) trained a simple concatenation-based model with *varying* context size during training to have a model that is able to translate with any context size, similar to what we did. Similarly to what we do with CoWord dropout, Jean and Cho (2019) attempted to maximise sensitivity to context by introducing a margin-based regularization term to explicitly encourage context usage.

For a more detailed overview, Maruf, Saleh, and Haffari (2019) extensively describe the different approaches and how they leverage context. While these models lead to improvements with small training sets, Lopes et al. (2020) showed that the improvements are negligible when compared with the concatenation baseline when using larger datasets. However, importantly, both our metric CXMI for measuring context usage and the proposed regularization method of CoWord dropout, can theoretically be applied to any of the above-mentioned methods.

Evaluation In terms of evaluation, most previous work focuses on targeting a system's performance on contrastive datasets for specific inter-sentential discourse phenomena. Müller et al. (2018) built a large-scale dataset for anaphoric pronoun resolution, Bawden et al. (2018) manually created a dataset for both pronoun resolution and lexical choice and Voita et al. (2019) created a dataset that targets deixis, ellipsis and lexical cohesion. Stojanovski et al. (2020) showed through adversarial attacks that models that do well on other contrastive datasets rely on surface heuristics and create a contrastive dataset to address this. In contrast, our CXMI metric is phenomenon-agnostic and can be measured with respect to *all* phenomena that require context in translation.

Information-Theoretic Analysis Bugliarello et al. (2020) first proposed cross-mutual information (XMI) in the context of measuring the difficulty of translating between languages. Our work differs in that we propose a *conditional* version of XMI, where S is always observed, and we use it to assess the information gain of context rather than the difficulty of translating different languages.

4.6 Conclusion and Subsequent Work

We introduce a new, architecture-agnostic, metric to measure how context-aware machine translation models are using context and propose a simple regularization technique to increase context usage by these models. Our results are theoretically applicable to almost all recently proposed context-aware models.

Subsequent work has shown that CXMI can be used to measure context usage in language modelling more generally (O'Connor & Andreas, 2021), and even for understanding cross-modal interactions in multimodal models (Frank et al., 2021). We have also since shown how we can use a *pointwise* version our metric to identify how much context these models use for each *individual* translation (Fernandes, Yin, et al., 2023), bridging the gap to the *local* explanations we will discuss in the following chapters.

Part II

Explaining as Teaching

Chapter 5

Optimizing Model Explanations for Teaching

Modern, large-scale neural network models introduce intepretability problems not only at a *global behavioural* level, but also in providing *local explanations* for individual predictions. However the precise goal of such explanations, and how to demonstrate they achieve such goal, remains an open problem. Some research argues that explanations should help *teach* a student (either human or machine) to simulate the model being explained, and that the quality of explanations can be measured by the simulation accuracy of students on unexplained examples.

In this chapter, leveraging meta-learning techniques, we extend this idea to *improve the quality of the explanations themselves*, specifically by optimizing explanations such that student models more effectively learn to simulate the original model. We train models on three natural language processing and computer vision tasks, and find that students trained with explanations extracted with our framework are able to simulate the teacher significantly more effectively than ones produced with previous methods. Through human annotations and a user study, we further find that these learned explanations more closely align with how humans would explain the required decisions in these tasks.

This chapter is based on Fernandes, Treviso, et al. (2022).

5.1 Introduction

While deep learning's performance has led it to become the dominant paradigm in machine learning, its relative opaqueness has brought great interest in methods to improve *model interpretability*. Many recent works propose methods for extracting *explanations* from neural networks (section 5.6), which vary from the highlighting of relevant input features (Arras et al., 2017; Ding et al., 2019; Simonyan et al., 2014) to more complex representations of the reasoning of the network (Mu & Andreas, 2020; T. Wu et al., 2021).

However, are these methods actually achieving their goal of making models more interpretable? Some concerning findings have cast doubt on this proposition; different explanations methods have been found to disagree on the same model/input (Bastings et al., 2021; Neely et al., 2021) and explanations do not necessarily help predict a model's output and/or its failures (Chandrasekaran et al., 2018).

In fact, the research community is still in the process of understanding *what* explanations are supposed to achieve, and *how* to assess success of an explanation method (Doshi-Velez & Kim, 2017; Miller, 2019). Many early works on model interpretability designed their methods around a set of desiderata (Lertvittayakumjorn & Toni, 2019; Sundararajan et al., 2017b) and relied on qualitative assessment of a handful of samples with respect to these desiderata; a process that is highly subjective and is hard to reproduce. In contrast, recent works have focused on more quantitative criteria: correlation between explainability methods for measuring *consistency* (Jain & Wallace, 2019; Serrano & Smith, 2019), *sufficiency* and *comprehensiveness* (DeYoung et al., 2020), and *simulability*: whether a human or machine consumer of explanations understands the model behavior well enough to predict its output on unseen examples (Doshi-Velez & Kim, 2017; Lipton, 2016a). Simulability, in particular, has a number of desirable properties, such as being intuitively aligned with the goal of *communicating* the underlying model behavior to humans and being measurable in manual and automated experiments (Hase & Bansal, 2020; Pruthi, Dhingra, et al., 2020; Treviso & Martins, 2020).

For instance, Pruthi, Dhingra, et al. (2020) proposed a framework for automatic evaluation of simulability that, given a *teacher model* and explanations of this model's predictions, trains a *student model* to match the teacher's predictions. The explanations are then evaluated with respect to how well they help a student *learn to simulate* the teacher (section 8.2). This is analogous to the concept in pedagogy of **instructional scaffolding** (Van de Pol et al., 2010), a process through which a teacher adds support for students to aid learning. More effective scaffolding—in our case, better explanations—is assumed to lead to better student learning. However, while this previous work provides an attractive way to *evaluate* existing explanation methods, it stops short of proposing a method to actually *improve* them.

In this chapter, we propose to *learn to explain* by directly learning explanations that provide better scaffolding of the student's learning, a framework we term ***S**caffold-**Ma**ximizing **T**raining* (**SMaT**). Figure 5.1 illustrates the framework: the explainer is used to *scaffold* the student training, and is updated based on how well the student does at *test* time at simulating the teacher model. We take insights from research on meta-learning (Finn et al., 2017; Raghu et al., 2021), formalizing our setting as a bi-level optimization problem and optimizing it based on higher-order differentiation (section 5.3). Importantly, our high-level framework makes few assumptions about the model we are trying to explain, the structure of the explanations or the modalities considered. To test our framework, we then introduce a *parameterized* attention-based explainer optimizable with SMaT that works for any model with attention mechanisms (section 5.4).

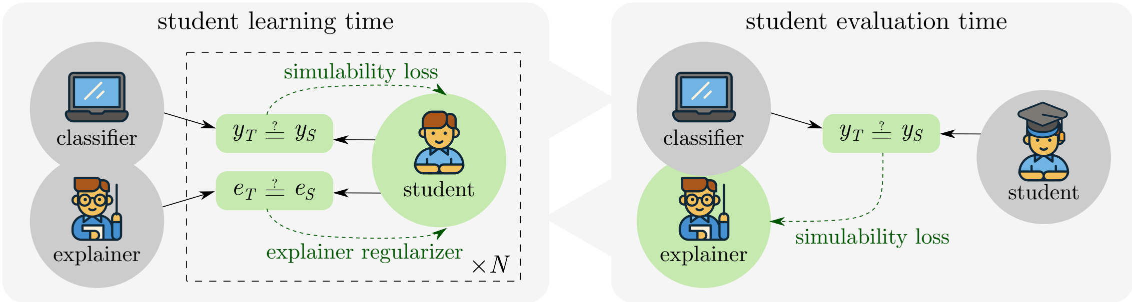

Figure 5.1: Illustration of our SMaT framework. First, a student model is trained to recover the classifier's predictions and to match the explanations given by the explainer. Then, the explainer is updated based on how well the trained student *simulates* the classifier (without access to explanations). In practice, we repeat these two consecutive processes for several steps. Green arrows and boxes represent learnable components.

We experiment with SMaT in text classification, image classification, and (multilingual) text-based regression tasks using pretrained transformer models (section 5.5). We find that our framework is able to effectively optimize explainers across all the considered tasks, where students trained with *learned* attention explanations achieve better simulability than baselines trained with *static* attention or gradient-based explanations. We further evaluate the *plausability* of our explanations (i.e., whether produced explanations align with how people would justify a similar choice) using human-labeled explanations (text classification and text regression) and through a human study (image classification) and find that explanations learned with SMaT are more plausible than the static explainers considered. Overall, the results reinforce the utility of scaffolding as a criterion for evaluating and improving model explanations.

5.2 Evaluating Explanations with Simulability

Consider a model $T : \mathcal{X} \to \mathcal{Y}$ trained on some dataset $\mathcal{D}_{\text{train}} = \{(x_i, y_i)\}_{i=1}^N$. For example, this could be a text or image classifier that was trained on a particular downstream task (with $\mathcal{D}_{\text{train}}$ being the training data for that task). *Post-hoc* interpretability methods typically introduce an *explainer* module $E_T : \mathcal{T} \times \mathcal{X} \to \mathcal{E}$ that takes a model and an input, and produces an explanation $e \in \mathcal{E}$ for the output of the model given that input, where $\mathcal{E}$ denotes the space of possible explanations. For instance, interpretability methods using saliency maps define $\mathcal{E}$ as the space of *normalized* distributions of importance over L input elements $e \in \triangle_{L-1}$ (where $\triangle_{L-1}$ is the $(L-1)$-probability simplex).

Pruthi, Dhingra, et al. (2020) proposed an automatic framework for evaluating explainers that trains a *student* model $S_\theta : \mathcal{X} \to \mathcal{Y}$ with parameters θ to *simulate* the *teacher* (i.e., the original classifier) in a *constrained* setting. For example, the student can be constrained to have less capacity than the teacher by using a simpler model or trained with a subset of the dataset used for the teacher ($\hat{\mathcal{D}}_{\text{train}} \subsetneq \mathcal{D}_{\text{train}}$).

In this framework, a baseline student S_θ is trained according to

$$\theta^* = \operatorname*{argmin}_\theta \mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_{\text{train}}} \left[\mathcal{L}_{\text{sim}}(S_\theta(x), T(x))\right]$$

and its simulability $\textsc{sim}(S_{\theta^*}, T)$ is measured on an unseen test set. The actual form of $\mathcal{L}_{\text{sim}}$ and $\textsc{sim}(S_{\theta^*}, T)$ is task-specific. For example, in a classification task, we use cross-entropy as the simulation loss $\mathcal{L}_{\text{sim}}$ over the teacher's predictions, while the simulability of a model S_{θ^*} can be defined as the simulation accuracy, i.e., what percentage of the student and teacher predictions match over a *held-out* (validation) set $\mathcal{D}_{\text{test}}$:

$$\textsc{sim}(S_{\theta^*}, T) = \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{val}}}[\mathbb{1}\{S_{\theta^*}(x) = T(x)\}]. \tag{5.1}$$

Next, the training of the student is augmented with explanations produced by the explainer E. We introduce a student explainer $E_S : \mathcal{S} \times \mathcal{X} \rightarrow \mathcal{E}$, (the S-explainer) to extract explanations from the student, and *regularizing* these explanations on the explanations of teacher (the T-explainer), using a loss $\mathcal{L}_{\text{expl}}$ that takes explanations for both models:

$$\theta_E^* = \operatorname*{argmin}_\theta \mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_{\text{train}}} \left[\underbrace{\mathcal{L}_{\text{sim}}\left(S_\theta(x), T(x)\right)}_{\text{simulability loss}} + \beta \underbrace{\mathcal{L}_{\text{expl}}\left(E_S(S_\theta, x), E_T(T, x)\right)}_{\text{explainer regularizer}}\right]. \tag{5.2}$$

For example, Pruthi, Dhingra, et al. (2020) considered as a teacher explainer E_T various methods such as LIME (Ribeiro et al., 2016), Integrated Gradients (Sundararajan et al., 2017b), and attention mechanisms, and explored both attention regularization (using Kullback-Leibler divergence) and multi-task learning to regularize the student.

The key assumption surrounding this evaluation framework is that a student trained with *good* explanations should learn to simulate the teacher better than a student trained with bad or no explanations, that is, $\textsc{sim}\left(S_{\theta_E^*}, T\right) > \textsc{sim}\left(S_{\theta^*}, T\right)$. For clarity, we will refer to the simulability of a model $S_{\theta_E^*}$ trained using explanations as *scaffolded* simulability.

5.3 Optimizing Explainers for Teaching

As a **first contribution** of this chapter, we extend the previously described framework to make it possible to directly optimize the teacher explainer so that it can most effectively teach the student the original model's behavior. To this end, consider a *parameterized* T-explainer E_{ϕ_T} with parameters ϕ_T, and equivalently a *parameterized* S-explainer E_{ϕ_S} with parameters ϕ_S. We can write the loss function for the student and S-explainer as:

$$\mathcal{L}_{\text{student}}(S_\theta, E_{\phi_S}, T, E_{\phi_T}, x) = \mathcal{L}_{\text{sim}}\left(S_\theta(x), T(x)\right) + \beta\mathcal{L}_{\text{expl}}\left(E_{\phi_S}(S_\theta, x), E_{\phi_T}(T, x)\right). \tag{5.3}$$

While this framework is flexible enough to rigorously and automatically evaluate many types of explanations, calculating scaffolded simulability requires an optimization procedure to learn the student and S-explainer parameters θ, ϕ_S. This makes it non-trivial to achieve our goal of directly finding the teacher explainer parameters ϕ_T that optimize scaffolded simulability. To overcome this challenge, we draw inspiration from the extensive literature on meta-learning (Finn et al., 2017; Schmidhuber, 1987), and frame the optimization as the following bi-level optimization problem (see Grefenstette et al. (2019) for a primer):

$$\theta^*(\phi_T), \phi_S^*(\phi_T) = \underset{\theta, \phi_S}{\operatorname{argmin}} \mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_{\text{train}}} [\mathcal{L}_{\text{student}}(S_\theta, E_{\phi_S}, T, E_{\phi_T}, x)] \tag{5.4}$$

$$\phi_T^* = \underset{\phi_T}{\operatorname{argmin}} \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{val}}} \left[\mathcal{L}_{\text{sim}} \left(S_{\theta^*(\phi_T)}(x), T(x)\right)\right]. \tag{5.5}$$

Here, the *inner* optimization updates the student and the S-explainer parameters (Equation 5.4), and in the *outer* optimization we update the T-explainer parameters (Equation 5.5). **Importantly**, our framework does not modify the teacher, as our goal is to explain a model without changing its original behavior. Notice that we also simplify the problem by considering the more tractable simulation loss $\mathcal{L}_{\text{sim}}$ instead of the simulability metric $\textsc{sim}(S_{\theta^*}, T)$ as part of the objective for the outer optimization.

Now, if we assume the explainers E_{ϕ_T} and E_{ϕ_S} are differentiable, we can use gradient-based optimization (Finn et al., 2017) to optimize both the student (with its explainer) and the T-explainer. In particular, we use *explicit* differentiation to solve this optimization problem. To compute gradients for ϕ_T, we have to differentiate through a gradient operation, which requires Hessian-vector products, an operation supported by most modern deep learning frameworks (Bradbury et al., 2018; Grefenstette et al., 2019). However, explicitly computing gradients for ϕ_T through a large number of inner optimization steps is computationally intractable. To circumvent this problem, typically the inner optimization is run for only a couple of steps or a *truncated* gradient is computed (Shaban et al., 2019). In this chapter, we take the approach of taking a *single* inner optimization step and learning the student and S-explainer jointly with the T-explainer *without* resetting the student (Dery et al., 2021). At each step, we update the student and S-explainer parameters as follows:

$$\theta^{t+1} = \theta^t - \eta_{\text{INN}} \nabla_\theta \mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_{\text{train}}} \left[\mathcal{L}_{\text{student}}(S_{\theta^t}, E_{\phi_S^t}, T, E_{\phi_T^t}, x)\right] \tag{5.6}$$

$$\phi_S^{t+1} = \phi_S^t - \eta_{\text{INN}} \nabla_{\phi_S} \mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_{\text{train}}} \left[\mathcal{L}_{\text{student}}(S_{\theta^t}, E_{\phi_S^t}, T, E_{\phi_T^t}, x)\right]. \tag{5.7}$$

After updating the student, we take an extra gradient step with the new parameters but only use these updates to calculate the *outer*-gradient for ϕ_T, without actually updating θ. This approach is similar to the *pilot update* proposed by Zhou et al. (2021), and we verified that it led to more stable optimization in practice:

$$\theta(\phi_T^t) = \theta^{t+1} - \eta_{\text{INN}} \nabla_\theta \mathbb{E}_{(x,y)\sim\hat{\mathcal{D}}_{\text{train}}} \left[\mathcal{L}_{\text{student}}(S_{\theta^{t+1}}, E_{\phi_S^{t+1}}, T, E_{\phi_T^t}, x)\right] \tag{5.8}$$

$$\phi_T^{t+1} = \phi_T^t - \eta_{\text{OUT}} \nabla_{\phi_T} \mathbb{E}_{(x,y)\sim\mathcal{D}_{\text{test}}} \left[\mathcal{L}_{\text{sim}}\left(S_{\theta(\phi_T^t)}(x), T(x)\right)\right]. \tag{5.9}$$

5.4 Parameterized Attention Explainer

As a **second contribution** of this chapter, we introduce a novel *parameterized* attention-based explainer that can be learned with our framework. Transformer models (Vaswani et al., 2017b) are currently the most successful deep-learning architecture across a variety of tasks (Shoeybi et al., 2019; Wortsman et al., 2022). Underpinning their success is the *multi-head attention mechanism*, which computes a *normalized* distribution over the $1 \leq i \leq L$ input elements in parallel for each head h:

$$A^h = \text{SOFTMAX}(Q^h(K^h)^\top), \tag{5.10}$$

where $Q^h = [q_0^h, \cdots, q_L^h]$ and $K^h = [k_0^h, \cdots, k_L^h]$ are the *query* and *key* linear projections over the input element representations for head h. Attention mechanisms have been used extensively for producing saliency maps (Vashishth et al., 2019; Wiegreffe & Pinter, 2019) and while some concerns have been raised regarding their faithfulness (Jain & Wallace, 2019), overall attention-based explainers have been found to lead to relatively good explanations in terms of *plausibility* and *simulability* (Kobayashi et al., 2020; Pruthi, Dhingra, et al., 2020; Treviso & Martins, 2020).

However, to extract explanations from multi-head attention, we have two important design choices:

1. **Single distribution selection:** Since self-attention produces an attention matrix $A^h \in \triangle_{L-1}^L$, we need to *pool* these attention distributions to produce a single saliency map $e \in \triangle_{L-1}$. Typically, the distribution from a single token (such as `[CLS]`) or the *average* of the attention distributions from all tokens $1 \leq i \leq L$ are used.
2. **Head selection:** We also need to *pool* the distributions produced by each head. Typical ad-hoc strategies include using the mean over all heads for a certain layer (Fomicheva, Specia, & Aletras, 2021) or selecting a single head based on plausibility on validation set (Treviso et al., 2021). However, since transformers can have hundreds or even thousands of heads, these choices rely on human intuition or require large amounts of plausibility labels.

In this chapter, we approach the latter design choice in a more principled manner. Concretely, we associate each head with a weight and then perform a weighted sum over all heads. These weights are learned such that the resulting explanation maximizes simulability, as described in section 5.3. More formally, given a model T_{θ_T} and its query and

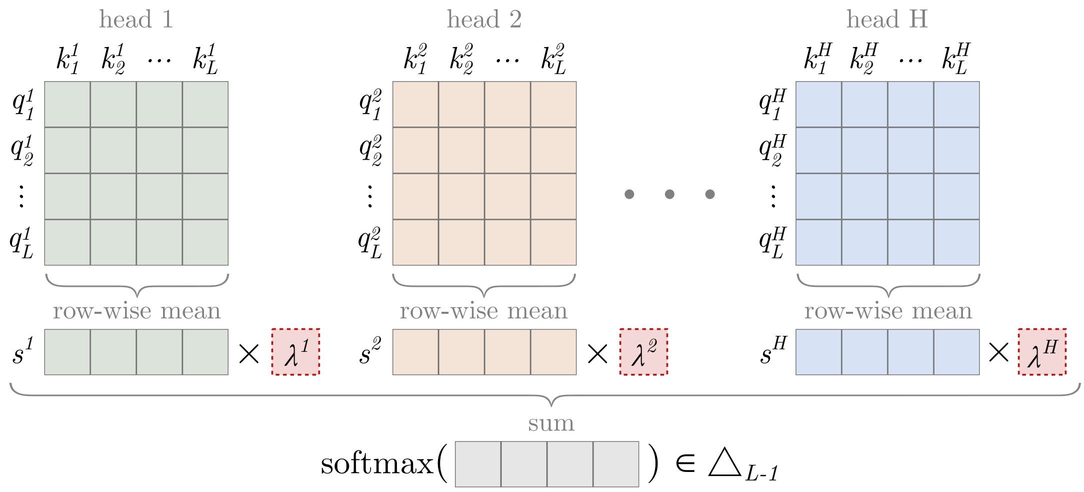

Figure 5.2: Our parameterized attention-based explainer. Dashed red boxes represent learned parameters $\lambda_T = \text{SPARSEMAX}(\phi_T) \in \triangle_{H-1}$, weighting average attention logits of each head $1 \leq h \leq H$. A softmax over the weighted sum generates the attention probabilities.

key projections for an input x for each layer and head $h \leq H$, we define a *parameterized, differentiable* attention explainer $E_{\phi_T}(T_{\theta_T}, x)$ as

$$s^h = \frac{1}{L}\sum_{i=1}^{L}(q_i^h)^\top K^h, \qquad E_{\phi_T}(T, x) = \textsc{softmax}\left(\sum_{h=1}^{H}\lambda_T^h s^h\right), \tag{5.11}$$

where the teacher's head coefficients $\lambda_T \in \triangle_{H-1}$ are $\lambda_T = \textsc{normalize}(\phi_T)$ with $\phi_T \in \mathbb{R}^H$.

In this formulation, $s^h \in \mathbb{R}^L$ represents the average *unnormalized attention logits* over all input elements, which are then combined according to λ_T and normalized with SOFTMAX to produce a distribution in $\triangle_{L-1}$. We apply a normalization function NORMALIZE to head coefficients involved to create a *convex* combination over all heads in all layers. In this work we consider the sparse projection function NORMALIZE = SPARSEMAX (A. Martins & Astudillo, 2016), as:

$$\text{SPARSEMAX}(z) = \underset{p \in \triangle_{H-1}}{\text{argmin}} \|p - z\|_2.$$

We choose SPARSEMAX due to its benefits in terms of interpretability, since it leads to many heads having zero weight. We also found it outperformed every other projection we tried. Figure 5.2 illustrates each step of our parameterized attention explainer.

5.5 Experiments

To evaluate our framework, we attempt to learn explainers for transformer models trained on three different tasks: text classification (subsection 5.5.1), image classification (subsection 5.5.2), and machine translation quality estimation (a text-based regression task, detailed in subsection 5.5.3). We use JAX (Bradbury et al., 2018) to implement the higher-order differentiation, and use pretrained transformer models from the Huggingface Transformers library (Wolf et al., 2020), together with Flax (Heek et al., 2020). For each task, we train a teacher model with AdamW (Loshchilov & Hutter, 2019) but, as explained in section 5.3, we use SGD for the student model (inner loop). We also use scalar mixing (Peters et al., 2018) to pool representations from different layers automatically.[1] We train students with a teacher explainer in three settings:

- **No Explainer**: No explanations are provided, and no explanation regularization is used for training the student (i.e. $\beta = 0$ in Equation 5.3). We refer to studentsin this setting as **baseline** students.
- **Static Explainer**: Explanations for the teacher model are extracted with five commonly-used saliency-based explainers: (1) L2 norm of gradients; (2) a *gradient* $\times$ *input* explainer (Denil et al., 2014); (3) an *integrated gradients* explainer (Sundararajan et al., 2017b); and *attention* explainers that uses the *mean* pooling over attention from (4) all heads in the model and (5) from the heads of the last layer (Fomicheva, Specia, & Aletras, 2021; Vafa et al., 2021).
- **Learned Explainer (SMaT)**: Explanations are extracted with the explainer described in section 5.4, with coefficients for each head that are trained with **SMaT** jointly with the student. We initialize the coefficients such that the model is initialized to be the same as the *static* attention explainer (i.e., performing the mean over all heads).

Independently of the T-explainer, we always use a learned attention-based explainer as the S-explainer, considering all heads except when the T-explainer is a static attention explainer that only considers the last layers' heads, where we do the same for the S-explainer. We use the Kullback-Leibler divergence as $\mathcal{L}_{\text{expl}}$, and we set $\beta = 5$ for attention-based explainers and $\beta = 0.2$ for gradient-based explainers (since we found smaller values to be better). We set $\mathcal{L}_{\text{sim}}$ as the cross-entropy loss for classification tasks, and as the mean squared error loss for text regression. For each setting, we train five students with different seeds. Since there is some variance in students' performance (we hypothesize due to the small training sets) we report the **median** and **interquantile range (IQR)** around it (relative to the 25-75 percentile).

5.5.1 Text Classification

For text classification, we consider the IMDB dataset (Maas et al., 2011), a binary sentiment classification task over highly polarized English movie reviews. As the base pretrained model, we use the small ELECTRA model (Clark et al., 2020), with 12 layers and 4 heads in each (total 48 heads).

[1]While scalar mixing reduced variance of student performance, SMaT also worked with other common pooling methods.

	500	1,000	2,000
No Explainer	81.72 [81.24:81.75]	83.44 [83.36:83.63]	84.84 [84.80:84.88]
Gradient L2	81.66 [81.32:82.00]	82.98 [82.72:83.08]	84.78 [84.96:85.08]
Gradient × Input	84.83 [84.79:84.88]	81.15 [80.95:81.36]	83.84 [83.59:84.99]
Integrated Gradients	82.99 [82.59:82.99]	81.79 [81.72:81.87]	84.20 [84.03:85.03]
Attention (*all layers*)	83.00 [82.60:83.00]	85.72 [85.72:86.23]	90.08 [89.72:90.11]
Attention (*last layer*)	80.91 [79.99:81.07]	83.15 [82.91:83.51]	91.47 [91.39:91.56]
Attention (**SMaT**)	**91.48** [91.40:91.56]	**92.56** [92.28:92.83]	**92.84** [92.84:93.08]

Table 5.1: Results for the IMDB dataset with respect to student *simulability* in terms of accuracy. *Underlined* values indicate higher simulability than baseline with non-overlapping IQR.

Like the setting in (Pruthi, Dhingra, et al., 2020), we use the original training set with 25,000 samples to train the teacher, and further split the test set into a training set for the student and a dev and test set. We vary the number of samples the student is trained on between 500, 1,000, and 2,000. We evaluate *simulability* using accuracy (i.e., what percentage of student predictions match with teacher predictions). The teacher model obtains 91% accuracy on the student test set.

Table 5.1 shows the results in terms of simulability (Equation 5.1) for the three settings. We can see that, overall, the attention explainer trained with SMaT leads to students that simulate the teacher model much more accurately than students trained without any explanations, and more accurately than students trained with any *static* explainer across all student training set sizes. Interestingly, the gradient-based explainers only improve over the baseline students when the amount of training data is very low, and actually degrade simulability for larger amounts of data Using only heads from the last layer seems to have the opposite effect, leading to higher simulability than all other static explainers only for larger training sets.

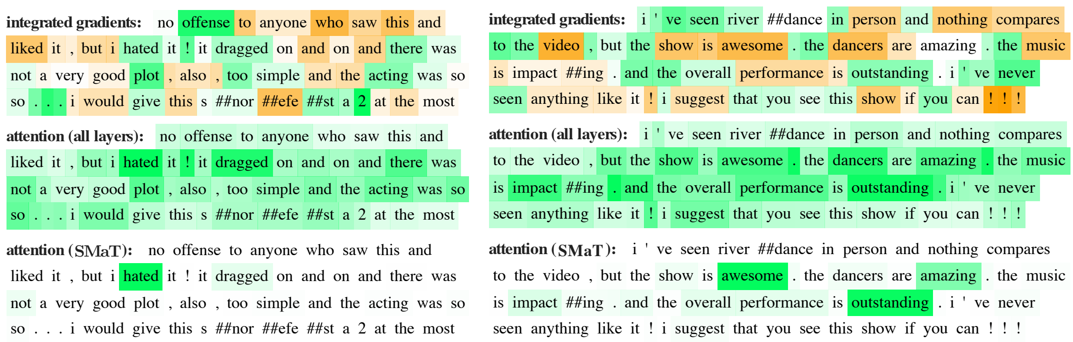

Figure 5.3: Explanations given by integrated gradients, attention (*last layer*), and our learned attention explainer (SMaT) for two movie reviews of the IMDB dataset (negative and positive examples). Green and orange represent positive and negative contributions, respectively.

	AUC
Gradient L2	0.65
Gradient × Input	0.51
Integrated Gradient	0.53
Attention (*all layers*)	0.68
Attention (*last layer*)	0.61
Attention (**SMaT**)	**0.73**
Attention (*best layer*)*	0.75
Attention (*best head*)*	0.75

Table 5.2: *Plausibility* on *MovieReviews* in terms of AUC. * represents methods that use human labels.

Plausibility analysis. We select the median model trained with 1,000 samples and extract explanations for test samples from the MovieReviews dataset (DeYoung et al., 2020), which contains binary sentiment movie reviews from Rotten Tomatoes alongside human-rationale annotation. Since the labels are binary (indicating whether a token is part of the explanation or not) and the predicted scores are real values, we follow (Fomicheva, Lertvittayakumjorn, et al., 2021) and report our results in terms of the Area Under the Curve (AUC), which automatically considers multiple binarization thresholds. The results are shown in Table 5.2 along with two randomly selected examples of extracted explanations in Figure 5.3. We found that gradient-based explanations are less plausible than those using attention (with the exception of *Grad. L2*, which is similar to static attention) and that ones produced with SMaT achieve the highest plausibility, indicating that our learned explainer can produce human-like explanations while maximizing simulability. Moreover, SMaT achieves a similar AUC score to the best performing attention layer and head,[2] while not requiring *any* human annotations. This is evidence that scaffolded simulability, while not explicitly designed for it, is a good proxy for plausibility and "human-like" explanations.

5.5.2 Image Classification

To validate our framework across multiple modalities, we consider image classification on the CIFAR-100 dataset (Krizhevsky, 2009). We use as the base model the Vision Transformer (ViT) (Dosovitskiy et al., 2020), in particular the base version with 16×16 patches that was only pretrained on ImageNet-21k (Ridnik et al., 2021). We up-sample images to to a 224×224 resolution.

Since the self-attention mechanism in the ViT model only works with patch representations, the explanations produced by attention-based explainers will be at patch-level rather than pixel-level. We split the original CIFAR-100 training set into a new training set with 45,000 and a validation set with 5,000. Unlike the previous task, we reuse the training set for both the teacher and student, varying the number of samples the student is trained with between 2,250 (5%), 4,500 (10%) and 9,000 (20%). We use accuracy as the simulability metric and the teacher obtains 89% on test set.

Table 5.3 shows the results for the three settings. Similarly to the results in the text modality, the attention explainer trained with SMaT achieves the best scaffolding performance, although the gaps to static attention-based explainers are smaller (especially when

[2] AUC scores obtained by independently trying all attention heads and layers of the model.

	2,250	4,500	9,000
No Explainer	81.16 [80.98:81.26]	84.02 [83.98:84.24]	85.20 [85.17:85.26]
Gradient L2	80.97 [80.91:81.10]	83.98 [83.81:84.23]	85.13 [84.97:85.50]
Gradient × Input	80.93 [80.82:81.04]	83.99 [83.98:84.13]	85.33 [84.85:85.35]
Integrated gradients	80.22 [80.17:80.35]	83.44 [83.25:83.44]	84.99 [84.76:85.22]
Attention (*all layers*)	<u>82.53</u> [82.53:82.62]	<u>84.81</u> [84.74:84.92]	<u>**85.92**</u> [85.78:85.94]
Attention (*last layer*)	<u>82.34</u> [82.30:82.60]	<u>84.65</u> [84.56:84.81]	85.31 [84.84:85.31]
Attention (**SMaT**)	<u>**83.09**</u> [82.77:83.28]	<u>**85.42**</u> [85.39:85.85]	<u>**85.96**</u> [85.74:86.35]

Table 5.3: *Simulability* results, in terms of accuracy (%), on the CIFAR100 dataset. *Underlined* values represent better performance than baseline with non-overlapping IQR

students are trained with more samples). Here, the gradient-based explainers always degrade simulability across the tested training set sizes and and it seems important that the explanations include attention information from layers other than the last one.

Plausibility analysis. Since there are no available human annotations for plausibility in the CIFAR-100 dataset, we design a user study to measure the plausability of the considered methods. The original image and explanations extracted with Gradient × Input, Integrated Gradients, Attention (*all layers*), and Attention (SMaT) are shown to the user, and the user has to rank the different explanations to answer the question *"Which explanation aligns the most with how you would explain a similar decision?"*. Explanations were annotated by three volunteers. After collecting results, we compute the *rank* and the *TrueSkill* rating (Herbrich et al., 2007) for each explainer (roughly, the "skill" level if the explainers where players in game). The results are shown in Table 5.4. As in the previous task, attention trained with SMaT outperforms all other explainers in terms of plausibility, and its predicted *rating* is much higher than all other explainers. We also show examples of explanations for a set of randomly selected images in Figure 5.4.

	Rank	*TrueSkill*
Grad. × Input	3-4	-2.7±.67
Integ. Grad.	3-4	-2.1±.67
Attn. (*all lx.*)	2	0.7±.67
Attn. (**SMaT**)	**1**	**4.3±.70**

Table 5.4: *Plausibility* results of the human study on visual explanations.

5.5.3 Machine Translation Quality Estimation

Quality Estimation (QE) is the task of predicting a quality score given a sentence in a source language and a translation in a target language from a machine translation system, which requires models that consider interactions between the two inputs, source and target. Scores tend to be continuous values (making this a regression task) that were collected from expert annotators.

Interpreting quality scores of machine translated outputs is a problem that has re-

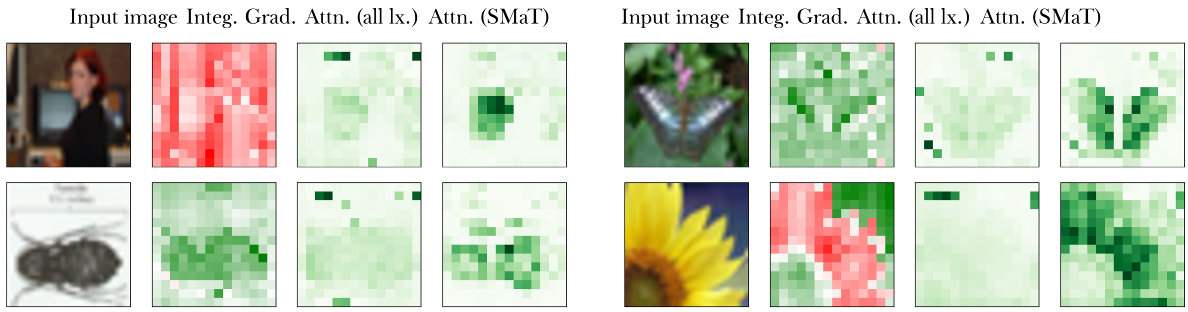

Figure 5.4: Explanations given by integrated gradients, attention (all layers), and learned attention explainer for a set of input images of CIFAR-100. Gold labels are: "television", "butterfly", "cockroach", and "sunflower".

	2,100	4,200	8,400
No Explainer	.7457 [.7366:.7528]	.7719 [.7660:.7802]	.7891 [.7860:.7964]
Gradient L2	$\underline{.8065}$ [.8038:.8268]	$\underline{.8535}$ [.7117:.8544]	$\underline{\mathbf{.8638}}$ [.8411:.8657]
Gradient × Input	.6846 [.6781:.6894]	.6922 [.6885:.6965]	.7141 [.7136:.7147]
Integrated gradients	.6686 [.6677:.6694]	.7086 [.6994:.7101]	.7036 [.6976:.7037]
Attention (*all layers*)	$\underline{.8120}$ [.7955:.8125]	$\underline{.8193}$ [.8186:.8280]	$\underline{\mathbf{.8467}}$ [.8464:.8521]
Attention (*last layer*)	.7486 [.7484:.7534]	.7720 [.7672:.7726]	.7798 [.7717:.7814]
Attention (**SMaT**)	$\underline{\mathbf{.8156}}$ [.8096:.8183]	$\underline{\mathbf{.8630}}$ [.8412:.8724]	$\underline{\mathbf{.8561}}$ [.8512:.8689]

Table 5.5: *Simulability* results, in terms of Pearson correlation, on the ML-QE dataset. *Underlined* values represent better performance than baseline with non-overlapping IQR.

ceived recent interest (Fomicheva, Lertvittayakumjorn, et al., 2021) since it allows identifying which words were responsible for a bad translation. We use the MLQE-PE dataset (Fomicheva et al., 2020), which contains 7,000 training samples for each of seven language pairs alongside word-level human annotation. We use as the base model a pretrained XLM-R-base (Conneau et al., 2019), a multilingual model with 12 layers and 12 heads in each (total of 144 heads).

We exclude one of the language pairs in the dataset (`si-en`) since the XLM-R model did not support it, leading to a training set with 42,000 samples. Similar to the CIFAR-100 case, we reuse the same training set for both the teacher and student, sampling a subset for the latter. We vary the number of samples the student is trained with between 2,100 (5%), 4,200 (10%) and 8,400 (20%). Since this is a regression task, we evaluate simulability using the Pearson correlation coefficient between student and teacher's predictions.[3] The teacher achieves 0.63 correlation on the test set.

Table 5.5 shows the results for the three settings. Similar to other tasks, the attention explainer trained with SMaT leads to students with higher simulability than baseline students and similar or higher than *static* explainer across all training set sizes. Curiously, the

[3]Pearson correlation is the standard metric used to evaluate sentence-level QE models.

	EN-DE		EN-ZH		ET-EN		NE-EN		RO-EN		RU-EN		OVERALL	
	src.	tgt.	src.	tgt.	src.	tgt.	src.	tgt.	src.	tgt.	src.	tgt.	src.	tgt.
Gradient L2	**0.64**	**0.65**	0.65	0.49	**0.67**	0.61	**0.68**	**0.55**	**0.72**	0.68	**0.65**	0.54	**0.67**	0.59
Gradient × Input	0.58	0.60	0.61	0.51	0.60	0.54	0.61	0.49	0.64	0.59	0.58	0.51	0.61	0.54
Integrated Gradients	0.59	0.60	0.63	0.49	0.60	0.52	0.64	0.48	0.64	0.59	0.60	0.51	0.62	0.53
Attention (*all layers*)	0.60	0.63	**0.68**	**0.52**	0.60	0.61	0.58	**0.55**	0.66	**0.70**	0.62	**0.55**	0.62	0.59
Attention (*last layer*)	0.51	0.49	0.61	0.49	0.51	0.50	0.55	0.48	0.52	0.57	0.56	0.50	0.54	0.50
Attention (**SMaT**)	**0.64**	**0.65**	**0.68**	**0.52**	0.66	**0.64**	0.66	0.54	0.71	**0.70**	0.61	0.54	0.66	**0.60**
Attention (*best layer*)*	0.64	0.65	0.69	0.64	0.64	0.68	0.68	0.68	0.71	0.76	0.64	0.59	0.65	0.65
Attention (*best head*)*	0.67	0.67	0.70	0.65	0.70	0.70	0.70	0.69	0.73	0.75	0.67	0.60	0.67	0.66

Table 5.6: Plausibility results for source and target inputs for each language pair of the MLQE-PE dataset in terms of AUC. * represents *supervised* methods that use human labels in some form.

Grad. L2 explainer achieves very high simulability for this task. It even has a higher *median* simulability score than SMaT for 8,400 samples. However, we attribute this to variance in the student training set sampling (that could lead to an imbalance in language pair proportions) which could explain why SMaT performance degrades with more samples. For this task, the gradient-based explainers always degrade simulability across the tested training set size. It also seems that using only the last layer's attention is also ineffective at teaching students, achieving the same performance as the baseline.

Plausibility analysis. We select the median model trained with 4,200 samples and follow the approach devised in the Explainable QE shared task to evaluate plausibility (Fomicheva, Lertvittayakumjorn, et al., 2021), which consists of evaluating the human-likeness of explanations in terms of AUC only on the subset of translations that contain errors. The results are shown in Table 5.6. We note that for all language pairs, SMaT performs on par or better than static explainers, and only being surpassed by *Grad. L2* in the *source-side* over all languages. Comparing with the best attention layer/head, an approach used by Fomicheva, Specia, and Aletras (2021) and Treviso et al. (2021), SMaT achieves similar AUC scores for source explanations, but lags behind the best attention layer/head for target explanations on *-EN language pairs. However, as stressed previously for text and image classification, SMaT sidesteps human annotation and avoids the cumbersome approach of independently computing plausibility scores for all heads.

5.6 Related Work

Explainability for text & vision. Several works propose explainability methods to interpret decisions made by NLP and CV models. Besides gradient and attention-based approaches already mentioned, some extract explanations by running the models with perturbed inputs (Feng et al., 2018; S. Kim et al., 2020; Ribeiro et al., 2016). Others even define custom backward passes to assign relevance for each feature (Bach et al., 2015). These methods are commonly employed together with post-processing heuristics, such as selecting only the top-k tokens/pixels with higher scores for visualization. Another line of work seeks to build a classifier with inherently interpretable components, such as methods based on attention mechanisms and rationalizers (Bastings et al., 2019; Lei et al., 2016).

Evaluation of explainability methods. As mentioned in the introduction, early works evaluated explanations based on properties such as *consistency*, *sufficiency* and *comprehensiveness*. Jacovi and Goldberg (2020) recommended the use of a graded notion of faithfulness, which the ERASER benchmark quantifies using the idea of sufficient and comprehensive rationales, alongside compiling datasets with human-annotated rationales for calculating plausibility metrics (DeYoung et al., 2020). Given the disagreement between explainability methods, Neely et al. (2021) showed that without a faithful ground-truth explanation it is impossible to determine which method is better. Diagnostic tests such as the ones proposed by Adebayo et al. (2018) and Wiegreffe and Pinter (2019) and Atanasova et al. (2020) are more informative yet they do not capture the main goal of an explanation: the ability to communicate an explanation to a practitioner.

Simulability. A new dimension for evaluating explainability methods relies on the forward prediction/simulation proposed by Lipton (2016a) and Doshi-Velez and Kim (2017), which states that humans should be able to correctly simulate the model's output given the input and the explanation. Arora et al. (2022), Chandrasekaran et al. (2018), and Hase and Bansal (2020) analyze simulability via human studies across text classification datasets. Treviso and Martins (2020) designed an automatic framework where students (machine or human) have to predict the model's output given an explanation as input. Similarly, Pruthi, Dhingra, et al. (2020) proposed the simulability framework that was extended in our work, where explanations are used to regularize the student rather than passed as input.

Learning to explain. The concept of simulability also opens a path to learning explainers. In particular Treviso and Martins (2020) learn an attention-based explainer that maximizes simulability. However, directly optimizing for simulability sometimes led to explainers that learned trivial protocols (such as selecting only punctuation symbols or stopwords to leak the label). Our approach of optimizing a teacher-student framework is similar to approaches that optimize for model distillation (Zhou et al., 2021). However, these approaches modify the original model rather than introduce a new explainer module. Raghu et al. (2021) propose a framework similar to ours for learning *commentaries* for inputs that speed up and improve the training of a model. However commentaries are model-independent and are optimised to improve performance on the real task. Rationalizers (J. Chen et al., 2018; Guerreiro & Martins, 2021; Jacovi & Goldberg, 2021) also directly learn to extract explanations, but can also suffer from trivial protocols.

5.7 Conclusion & Subsequent Work

We proposed **SMaT**, a framework for directly optimizing explanations of the model's predictions to improve the training of a student *simulating* the said model. We found that, across tasks and domains, explanations learned with SMaT both lead to students that simulate the original model more accurately and are more aligned with how people explain similar decisions when compared to previously proposed methods. On top of that, our parameterized attention explainer provides a principled way for discovering relevant attention heads in transformers.

Our work shows that scaffolding/simulability is a suitable criterion for both evaluating and optimizing explainability methods. Crucially, we believe that there are further unexplored connections between explainability and model distillation: for instance, subsequent work has shown *chain-of-thought* rationales extracted from large LLMs can be used to improve the training of smaller models (L. H. Li et al., 2023), which means that these can indeed be *good* explanations for models' decisions.

Chapter 6

Can Reasoning Models Explain and Improve Low-Resource MT?

Large language models (LLMs) have shown remarkable capabilities in complex reasoning tasks, often attributed to their ability to generate *chain-of-thought* (CoT) rationales. This has led to the surge of *reasoning models* that are optimized to solve hard reasoning problems (mathematical, logical, etc.) by leveraging the additional computation in producing the tokens in their reasoning *traces*. This additional "reasoning" has also been shown effective at improving performance in other tasks, including machine translation (MT).

In this chapter, we investigate whether such reasoning models can produce effective explanations for machine translation and the translation process. Leveraging the scaffolding criteria from the previous chapter, we evaluate the explanatory power of the reasoning traces produced by state-of-the-art reasoning models (such as Gemini 2.5 Pro and DeepSeek R1) by leveraging them to teach smaller LLMs to translate into very low-resource language pairs. Our findings suggest that, besides being state-of-the-art MT systems, reasoning models' explanations do have some explanatory power even in problems like machine translation, particularly when training students through *posthoc regularization* (predicting the explanation after the translation) rather than CoT distillation.

This chapter is based on unpublished work.

6.1 Introduction

One of the suprising emergent capabilities of large language models (LLMs) is the effectiveness of *chain-of-thought* prompting, where prompting the model to produce a series of intermediate reasoning steps (*"let's think step-by-step"*) leads to additional intermediate generation tokens before a (generally better) output (Wei et al., 2022). Its discovery led to considerable interest in leveraging this mechanism as an adaptive way to improve task performance at the cost of additional *test-time compute*. This, coupled with the plateau-

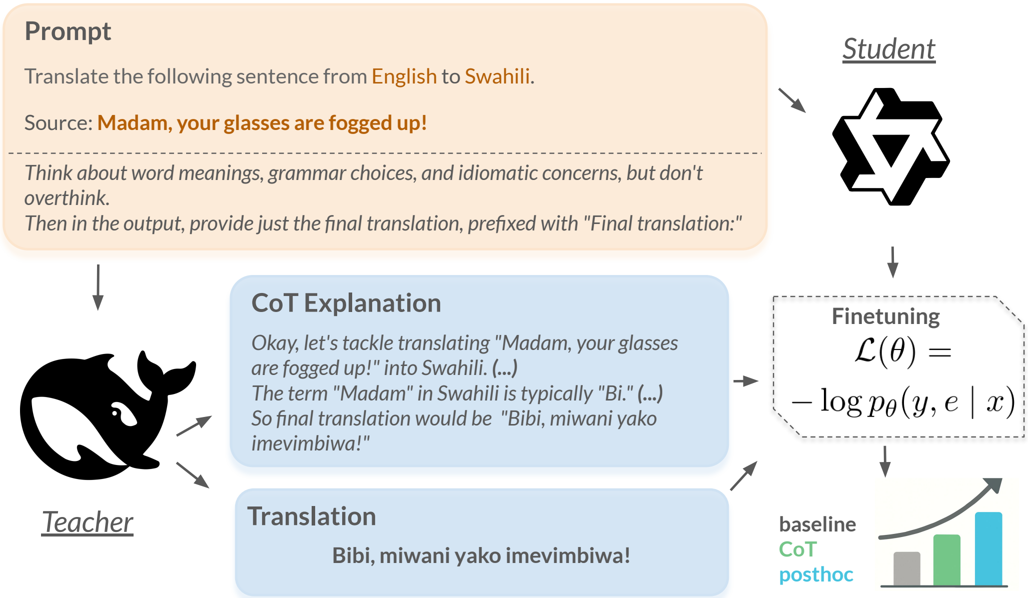

Figure 6.1: Illustration of a teacher–student framework use to measure the explanatory power of reasoning models' explanations for MT. Text in the *prompt* under the dotted line is only passed to the teacher.

ing effects of scaling training compute, precipitated the surge and widespread adoption of *reasoning models*: LLMs that are *post-trained* to solve hard reasoning problems that can be verified, such as mathematical and coding problems, by producing increasingly long reasoning *traces* (DeepSeek-AI, 2025). Remarkably, these models show not only better performance at reasoning tasks, but sometimes also outperform non-reasoning LLMs on generative, non-verifiable tasks such as machine translation (Deutsch et al., 2025).

The additional text produced is often intepreted as a free-form *rationale* for the model's decision, and there is widespread debate about the usefulness of these explanations as insights into the model's decision-making process (Lanham et al., 2023; Turpin et al., 2023). Nevertheless, prior work has shown that these explanations are effective at teaching smaller models to perform reasoning tasks (DeepSeek-AI, 2025; Ho et al., 2023), but their explanatory power in other more open-ended tasks has not been fully explored.

In this chapter, we investigate whether reasoning models can provide good machine translation systems, and whether their reasoning traces can be used to teach smaller models. In particular, we focus on low-resource language pairs, where the low baseline performance of current systems and small amounts of training data make the explanatory power of reasoning traces particularly valuable. We evaluate recent, state-of-the-art reasoning models such as Gemini 2.5 Pro and DeepSeek R1, on English→Swahili and English→Zulu, using the WMT24++ testsets (Deutsch et al., 2025), and find that *reasoning* seems to benefit (or at least, not degrade) low-resource MT, with Gemini 2.5 Pro outperforming more *"non-reasoning"* LLMs.

Then, we finetune smaller pretrained LLMs from the Qwen family to translate between the two language pairs using the Smol dataset (Caswell et al., 2025) (a minimal highly-curated dataset with less than 1000 parallel sentences) augmented with *CoT explanations* from reasoning models. By comparing different ways of *explanation-augmented* fine-tuning, we find that explanations from reasoning models uniformly help students to learn low-resource MT, and that the value of these explanations isn't tied to the teacher models' performance, with DeepSeek R1 producing particularly effective explanations, despite their weaker translation performance compared to Gemini 2.5 Pro. Surprisingly, we also find that traditional CoT distillation proves relatively ineffective, but *posthoc regularization* (where the student is trained to generate the explanation after the translation) consistently leads to the best student models, with the added benefit of no extra test-time compute.

6.2 Related Work

Knowledge Distillation for MT Knowledge distillation (KD) (which the scaffolding criteria can be seen as) has been widely used to train compact MT models from larger teachers. *Sequence-level* KD was introduced in the context of MT (Y. Kim & Rush, 2016) and there is some understanding of why synthetic data works (D. Zhang et al., 2018). However, traditional KD information signals like full *logits* distributions are hardly *interpretable* to humans, and provide little hope of insight into the model's decision-making process.

Chain-of-Thought Distillation Small LLMs struggle to produce and benefit from *chain-of-thought* (CoT) prompting on their own and distilling the CoT *rationale* as an information signal is particularly effective at improving their capabilities in reasoning tasks (Fu et al., 2023; Ho et al., 2023; S. Li et al., 2022). Reasoning models seem particularly effective at producing explanations for these tasks, with distilled models far surpassing counterparts optimized for the same task (DeepSeek-AI, 2025). However, prior work has primarily focused on traditional CoT distillation, and alternative approaches to leverage these explanation (like post-hoc regularization) haven't been extensively explored.

Are *chains-of-thought* good explanations? The human-readable nature of CoT rationales has naturally led to claims that these explanations increase the *interpretability* of LLMs (S. Li et al., 2022; Wei et al., 2022). However, there is widespread debate if these explanations are *faithful* to the models' decision process. Turpin et al. (2023) found that models don't always rely on their CoT to arrive at decisions and reason *posthoc*, generating plausible explanation that ignore explicitly introduced biasing factors. Lanham et al. (2023) nevertheless found that while chain of thought reasoning is not always faithful, it is possible to find conditions where it is more faithful (e.g. more capable models are more likely to actually rely on their *rationales*) and reasoning models seem to produce more faithful explanations in specific synthetic tasks. In this chapter, we evaluate explanations for the more implicit, open-ended problem of machine translation, taking the instrumentalist approach of evaluating explanations by how much they help teaching (*scaffolding*) since we know it *correlates* with other properties such as plausability (Chapter 5).

6.3 Explanation-Augmented MT Finetuning

Given a dataset $\{x^{(i)}, e_T^{(i)}, y_T^{(i)}\}_{i=1}^N$ where $x^{(i)}$ is a source sentence and $e^{(i)}$ and $y^{(i)}$ are the CoT explanation and final translation from a teacher model asked to translate the former $e^{(i)}, y^{(i)} = T(x^{(i)})$, we compare three different approaches to finetune a student model.

- A *baseline*, sequence-level knowledge distillation approach, where the student is trained to predict the translation from the source $p(y_i|x_i)$.
- A *CoT distillation* approach, where the student is trained to predict the explanation $p(e_i|x_i)$ and finish with the translation $p(y_i|x_i, e_i)$ (as the teacher originally did). This is the standard distillation approach, and leads to students leads to students that require additional test-time compute during inferenc.
- A *posthoc regularization* approach, where the student is trained to directly predict the translation $p(y_i|x_i)$ and *after* predict the explanation $p(e_i|x_i, y_i)$. This is inspired by our previous findings (Chapter 5) where good explanations helped through a regularization effect during training, rather than simply inducing and requiring extra computation during inference.

6.4 Experiments

6.4.1 Experimental Setup

Datasets & Language Pairs To train our student models for low-resource machine translation, we use the SMOL dataset (Caswell et al., 2025), a highly-curated professionally translated dataset into 221 languages. In particular, we use the sentence-level SMOLSENT subset, containing 863 sentences covering the 5.5k most common English words. For evaluation, we use the WMT24++ test sets (Deutsch et al., 2025), an extension of the WMT24 shared task (Kocmi et al., 2024) data to 55 languages. Despite the large number of languages covered by both, they overlap on only four languages (Arabic, Mandarin, Swahili and Zulu) and focus on English→Swahili and English→Zulu as the more low-resource language pairs.

Models As teacher models, we focus on recently released, state-of-the-art *reasoning* LLMs, optimized (generally through reinforcement learning) to solve hard reasoning tasks through the chain-of-thought mechanism. In particular, we use through the LiteLLM API (a) Gemini 2.5 Pro (Team, 2025) with a *"high"* thinking budget (4096 tokens) and (b) DeepSeek R1 (DeepSeek-AI, 2025) with max response size of 8192 tokens.

For our student models, we use the Qwen2.5 family of models (Qwen et al., 2025), in particular focusing on the 3B, 7B and 14B models. We use the *base* models as INSTRUCT variants have a higher likelihood of having seen these language pairs during training (but use them as baselines).

We use AdamW as our optimizer, and perform hyperparameter optimization separately for each approach, converging on $2\times10^{-5}/5\times10^{-5}$ learning rate for the baseline/explanation-regularized approaches and 5 epochs with a batch size of 16 sentences all students.

Evaluation To assess and compare the simulability of the reasoning traces of the different teacher models, we measure chrF (Popović, 2015) and XCOMET (Guerreiro et al., 2023) against the reference (human) translations.

6.4.2 Are Reasoning Models Good Low-Resource MT Systems?

Table 6.1 shows the performance of reasoning models, as well as standard *non-reasoning* models, in English→Swahili and English→Zulu, on both the SMOL and WMT24++. The most obvious finding is that Gemini 2.5 Pro dominates across all LPs, datasets and metrics, outperforming all other reasoning and non-reasoning models. For this model, however, thinking seems to play a small role, only marginally boosting performance. Interestingly, Deepseek R1 performs competitively with the other models, despite being primarily optimized for mathematical and coding tasks rather than translation. This is particularly the case for translating into the lower-resourced Zulu language, where it ranks second behind Gemini 2.5 Pro on the WMT24++. This suggests the reasoning capabilities may indeed transfer to traditional, non-thinking domains, particulary in low-resource settings.

	SMOL				WMT24++			
	→ Swahili		→ Zulu		→ Swahili		→ Zulu	
Model	chrF	XCOMET	chrF	XCOMET	chrF	XCOMET	chrF	XCOMET
GPT-4o*	–	–	–	–	58.6	0.688	56.1	0.494
Claude 3.5 Sonnet*	–	–	–	–	55.1	0.677	52.1	0.463
Gemini 2.5 Pro (NT)	–	–	–	–	60.0	0.719	**58.7**	0.508
o1*	–	–	–	–	56.3	0.700	53.1	0.485
Gemini 2.5 Pro	**56.2**	**0.815**	**53.7**	**0.582**	**60.1**	**0.721**	58.4	**0.509**
DeepSeek R1	56.0	0.784	51.1	0.551	56.2	0.668	54.2	0.501

Table 6.1: Performance comparison across different models for English to → Swahili, → Zulu translation on SMOL, WMT24++ dataset(s). * models' performance was obtain directly from WMT24++ paper (Deutsch et al., 2025). (NT) means translations were obtained in a *non-reasoning* mode.

Figure 6.2 shows an example source from the SMOL dataset, and the reasoning traces from DeepSeek R1 and Gemini 2.5 Pro.

Source: *With amazing enthusiasm they are at work, and wherever I pass they wave and greet with a loud call.*

Gemini 2.5 Pro

Analysis and Translation of "With amazing enthusiasm they are at work, and wherever I pass they wave and greet with a loud call."

Okay, so the task is to translate this sentence into Swahili. Let's break it down. "With amazing enthusiasm" - that's a modifier. We've got *hamasa* (enthusiasm), *ari* (zeal), and *shauku* (desire) as possibilities. *Hamasa* feels like the best fit here. "Amazing" is *ajabu* or *kushangaza* - both work. "With" is *kwa*. So, "with amazing enthusiasm" becomes *kwa hamasa ya ajabu*. Or, perhaps *kwa ari ya kushangaza* if we want a slightly different feel, or even *wakiwa na hamasa ya ajabu* ("while they have amazing enthusiasm") for a more literal but still natural phrase.

(...)

A key point is that "they wave and greet with a loud call" suggests a habitual action. Swahili has a habitual tense! So, how can I best incorporate that? Maybe *hunipungia mikono* (they habitually wave at me) and *hunisalimu* (they habitually greet me)? The use of the habitual tense is a great idea to express that the act is a repeated one. Also, *mikono* (hands) vs. *mkono* (a hand). The original sentence says "they wave", so using the plural might be even better.

Final Translation:

Wanafanya kazi kwa hamasa ya ajabu, na popote ninapopita hunipungia mikono na kunisalimu kwa sauti kubwa.

Or, since *kupungia* on its own implies a waving action, maybe this:

Wanafanya kazi kwa hamasa ya ajabu, na popote ninapopita hunipungia na kunisalimu kwa sauti kubwa.

Translation: *Wanafanya kazi kwa hamasa ya ajabu, na popote ninapopita hunipungia mikono na kunisalimu kwa sauti kubwa.*

DeepSeek R1

Okay, let's tackle this translation. The sentence is: "With amazing enthusiasm they are at work, and wherever I pass they wave and greet with a loud call."

First, I need to break down each part. Starting with "With amazing enthusiasm..." The Swahili word for "with" here might be "kwa" since it's used to mean "with" in terms of manner. "Amazing enthusiasm" can be "bidii ya kushangaza" or "hamu ya kushangaza". "Bidii" is more about diligence, while "hamu" is eagerness or desire. "Kushangaza" means to amaze. So maybe "kwa bidii ya kushangaza" or "kwa hamu ya kushangaza". Hmm, "enthusiasm" might be closer to "hamu" here.

(...)

Putting it all together:

"Kwa hamu ya kushangaza wako kazini, na popote nilipopita wanapunga mikono na kusalimu kwa mwito mkubwa."

Wait, but "greet with a loud call" might need to adjust. Maybe "kutoa salamu kwa kupaiga kelele" but that might be too literal. Alternatively, "kusalimu kwa sauti kubwa" or "kwa mwito". "Mwito" is a call or shout.

(...)

Revising the first part: "Kwa hamu ya kushangaza wako kazini, na popote ninapopita wanapunga mikono na kusalimu kwa mwito mkubwa."

That seems better. Let me check each part again:

- "with amazing enthusiasm": "kwa hamu ya kushangaza" – correct? Yes, "hamu" is strong desire or eagerness, which fits enthusiasm.
- "they are at work": "wako kazini" – yes, more natural than "wanafanya kazi".

(...)

Okay, that should be the final translation. Let me make sure to use commas correctly and that the sentence structure flows naturally in Swahili.

Translation: *Kwa bidii ya kushangaza wako kazini, na popote ninapopita wanapunga mikono na kusalimu kwa sauti kubwa.*

Figure 6.2: Example qualitative comparison showing the explanation and final translation produced by Gemini 2.5 Pro and DeepSeek R1.

6.4.3 Do Reasoning Explanations Help Teach MT?

Given the success of reasoning models for low-resource MT, we now investigate wether the produced traces have any explanatory power, insofar as they help teach smaller models. Table 6.2 shows the *zero-shot* performance of both pretrained and instruction-tuned Qwen models, as well as finetuned versions of these on the SMOL dataset. Given the sheer amount of results, we highligh some core findings.

Teacher	Model Size	Explanation	→ Swahili chrF	→ Swahili XCOMET	→ Zulu chrF	→ Zulu XCOMET
Zero-Shot						
✗	3B	-	7.2	0.155	8.1	0.180
	+ INSTRUCT	-	10.8	0.239	7.4	0.241
	7B	-	7.3	0.168	6.7	0.197
	+ INSTRUCT	-	27.4	0.280	18.9	0.270
	14B	-	9.3	0.165	8.1	0.194
	+ INSTRUCT	-	34.3	0.302	24.8	0.280
Finetuned						
Human	3B	✗	29.3	0.319	22.9	0.305
	7B	✗	32.3	0.354	28.4	0.331
	14B	✗	39.7	0.386	30.2	0.329
DeepSeek R1	3B	✗	29.7	0.318	24.3	0.308
		CoT	13.8	0.190	14.2	0.189
		posthoc	**33.4**	0.333	**28.4**	0.317
	7B	✗	34.8	0.357	28.9	0.328
		CoT	15.4	0.225	15.4	0.207
		posthoc	**40.0**	**0.385**	**33.9**	**0.345**
	14B	✗	**39.7**	0.377	31.4	0.336
		CoT	16.0	0.237	16.1	0.224
		posthoc	39.2	**0.411**	**39.0**	**0.348**
Gemini 2.5 Pro	3B	✗	26.9	0.319	23.7	0.306
		CoT	26.4	**0.336**	24.0	0.307
		posthoc	24.8	0.333	17.7	**0.318**
	7B	✗	33.8	0.351	26.4	0.322
		CoT	27.9	0.367	24.0	0.321
		posthoc	33.7	0.382	30.1	0.345
	14B	✗	38.3	0.372	31.5	0.330
		CoT	28.1	0.378	27.3	0.342
		posthoc	38.8	0.405	32.1	0.346

Table 6.2: Performance of different models from the Qwen2.5 family, when evaluated on → Swahili, and → Zulu using the WMT24++ test set. *Finetuned* version used the SMOL dataset, either with original *human* translation, or using the translations (and explanations) from different reasoning models.

Small LLMs struggle with low-resource translation. The zero-shot results highlight the challenge facing smaller models in low-resource settings. Pretrained models perform extremely poorly but even instruction-tuned variants show weak performance, barely achieving 0.3 XCOMET for either of the LPs, and well below the performance state-of-the-art models ($6.4.2). However, even simple finetuning on the (minimal) SMOL dataset, without any explanation, can dramatically improve performance, with all finetuned (pretrained) models outperforming their instruction-tuned counterparts, and with the 14B model almost achieving 40 chrF/0.4 XCOMET for English → Swahili translation.

Reasoning traces help smaller models learn, but only through post-hoc regularization. When the consider explanation-augmented finetuning, the most striking finding is that models finetuned trought standard CoT distillation perform poorly (particularly when using R1 as teacher) barely outperforming just finetuning with the translations. In contrast, *post-hoc* regularization consistently outperforms both baseline finetuning and CoT distillation across all model sizes, particularly for the lower-resourced → Zulu LP.

DeepSeek R1 produces more useful explanations, despite weaker translation performance. Interestingly, despite the state-of-the-art performance of Gemini 2.5 Pro, DeepSeek R1's explanations proved more effective training student models, with the best student models always using R1's explanations (through post-hoc regularization). Why this is the case is not clear, but could be related to the fact that reasoning seems to have little effect on Gemini 2.5 Pro's translation performance .

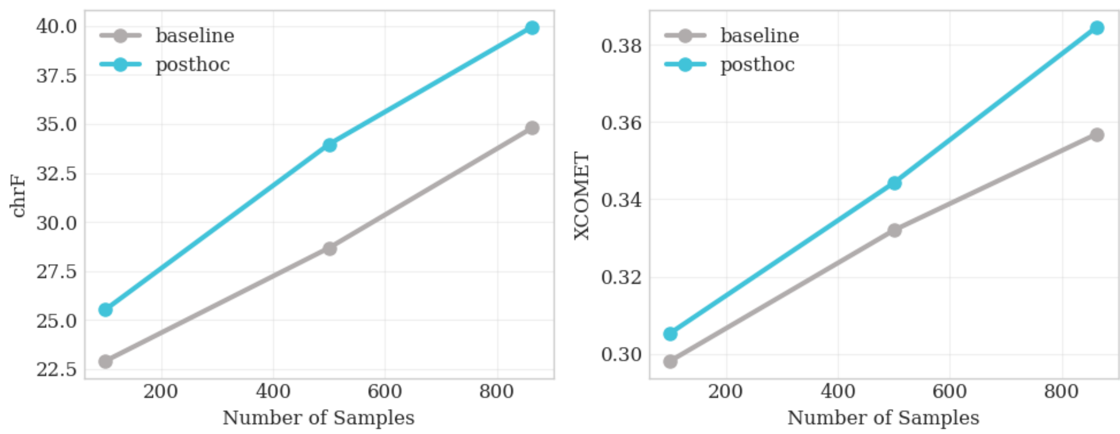

Figure 6.3: Performance scaling as we increase the amount of parallel sentences and explanations used for finetuning a Qwen2.5 7B model

Explanation-augmented finetuning works better with *more* data. Figure 6.3 shows how adding more parallel sentences and explanations (from DeepSeek R1) to a Qwen2.5 7B students' training data changes the performance. While explanations always seem to help when compared with just using the teacher's translations, suprisingly we found that the gap increased as we added more data, especially for English→Zulu translation.

6.5 Conclusion

In this chapter, we preliminarily investigated whether reasoning models are good *low-resource* machine translation systems, and whether they provide good explanations for this task by using their traces to teach smaller student LLMs. We start by showing that current *frontier* reasoning models have very strong performance when translating from English into Swahili and Zulu, with Gemini 2.5 Pro achieving state-of-the-art performance. Then, by adopting the *scaffolding* framework to evaluate explanatory power, we find that their explanations are provide valuable training signals to train smaller student models, and that explanation quality does not necessarily correlate with translation performance: despite Gemini 2.5 Pro's superior translation quality, DeepSeek R1's explanations proved more effective for teaching student models. However, suprisingly, we found that traditional chain-of-thought distillation failed to fully leverage these explanations, and that *post-hoc* regularization lead to best student models.

Our preliminary study has several limitations that suggest directions for future work. Most notably, we relied solely on automated metrics and the scaffolding framework to evaluate explanation quality, without conducting human studies to assess whether the reasoning traces are actually interpretable, plausible, and/or helpful to human translators. Additionally, our evaluation was limited to two language pairs and English as the source language, leaving questions about generalizability to other low-resource settings. However, despite these limitations, we believe our findings might have broader implications beyond machine translation: the effectiveness of post-hoc regularization offers a promising approach for distilling of expensive reasoning models into student models that don't use additional test-time computational costs. Additionally, optimizing models to give CoT explanations that help teach well (Cetin et al., 2025), akin to SMAT (5), could help increase trust in models (even if at a cost in accuracy).

Part III

Aligning with Feedback

Chapter 7

Quality-Aware Decoding for Neural MT

Despite the progress of scaling at improving machine translation systems, these systems still inherit biases from the data used to trained them, and the paradigm finding the most probable translation according to the model (MAP decoding) exacerbates these biases. Modern quality estimation and evaluation metrics have explicitly tried to address this problem by leveraging *human feedback*, but these advances were largely not reflected in MT systems themselves.

In this chapter, we bring together these two lines of research and propose *quality-aware decoding* for NMT, by leveraging recent breakthroughs in reference-free and reference-based MT evaluation through various inference methods like N-best reranking and minimum Bayes risk decoding. We perform an extensive comparison of various possible candidate generation and ranking methods across four datasets and two model classes and find that quality-aware decoding consistently outperforms MAP-based decoding according both to state-of-the-art automatic metrics (COMET and BLEURT) and to human assessments.

This chapter is based on Fernandes, Farinhas, et al. (2022).

7.1 Introduction

The most common procedure in neural machine translation (NMT) is to train models using maximum likelihood estimation (MLE) at training time (Equation 2.6), and to decode with beam search at test time, as a way to approximate maximum-a-posteriori (MAP) decoding. However, several works have questioned the utility of model likelihood as a good proxy for translation quality (Eikema & Aziz, 2020; Koehn & Knowles, 2017; Ott et al., 2018; Stahlberg & Byrne, 2019), with the most probable translation of an MLE-trained model often having poor quality[1]. In parallel, significant progress has been made in meth-

[1]While this is effectively a *alignment* problem between the models training/decoding objective and translation quality, the reason for this misalignment aren't fully clear.

ods for quality estimation and evaluation of generated translations (Mathur, Wei, et al., 2020; Specia et al., 2020) by training models on *translation feedback* from humans, but this progress is, by and large, not yet reflected in either training or decoding methods. Exceptions such as minimum risk training (Edunov et al., 2018; S. Shen et al., 2016) come at a cost of more expensive and unstable training, often with modest quality improvements.

An appealing alternative is to modify the decoding procedure only, separating it into two stages: *candidate generation* (where candidates are generated with beam search or sampled from the whole distribution) and *ranking* (where they are scored using a quality metric of interest, and the translation with the highest score is picked). This strategy has been explored in approaches using N-best reranking (Bhattacharyya et al., 2021; Ng et al., 2019) and minimum Bayes risk (MBR) decoding (Eikema & Aziz, 2021; Müller & Sennrich, 2021; Shu & Nakayama, 2017). While this previous work has exhibited promising results, it has mostly focused on optimizing lexical metrics such as BLEU or METEOR (Lavie & Denkowski, 2009; Papineni et al., 2002c), which have limited correlation with human judgments (Freitag, Foster, et al., 2021; Mathur, Baldwin, & Cohn, 2020) and only loosely aligning with translation quality. Moreover, a rigorous apples-to-apples comparison among this suite of techniques and their variants is still missing, even though they share similar building blocks.

Our work fills these gaps by asking the question:

> "*Can we leverage recent advances in MT quality evaluation to generate better translations? If so, how can we most effectively do so?*"

To answer this question, we systematically explore NMT decoding using a suite of ranking procedures. We take advantage of recent state-of-the-art learnable metrics, both reference-based, such as COMET and BLEURT (Rei et al., 2020a; Sellam et al., 2020), and reference-free (also known as *quality estimation*; QE), such as TransQuest and OpenKiwi (Kepler et al., 2019; Ranasinghe et al., 2020). We compare different ranking strategies under a unified framework, which we name **quality-aware decoding**. First, we analyze the performance of decoding using N-best reranking, both *fixed* according to a single metric and *learned* using multiple metrics, where the coefficients for each metric are optimized according to a reference-based metric. Second, we explore ranking using reference-based metrics directly through MBR decoding. Finally, to circumvent the expensive computational cost of the latter when the number of candidates is large, we develop a two-stage ranking procedure, where we use N-best reranking to pick a subset of the candidates to be ranked through MBR decoding. We explore the interaction of these different ranking methods with various candidate generation procedures including beam search, vanilla sampling, and nucleus sampling.

Experiments with two model sizes and four datasets reveal that while MAP-based decoding appears competitive when evaluating with lexical-based metrics (BLEU and ChrF), the story is very different with state-of-the-art evaluation metrics, where quality-aware decoding shows significant gains, both with N-best reranking and MBR decoding. We

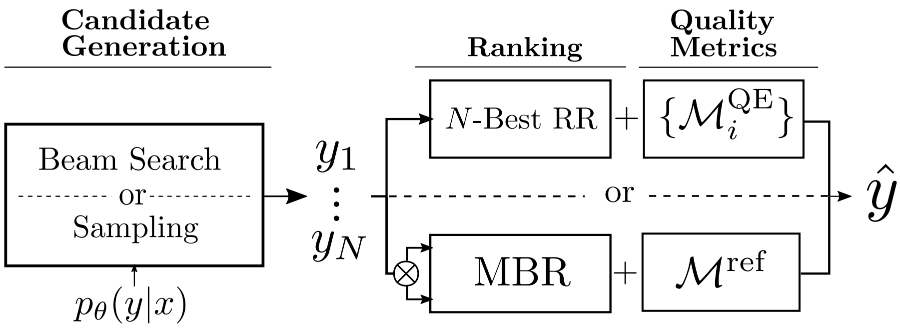

Figure 7.1: Quality-aware decoding framework. First, translation candidates are *generated* according to the model. Then, using reference-free and/or reference-based MT metrics, these candidates are *ranked*, and the highest ranked one is picked as the final translation.

perform a human-study to more faithfully evaluate our systems and find that, while performance on learnable metrics is not always predictive of the best system, quality-aware decoding usually results in translations with higher quality than MAP-based decoding.

7.2 Candidate Generation and Ranking

7.2.1 Candidate Generation

An NMT model defines a probability distribution $p_\theta(y|x)$ over a set of hypotheses $\mathcal{Y}$, conditioned on a source sentence x, where θ are learned parameters. A translation is typically predicted using MAP decoding, formalized as

$$\hat{y}_{\text{MAP}} = \underset{y \in \mathcal{Y}}{\operatorname{argmax}} \;\; \log p_\theta(y|x). \tag{7.1}$$

In words, MAP decoding searches for the most probable translation under $p_\theta(y|x)$, *i.e.*, the mode of the model distribution. Finding the exact $\hat{y}_{\text{MAP}}$ is intractable since the search space $\mathcal{Y}$ is combinatorially large, thus, approximations like **beam search** (Graves, 2012a; Sutskever et al., 2014) are used. However, it has been shown that the translation quality *degrades* for large values of the beam size (Koehn & Knowles, 2017; Meister et al., 2020; Murray & Chiang, 2018; Yang et al., 2018), with the empty string often being the true MAP hypothesis (Stahlberg & Byrne, 2019).

A stochastic alternative to beam search is to *draw samples* directly from $p_\theta(y|x)$ with ancestral sampling, optionally with variants that truncate this distribution, such as top-k sampling (Fan et al., 2018) or p-**nucleus sampling** (Holtzman et al., 2020) – the latter samples from the smallest set of words whose cumulative probability is larger than a predefined value p. Deterministic methods combining beam and nucleus search have also been proposed (Shaham & Levy, 2021).

Unlike beam search, sampling is not a search algorithm nor a decision rule – it is not expected for a single sample to outperform MAP decoding (Eikema & Aziz, 2020). However, samples from the model can still be useful for alternative decoding methods, as we shall see. While beam search focus on high probability candidates, typically similar to each other, sampling allows for more *exploration*, leading to higher candidate *diversity*.

7.2.2 Ranking

We assume access to a set $\bar{\mathcal{Y}} \subseteq \mathcal{Y}$ containing N candidate translations for a source sentence, obtained with one of the generation procedures described in §7.2.1. As long as N is relatively small, it is possible to (re-)rank these candidates in a post-hoc manner, such that the best translation maximizes a given metric of interest. We highlight two different lines of work for ranking in MT decoding: first, **N-best reranking**, using reference-free metrics as features; second, **MBR decoding**, using reference-based metrics.

N-best Reranking

In its simplest form (which we call *fixed* reranking), a *single* feature h is used (*e.g.*, an estimated quality score), and the candidate that maximizes this score is picked as the final translation,

$$\hat{y}_{\text{F-RR}} = \underset{y \in \bar{\mathcal{Y}}}{\operatorname{argmax}} \; h(x, y). \tag{7.2}$$

When *multiple* features $[h_1, \ldots, h_K]$ are available, one can tune weights $[w_1, \ldots, w_K]$ for these features to maximize a given reference-based evaluation metric on a validation set (Duh & Kirchhoff, 2008; Och, 2003) – we call this *tuned* reranking. In this case, the final translation is

$$\hat{y}_{\text{T-RR}} = \underset{y \in \bar{\mathcal{Y}}}{\operatorname{argmax}} \; \textstyle\sum_{k=1}^{K} w_k h_k(x, y). \tag{7.3}$$

Minimum Bayes Risk (MBR) Decoding

While the techniques above rely on *reference-free* metrics for the computation of features, MBR decoding uses *reference-based* metrics to rank candidates. Unlike MAP decoding, which searches for the most probable translation, MBR decoding aims to find the translation that maximizes the expected *utility* (equivalently, that minimizes *risk*, (Eikema & Aziz 2020; Kumar & Byrne 2002; 2004)).[2] Let again $\bar{\mathcal{Y}} \subseteq \mathcal{Y}$ be a set containing N hypotheses and $h(x, y^*, y)$ a utility function measuring the similarity between a hypothesis $y \in \mathcal{Y}$ and a reference $y^* \in \bar{\mathcal{Y}}$, optionally conditioned on the source (*e.g*, an automatic evaluation

[2] Utility can be seen as the translation quality of the hypothesis when compared to the reference.

metric such as BLEU or COMET). MBR decoding seeks for

$$\hat{y}_{\text{MBR}} = \underset{y \in \bar{\mathcal{Y}}}{\text{argmax}} \quad \underbrace{\mathbb{E}_{Y \sim p_\theta(y|x)}[h(x, Y, y)]}_{\approx \frac{1}{M}\sum_{j=1}^{M} h(x, y^{(j)}, y)}, \tag{7.4}$$

where in Eq. 7.4 the expectation is approximated as a Monte Carlo (MC) sum using model samples $y^{(1)}, \ldots, y^{(M)} \sim p_\theta(y|x)$[3]. In practice, the translation with the highest expected utility can be computed by comparing each hypothesis $y \in \bar{\mathcal{Y}}$ to all the other hypotheses in the set.

7.3 Quality-Aware Decoding

While recent works have explored various combinations of candidate generation and ranking procedures for NMT (Bhattacharyya et al., 2021; Eikema & Aziz, 2021; Lee et al., 2021; Müller & Sennrich, 2021), they suffer from two limitations:

- The ranking procedure is usually based on simple lexical-based metrics (BLEU, chrF, METEOR). Although these metrics are well established and inexpensive to compute, they correlate poorly with human judgments at segment level (Freitag, Foster, et al., 2021; Mathur, Wei, et al., 2020).
- Each work independently explores N-best reranking or MBR decoding, making unclear which method produces better translations.

In this chapter, we hypothesize that using more powerful metrics in the ranking procedure may lead to better quality translations. We propose a unified framework for ranking with both reference-based and reference-free metrics, independently of the candidate generation procedure. We explore four methods with different computational costs for a given number of candidates, N.

Fixed N-best Reranker. An N-best reranker using a single reference-free metric as a feature, according to Eq. 7.2. The computational cost of this ranker is $\mathcal{O}(N \times C_{\mathcal{M}^{\text{QE}}})$, where $C_{\mathcal{M}^{\text{QE}}}$ denotes the cost of running an evaluation with a metric $\mathcal{M}^{\text{QE}}$.

Tuned N-best Reranker. An N-best reranker using as features *all* the reference-free metrics in, along with the model log-likelihood $\log p_\theta(y|x)$. The weights in Eq. 7.3 are optimized to maximize a given reference-based metric $\mathcal{M}^{\text{ref}}$ using MERT (Och, 2003), a coordinate-ascent optimization algorithm widely used in previous work. Note that $\mathcal{M}^{\text{ref}}$ is used for tuning only; at test time, only reference-free metrics are used. Therefore, the decoding cost is $\mathcal{O}(N \times \sum_i C_{\mathcal{M}_i^{\text{QE}}})$.

[3]We also consider the case where $y^{(1)}, \ldots, y^{(M)}$ are obtained from nucleus sampling or beam search. Although the original MC estimate is unbiased, these ones are biased.

MBR Decoding. Choosing as the utility function a reference-based metric $\mathcal{M}^{\text{ref}}$, we estimate the utility using a simple Monte Carlo sum, as shown in Eq. 7.4. The estimation requires computing pairwise comparisons and thus the cost of running MBR decoding is $\mathcal{O}(N^2 \times C_{\mathcal{M}^{\text{ref}}})$.

N**-best Reranker → MBR.** Using a large number of samples in MBR decoding is expensive due to its quadratic cost. To circumvent this issue, we explore a *two-stage* ranking approach: we first rank all the candidates using a tuned N-best reranker, followed by MBR decoding using the top M candidates. The computational cost becomes $\mathcal{O}(N \times \sum_i C_{\mathcal{M}_i} + M^2 \times C_{\mathcal{M}^{\text{ref}}})$. The first ranking stage *prunes* the candidate list to a smaller, higher quality subset, making possible a more accurate estimation of the utility with less samples, and potentially allowing a better ranker than *plain* MBR for almost the same computational budget.

7.3.1 Reference-based Metrics

Reference-based metrics are the standard way to evaluate MT systems; the most used ones rely on the lexical overlap between hypotheses and reference translations (Lavie & Denkowski, 2009; Papineni et al., 2002c; Popović, 2015). However, lexical-based approaches have important limitations: they have difficulties recognizing correct translations that are paraphrases of the reference(s); they ignore the source sentence, an important indicator of meaning for the translation; and they do not always correlate well with human judgments, particularly at segment-level (Freitag et al., 2022)

In this work, apart from BLEU and chrF, we use the following state-of-the-art trainable reference-based metrics for both ranking and performance evaluation of MT systems:

- BLEURT (Pu et al., 2021; Sellam et al., 2020), trained to regress on human direct assessments (DA; (Graham et al. 2013)). We use the largest multilingual version, *BLEURT-20*, based on the RemBERT model (Chung et al., 2021).
- COMET (Rei et al., 2020a), based on XLM-R (Conneau et al., 2020), trained to regress on quality assessments such as DA using both the reference and the source to assess the quality of a given translation. We use the publicly available model developed for the WMT20 metrics shared task (*wmt20-comet-da*).

These metrics have shown much better correlation at segment-level than previous lexical metrics in WMT metrics shared tasks (Freitag, Rei, Mathur, Lo, Stewart, et al., 2021; Mathur, Wei, et al., 2020), mainly because they are directly trained to approximate human translation feedback (§2.3.3). Hence, as discussed in §7.2.2, they are good candidates to be used either *indirectly* as an optimization objective for learning the tuned reranker's feature weights, or *directly* as a utility function in MBR decoding. In the former, the higher the metric correlation with human judgment, the better the translation picked by the tuned reranker. In the latter, we approximate the expected utility in Eq. 7.4 by letting a candidate generated by the model be a reference translation – a suitable premise *if* the model is good in expectation.

7.3.2 Reference-free Metrics

MT evaluation metrics have also been developed for the case where references are not available – they are called *reference-free* or *quality estimation* (QE) metrics. These more closely approximate how humans experts give feedback to translations (Freitag, Foster, et al., 2021), and more closely approximate feedback models trained for other tasks, but have generally lacked when compared to reference-based metrics due the extra difficulty of the task. However in the last years, considerable improvements have been made to such metrics, with state-of-the-art models having increasing correlations with human annotators (Freitag, Rei, Mathur, Lo, Stewart, et al., 2021; Specia et al., 2021). These improvements enable the use of such models for ranking translation hypotheses in a more reliable way than before.

In this chapter, we explore four recently proposed reference-free metrics as features for N-best reranking, all at the sentence-level:

- COMET-QE (Rei et al., 2020b), a reference-free version of COMET (§7.3.1). It was the winning submission for the QE-as-a-metric subtask of the WMT20 shared task (Mathur, Wei, et al., 2020).
- TransQuest (Ranasinghe et al., 2020), the winning submission for the sentence-level DA prediction subtask of the WMT20 QE shared task (Specia et al., 2020). Similarly to COMET-QE this metric predicts a DA score.
- MBART-QE (Zerva et al., 2021a), based on the mBART (Y. Liu et al., 2020) model, trained to predict both the *mean* and the *variance* of DA scores. It was a top performer in the WMT21 QE shared task (Specia et al., 2021).
- OpenKiwi-MQM (Kepler et al., 2019; Rei et al., 2021), based on XLM-R, trained to predict the *multidimensional quality metric* (MQM; (Lommel et al. 2014)).[4] This reference-free metric was ranked second on the QE-as-a-metric subtask from the WMT 2021 metrics shared task.

7.4 Experiments

7.4.1 Setup

We study the benefits of quality-aware decoding over MAP-based decoding in two regimes:

[4]MQM annotations are expert-level type of annotations more fine-grained then DA, with individual errors annotated. See Table 2.3 for an example.

- A high-resource, unconstrained, setting with *large* transformer models (6 layers, 16 attention heads, 1024 embedding dimensions, and 8192 hidden dimensions) trained by Ng et al. (2019) for the WMT19 news translation task (Barrault et al., 2019), using English to German (EN $\rightarrow$ DE) and English to Russian (EN $\rightarrow$ RU) language pairs. These models were trained on over 20 million parallel and 100 million back-translated sentences, being the winning submissions of that year's shared task. We consider the non-ensembled version of the model and use *newstest19* for validation and *newstest20* for testing.
- A more constrained scenario with a *small* transformer model (6 layers, 4 attention heads, 512 embedding dimensions, and 1024 hidden dimensions) trained from scratch in *Fairseq* (Ott et al., 2019) on the smaller IWSLT17 datasets (Cettolo et al., 2012) for English to German (EN $\rightarrow$ DE) and English to French (EN $\rightarrow$ FR), each with a little over 200k training examples. We chose these datasets because they have been extensively used in previous work (Bhattacharyya et al., 2021) and smaller model allows us to answer questions about how the training methodology affects ranking performance.

We use beam search with a beam size of 5 as our decoding baseline because we found that it resulted in better or similar translations than larger beam sizes. For tuned N-best reranking, we use Travatar's (Neubig, 2013) implementation of MERT (Och, 2003) to optimize the weight of each feature, as described in §7.3.2. Finally, we evaluate each system using the metrics discussed in §7.3.1, along with BLEU and chrF (Popović, 2015).

7.4.2 Results

Overall, given all the metrics, candidate generation, and ranking procedures, we evaluate over 150 systems per dataset. We report subsets of this data separately to answer specific research questions, and defer to the published version of this chapter for additional results.

Impact of Candidate Generation

First, we explore the impact of the candidate generation procedure and the number of candidates.

Which candidate generation method works best, beam search or sampling? We generate candidates with beam search, vanilla sampling, and nucleus sampling. For the latter, we use $p = 0.6$ based on early results showing improved performance for all metrics.[5] For N-best reranking, we use up to 200 samples; for MBR decoding, due to the quadratic computational cost, we use up to 100.

[5]We picked nucleus sampling over top-k sampling because it allows varying support size and has outperformed top-k in text generation tasks (Holtzman et al., 2020).

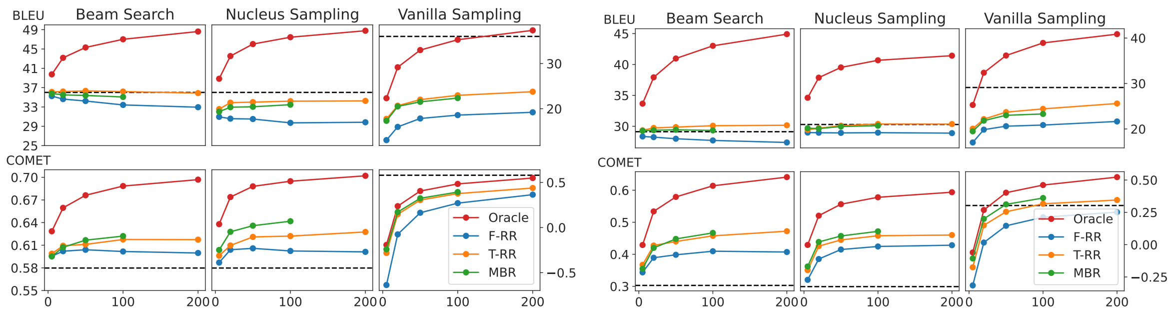

Figure 7.2: Values for BLEU (top) and COMET (bottom) for EN $\rightarrow$ DE as we increase the number of candidates for different generation and ranking procedures, as well as oracles with the respective metrics, for the *large* (left) and *small* (right) models. Baseline values (with beam size of 5) are marked with a dashed horizontal line.

Figure 7.2 shows BLEU and COMET for different candidate generation and ranking methods for the EN $\rightarrow$ DE WMT20 and IWSLT17 datasets, with increasing number of candidates. The baseline is represented by the dashed line. To assess the performance *ceiling* of the rankers, we also report results with an *oracle* ranker for the reported metrics, picking the candidate that maximizes it. For the *fixed* N-best reranker, we use COMET-QE as a metric, albeit the results for other reference-free metrics are similar. Performance seems to scale well with the number of candidates, particularly for vanilla sampling and for the *tuned* N-best reranker and MBR decoder. (Lee et al., 2021; Müller & Sennrich, 2021). However, all the rankers using vanilla sampling severely under-perform the baseline in most cases. In contrast, the rankers using beam search or nucleus sampling are competitive or outperform the baseline in terms of BLEU, and greatly outperform it in terms of COMET. For the larger models, we see that the performance according to the lexical metrics degrades with more candidates. In this scenario, rankers using nucleus sampling seem to have an edge over the ones that use beam search for COMET.

Based on the findings above, and due to generally better performance of COMET over BLEU for MT evaluation (Kocmi et al., 2021a), in following experiments we use nucleus sampling with the *large* model and beam search with the *small* model.

Impact of Ranking and Metrics

We now investigate the usefulness of the metrics presented in §7.3 as features and objectives for ranking. For N-best reranking, we use all the available candidates (200) while, for MBR, due to the computational cost of using 100 candidates, we report results with 50 candidates only (we found that ranking with *tuned* N-best reranking with $N = 100$ and MBR with $N = 50$ takes about the same time). We report results in Table 7.1, and use them to answer some specific research questions.

	Large (WMT20)				Small (IWSLT)			
	BLEU	chrF	BLEURT	COMET	BLEU	chrF	BLEURT	COMET
Baseline	**36.01**	63.88	0.7376	0.5795	29.12	56.23	0.6635	0.3028
F-RR w/ COMET-QE	29.83	59.91	<u>0.7457</u>	<u>0.6012</u>	<u>27.38</u>	54.89	<u>0.6848</u>	<u>0.4071</u>
F-RR w/ MBART-QE	<u>32.92</u>	<u>62.71</u>	0.7384	0.5831	27.30	<u>55.62</u>	0.6765	0.3533
F-RR w/ OpenKiwi	30.38	59.56	0.7401	0.5623	25.35	51.53	0.6524	0.2200
F-RR w/ Transquest	31.28	60.94	0.7368	0.5739	26.90	54.46	0.6613	0.2999
T-RR w/ BLEU	<u>35.34</u>	<u>63.82</u>	0.7407	0.5891	<u>**30.51**</u>	<u>**57.73**</u>	0.7077	0.4536
T-RR w/ BLEURT	33.39	62.56	<u>0.7552</u>	0.6217	30.16	57.40	<u>0.7127</u>	<u>0.4741</u>
T-RR w/ COMET	34.26	63.31	0.7546	<u>0.6276</u>	30.16	57.32	0.7124	0.4721
MBR w/ BLEU	<u>34.94</u>	<u>63.21</u>	0.7333	0.5680	29.25	56.36	0.6619	0.3017
MBR w/ BLEURT	32.90	62.34	<u>0.7649</u>	0.6047	28.69	56.28	<u>0.7051</u>	0.3799
MBR w/ COMET	33.04	62.65	0.7477	<u>0.6359</u>	<u>29.43</u>	<u>56.74</u>	0.6882	<u>0.4480</u>
T-RR+MBR w/ BLEU	<u>35.84</u>	<u>**63.96**</u>	0.7395	0.5888	<u>30.23</u>	<u>57.34</u>	0.6913	0.3969
T-RR+MBR w/ BLEURT	33.61	62.95	<u>**0.7658**</u>	0.6165	29.28	56.77	<u>**0.7225**</u>	0.4361
T-RR+MBR w/ COMET	34.20	63.35	0.7526	<u>**0.6418**</u>	29.46	57.13	0.7058	<u>**0.5005**</u>

Table 7.1: Evaluation metrics for EN $\rightarrow$ DE for the *large* and *small* model settings, using a *fixed* N-best reranker (F-RR), a *tuned* N-best reranker (T-RR), MBR decoding, and a two-stage approach. Best overall values are **bolded** and best for each specific group are <u>underlined</u>.

Which QE metric works best in a fixed N-best reranker? We consider a *fixed* N-best reranker with a single reference-free metric as a feature (see Table 7.1, second group). While none of the metrics allows for improving the baseline results in terms of the lexical metrics (BLEU and chrF), rerankers using COMET-QE or MBART-QE outperform the baseline according to BLEURT and COMET, for both the *large* and *small* models. Due to the aforementioned better performance of these metrics for translation quality evaluation, we hypothesize that these rankers produce better translations than the baseline. However, since the sharp drop in the lexical metrics is concerning, we will verify this hypothesis in a human study, in §7.4.2.

How does the performance of a tuned N-best reranker vary when we change the optimization objective? We consider a *tuned* N-best reranker using as features *all* the reference-free metrics in §7.3.2, and optimized using MERT. Table 7.1 (3rd group) shows results for EN $\rightarrow$ DE. For the *small* model, all the rankers show improved results over the baseline for all the metrics. In particular, optimizing for BLEU leads to the best results in the lexical metrics, while optimizing for BLEURT leads to the best performance in the others. Finally, optimizing for COMET leads to similar performance than optimizing for BLEURT. For the *large* model, although none of the rerankers is able to outperform the baseline in the lexical metrics, we see similar trends as before for BLEURT and COMET.

How does the performance of MBR decoding vary when we change the utility function? Table 7.1 (4th group) shows the impact of the utility function (BLEU, BLEURT,

	EN-DE (WMT20)					EN-RU (WMT20)				
	BLEU	chrF	BLEURT	COMET	Human	BLEU	chrF	BLEURT	COMET	Human
Reference	-	-	-	-	4.51	-	-	-	-	4.07
Baseline	**36.01**	**63.88**	0.7376	0.5795	4.28	**23.86**	51.16	0.6953	0.5361	3.62
F-RR w/ COMET-QE	29.83	59.91	0.7457	0.6012	4.19	20.32	49.18	0.7130	0.6207	3.25
T-RR w/ COMET	34.26	63.31	**0.7546**	0.6276	**4.33**	22.42	50.91	**0.7243**	0.6441	3.65
MBR w/ COMET	33.04	62.65	0.7477	0.6359	4.27	23.67	51.18	0.7093	0.6242	3.66
T-RR + MBR w/ COMET	34.20	63.35	0.7526	**0.6418**	4.30	23.21	**51.26**	0.7238	**0.6736**	**3.72**†
	EN-DE (IWSLT17)					EN-FR (IWSLT17)				
Reference	-	-	-	-	4.38	-	-	-	-	4.00
Baseline	29.12	0.6635	56.23	0.3028	3.68	38.12	0.6532	63.20	0.4809	3.92
F-RR w/ COMET-QE	27.38	0.6848	54.89	0.4071	3.67	35.59	0.6628	60.90	0.5553	3.63
T-RR w/ COMET	**30.16**	**0.7124**	**57.32**	0.4721	**3.90**†	**38.60**	**0.7020**	**63.77**	0.6392	4.05†
MBR w/ COMET	29.43	0.6882	56.74	0.4480	3.79†	37.77	0.6710	63.24	0.6127	4.05†
T-RR + MBR w/ COMET	29.46	0.7058	57.13	**0.5005**	3.83†	38.33	0.6883	63.53	**0.6610**	**4.09**†

Table 7.2: Results for automatic and human evaluation. Top: WMT20 (large models); Bottom: IWSLT17 (small models). Methods with † are statistically significantly better than the baseline, with $p < 0.05$.

or COMET). For the *small* model, using COMET leads to the best performance according to all the metrics except BLEURT (for which the best result is attained when optimizing itself). For the *large* model, the best result according to a given metric is obtained when using that metric as the utility function.

How do (tuned) N-best reranking and MBR compare to each other? Looking at Table 7.1 we see that, for the *small* model, N-best reranking seems to perform better than MBR decoding in all the evaluation metrics, including the one used as the utility function in MBR decoding. The picture is less clear for the *large* model, with MBR decoding achieving best values for a given fine-tuned metric when using it as the utility; this comes at the cost of worse performance according to the other metrics, hinting at a potential "*overfitting*" effect. Overall, N-best reranking seems to have an edge over MBR decoding. We will further clarify this question with human evaluation in § 7.4.2.

Can we improve performance by combining N-best reranking with MBR decoding? Table 7.1 shows that, for both the *large* and the *small* model, the two-stage ranking approach described in §7.3 leads to the best performance according to the fine-tuned metrics. In particular, the best result is obtained when the utility function is the same as the evaluation metric. These results suggest that a promising research direction is to seek more sophisticated pruning strategies for MBR decoding.

Human Evaluation

Which metric correlates more with human judgments? How risky is it to optimize a metric and evaluate on a related metric? Our experiments suggest that, overall, *quality-aware* decoding produces translations with better performance across most metrics than

MAP-based decoding. However, for some cases (such as fixed N-best reranking and most results with the *large* model), there is a concerning "metric gap" between lexical-based and fine-tuned metrics. While the latter have shown to correlate better with human judgments, previous work has not attempted to explicitly optimize these metrics, and doing so could lead to ranking systems that learn to exploit "pathologies" in these metrics rather than improving translation quality. To investigate this hypothesis, we perform a human study across all four datasets. We ask annotators to rate, from 1 (no overlap in meaning) to 5 (perfect translation), the translations produced by the 4 *ranking* systems in §7.3, as well as the baseline translation and the reference. We choose COMET-QE as the feature for the fixed N-best ranker and COMET as the optimization metric and utility function for the tuned N-best reranker and MBR decoding, respectively. The reasons for this are two-fold: (1) they are currently the reference-free and reference-based metrics with highest reported correlation with human judgments (Kocmi et al., 2021a), (2) we saw the largest "metric gap" for systems based on these metrics, hinting of a potential "overfitting" problem (specially since COMET-QE and COMET are similar models).

Table 7.2 shows the results for the human evaluation, as well as the automatic metrics. We see that, with the exception of T-RR w/ COMET, when fine-tuned metrics are explicitly optimized for, their correlation with human judgments decreases and they are no longer reliable indicators of system-level ranking. This is notable for the fixed N-best reranker with COMET-QE, which outperforms the baseline in COMET for every single scenario, but leads to markedly lower quality translations. However, despite the potential for overfitting these metrics, we find that *tuned* N-best reranking, MBR, and their combination consistently achieve better translation quality than the baseline, specially with the small model. In particular, N-best reranking results in better translations than MBR, and their combination is the best system in 2 of 4 LPs.

7.5 Related Work

Reranking. Inspired by the work of L. Shen et al. (2004) on discriminative reranking for SMT, Lee et al. (2021) trained a large transformer model using a reranking objective to optimize BLEU. Our work differs in which our rerankers are much simpler and therefore can be tuned on a validation set; and we use more powerful quality metrics instead of BLEU. Similarly, Bhattacharyya et al. (2021) learned an energy-based reranker to assign lower energy to the samples with higher BLEU scores. While the energy model plays a similar role to a QE system, our work differs in two ways: we use an existing, pretrained QE model instead of training a dedicated reranker, making our approach applicable to any MT system without further training; and the QE model is trained to predict human assessments, rather than BLEU scores. Leblond et al. (2021) compare a reinforcement learning approach to reranking approaches (but not MBR decoding, as we do). They investigate the use of reference-based metrics and, for the reward function, a reference-free metric based on a modified BERTScore (T. Zhang et al., 2020). This new multilingual BERTScore is not fine-tuned on human judgments as COMET and BLEURT and it is unclear what its level of agreement with human judgments is. Another line of work is *generative reranking*, where the reranker is not trained to optimize a metric, but rather as a generative noisy-channel model (Ng et al., 2019; Yee et al., 2019; Yu et al., 2017). As a follow-up to our work, Farinhas et al. (2023) explored using LLMs to both generate translation candidates *and* to rerank (or even combine) them, but found that this was still inferior to using learned metrics in the reranking phase.

Minimum Bayes Risk Decoding. MBR decoding (Kumar & Byrne, 2002; 2004) has recently been revived for NMT using candidates generated with beam search (Shu & Nakayama, 2017; Stahlberg et al., 2017) and sampling (Eikema & Aziz, 2020; Müller & Sennrich, 2021). Eikema and Aziz (2021) also explore a two-stage approach for MBR decoding. Additionally, there is concurrent work by Freitag, Grangier, et al. (2021) on using neural metrics as utility functions during MBR decoding: however they limit their scope to MBR with reference-based metrics, while we perform a more extensive evaluation over ranking methods and metrics. Amrhein and Sennrich (2022) also concurrently explored using MBR decoding with neural metrics, but with the purposes of identifying weaknesses in the metric (in their case COMET), similarly to the *metric overfitting* problem we discussed in §7.4.2. A comparison with N-best re-ranking was missing in these works, a gap this chapter fills. A related line of work is *minimum risk training* (MRT; (S. Shen et al. 2016; Smith & Eisner 2006)), which *trains* models to minimize risk, allowing arbitrary non-differentiable loss functions (Edunov et al., 2018; Wieting et al., 2019) and avoiding exposure bias (Kiegeland & Kreutzer, 2021; C. Wang & Sennrich, 2020). However, MRT is considerably more expensive and difficult to train and the gains are often small. Incorporating our quality metrics in MRT is an exciting research direction.

7.6 Conclusions & Subsequent Work

We leverage recent advances in MT quality estimation and evaluation and propose *quality-aware decoding* for NMT, effectively aligning MT with translation quality at decoding-time We explore different candidate generation and ranking methods, with a comprehensive empirical analysis across four datasets and two model classes. We show that, compared to MAP-based decoding, quality-aware decoding leads to better translations, according to powerful automatic evaluation metrics and human judgments.

Since the publication of this work (Fernandes, Farinhas, et al., 2022), the use of neural metrics as part of the MT pipeline has become popular technique to increase the final quality of translations. Follow-up works have studied how to improve the efficiency of MBR to make a more practical approach (Cheng & Vlachos, 2023), but neural metrics have also been used as feedback models to directly optimise the NMT parameters, particularly when using LLMs as MT models (Gulcehre et al., 2023; Ramos et al., 2024). However, these method still depend on neural metrics, which requires large amounts feedback and might suffer from problems of *reward overoptimization* (as we show in section 7.4.2). In the next chapter, we will show that multilingual LLMs can be used to provide *AI feedback* for MT, comparable to state-of-the-art learned metrics and providing more fine-grained and interpretable feedback.

Chapter 8

Fine-grained MT Evaluation with Large Language Models

As we showed, evaluation metrics provide can crucial to to improve model alignment. But while considerable progress has been made on estimating a single scalar quality score, current metrics lack the informativeness of more detailed schemes that annotate individual errors, such as Multidimensional Quality Metrics (MQM).

In this chapter, we help fill this gap by proposing AutoMQM, a prompting technique which leverages the reasoning and in-context learning capabilities of large language models (LLMs) and asks them to identify and categorize errors in translations. We start by evaluating recent LLMs, such as PaLM and PaLM-2, through simple score prediction prompting, and we study the impact of labeled data through in-context learning and finetuning. We then evaluate AutoMQM with PaLM-2 models, and we find that it improves performance compared to just prompting for scores (with particularly large gains for larger models) while providing interpretability through error spans that align with human annotations.

This chapter is based on Fernandes, Deutsch, et al. (2023).

8.1 Introduction

Evaluating natural language generation systems has always been challenging, and as the output quality of these systems has improved, evaluation has become even more challenging and critical. In Machine Translation, *learned* automatic metrics that leverage human-judgments to finetune language models (like the ones explored in previous chapter) currently represent the state-of-the-art in automatic evaluation benchmarks like the WMT Metrics task (Freitag et al., 2022), and show high correlation with human judgments. However, these metrics typically output a single, *uninterpretable* quality score, making it difficult to understand the type and extent of errors identified by them. The lack of insights makes it difficult for model developers to leverage these metrics to improve their systems.

Unlike automatic metrics that only provide a single scalar value as quality score, state-of-the-art human evaluation methodologies like Multidimensional Quality Metrics (MQM; Freitag, Foster, et al., 2021; Lommel et al., 2014) ask professional annotators to identify and label error spans with a category and severity. This much richer feedback can be used to gain a better understanding of the current limitations of the model under evaluation and improve it.

In this chapter, we ask whether large language models (LLMs) in combination with a few human annotations can be used to design an automatic metric that generates rich feedback similar to that generated by human experts in MQM. This chapter is motivated by recent papers that demonstrated that LLMs can be used as automatic metrics (Y. Liu et al., 2023) to generate a single quality score. In particular, Kocmi and Federmann (2023) showed that LLMs can be prompted to assess the quality of machine-generated translations, even achieving state-of-the-art performance on assessing system-level quality. However, previous work only provides a limited view of the capabilities of LLMs for machine translation evaluation: the focus has predominantly been on *score prediction* (i.e. predicting a numerical value for quality), without considering the use of *any* annotated data (either through in-context learning or finetuning), and only in *high-resource* language pairs.

We provide a large-scale study of the capabilities of LLMs (from the PaLM and PaLM-2 families; Anil et al., 2023; Chowdhery et al., 2022b) for machine translation evaluation (both with and without a reference translation), provide a novel comparison between prompting and finetuning. Inspired by findings that the performance of LLMs can be improved by prompting them for *rationales* of their predictions (Lu et al., 2023; Wei et al., 2022), we also propose **AutoMQM**, a prompting technique for MT evaluation that asks LLMs to identify error spans in a translation and to classify these errors according to the MQM framework, with a quality score derived automatically from the identified errors. A key advantage of AutoMQM is its *interpretability*, as users can inspect the errors responsible for a score (Figure 8.1).

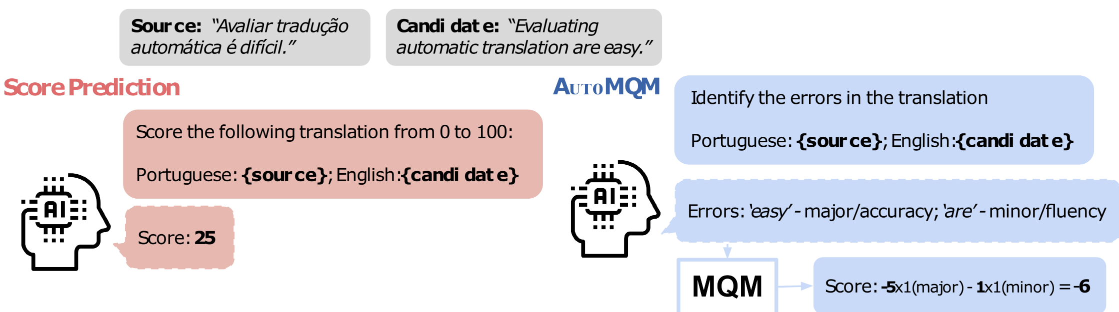

Figure 8.1: Illustration of how AutoMQM uses LLMs to assess the quality of a translation. Rather than asking for a single quality score, AutoMQM prompts models to identify and classify errors, and uses the MQM framework to produce a score.

Our contributions can be summarized as follows:

- We confirm the finding of (Kocmi & Federmann, 2023) that LLMs are *zero-shot* state-of-the-art system-level evaluators, but show low correlation with human judgment compared to *learned* metrics at the segment-level.
- We show that *finetuning* an LLM with human judgment mitigates its low segment-level performance (particularly for smaller LLMs), showing similar correlations with human judgment at both the system-level and segment-level to state-of-the-art learned metrics.
- We find that, with AutoMQM, PaLM-2 models can be prompted to generate rich MQM-like annotations, outperforming their score prediction counterparts at the segment-level.
- Furthermore, annotations predicted by PaLM-2 models correctly identify over 50% of words that are part of *major* errors, and are comparable to the ones produced by state-of-the-art *supervised* word-level evaluators.

Our findings might have significant implications for not only MT evaluation, but evaluation of machine-generated text in general, and further highlight the potential of using LLMs to provide *AI Feedback* (Fernandes, Madaan, et al., 2023).

The outputs of our models prompted with AutoMQM are available at github.com/google-research/google-research

8.2 Background: MT Evaluation

Machine translation evaluation is one of the most well-studied evaluation problems in NLP (Callison-Burch et al., 2008; Freitag et al., 2022). In this task, given

1. a *source* sentence in a (source) language
2. a *candidate* translation in a (target) language

an evaluation metric assesses the quality of the candidate translation by how well it conveys the meaning of the source sentence while considering other factors like *fluency*. Like many other natural language generation evaluation problems, this task is difficult because the set of correct translations for a given source sentence is often very large and not entirely known in advance. To simplify the problem of machine translation evaluation, often (3) a *reference* translation (typically created by a professional human translator) is included as additional information when assessing the candidate translation. This sub-problem is known as *reference-based* evaluation (as opposed *reference-less* evaluation or *quality estimation*).

Up until recently, human evaluation of machine translation was carried out predominantly with the aim of assigning a single quality score to a candidate translation. Consequently, *learned* metrics, which leverage collected human judgment data, are trained for and evaluated on the same task of *score prediction* (i.e., assigning a single quality score to a candidate translation), and can achieve high correlation with human-provided scores (Fre-

```
Based on the given source and reference, identify the major and minor errors in this
translation. Note that Major errors refer to actual translation or grammatical
    errors,
and Minor errors refer to smaller imperfections, and purely subjective opinions about
the translation.

{src_lang} source: "{source}"
{tgt_lang} human reference: "{reference}"
{tgt_lang} translation: "{candidate}"
Score (0-100): {score}
```

Figure 8.2: The *score prediction* prompt used in this chapter. (GEMBA-SQM; Kocmi and Federmann (2023)). Parts in purple are only included for *reference-based* evaluation, while parts in orange represent slots for outputs and are only included for in-context examples.

itag et al., 2022). However, framing machine translation evaluation as a score prediction task is problematic: any scoring or ranking of translations is implicitly based on an identification of errors in the candidate translations, and asking raters to solely provide a single score can lead to rushed and noisy judgments (Freitag, Foster, et al., 2021).

This insight has led to the adoption of the Multidimensional Quality Metrics (MQM) framework (Freitag, Foster, et al., 2021; Lommel et al., 2014) as the gold standard for evaluating machine translation. The MQM framework asks human evaluators to identify error spans in candidate translations and classify those errors according to various dimensions, e.g., *fluency*, *accuracy*, ... Importantly, the MQM framework *does not* ask annotators to provide a quality score for each translation, and instead derives one automatically from the identified error spans and their classifications. However, despite its richness, *most* automatic metrics that leverage MQM data only use the final quality score produced by the framework and discard the error span information and classification.

8.3 Using LLMs to Predict Quality Scores

Recent works have shown that large language models are versatile, general-purpose models that can be used to tackle many problems in NLP, including evaluation (Jain, Keshava, Sathyendra, et al., 2023; Kocmi & Federmann, 2023; Y. Liu et al., 2023). We begin by exploring how LLMs can be used for machine translation evaluation through *score prediction*.

8.3.1 Prompting

We start by measuring how far we can push the performance of LLMs with just *prompting* (P. Liu et al., 2023): by defining the task of MT evaluation and quality estimation as *textual templates* (with a general description of the problem and “slots” for the inputs and outputs), we can use general-purpose LLMs to perform these tasks at inference-time, without any parameter updates.

Throughout the chapter, we choose to use Kocmi and Federmann (2023)'s `GEMBA-SQM` prompt (Figure 8.2), which asks models to generate (a string representation of) a score from 0-100. We choose this prompt for two reasons: firstly, early explorations with various prompts showed that this generally performed well. Secondly, using a single prompt ensures a fairer comparison between the capabilities of different models.

In-Context Learning A surprising emergent capability of LLMs is their ability to improve on prompting-based tasks by including a very small amount of labeled data as part of the prompt/context (Brown et al., 2020a) and *without* parameter updates, a technique called *in-context learning* (ICL) or *few-shot prompting*. We thus investigate the impact that ICL has on LLMs' ability to assess translation quality. Recent works have shown that the impact of ICL is tightly tied with the exact examples included in the prompt, with a poor selection procedure leading to no improvements or even worse performance than the zero-shot case (Jain, Keshava, Sathyendra, et al., 2023). We therefore explore two sampling approaches to select in-context examples from a pre-defined "pool" of translation quality assessments: **uniform** and **stratified sampling**, where the example pool is bucketed by score ranges and examples are sampled from each bucket.

8.3.2 Finetuning

It has previously been shown that LLMs are capable of zero-shot evaluation (Kocmi & Federmann, 2023), but the extent to which *finetuning* on human judgment data can further boost the performance of LLMs has not been studied. In the WMT'22 Metrics Shared Task (Freitag et al., 2022), all top submissions were learned metrics; that is, pretrained models finetuned on human feedback.[1]

Thus, we investigate whether LLMs are amenable to finetuning on human judgment data. LLMs used in top-performing metrics are generally much larger than the pretrained language models leveraged by previous learned metrics (which generally have fewer than 1 billion parameters). Moreover, most learned metrics leverage pretrained encoder-only rather than (decoder-only) prefix language models. We experiment with finetuning LLMs using two objectives:

- *Regression* (**R**): Commonly used for training learned metrics (Rei, C. de Souza, et al., 2022), the objective here is a regression loss (e.g., mean squared error) between continuous scores obtained from the model (for example, with a *regression head*) and the human scores.

[1] While these metrics all leverage powerful pretrained (language) models, these generally aren't considered LLMs

- *Generative Classification* (**GC**): We bucket scores into discrete classes (e.g. *"bad"*, *"ok"* and *"good"*) and treat the MT evaluation task as a text-to-text classification problem (Raffel et al., 2020b) by having the model generate a template sentence with the class.

8.4 Using LLMs to Predict Error Spans

While producing quality scores that correlate with human judgments is an important part of translation quality assessment, metrics that solely do score prediction suffer from problems of **interpretability**: if a metric assigns a low score, the downstream users are left in the dark about which parts of the translation were responsible for the score and thus need to be corrected. This is especially problematic when the metric assigns a *wrong* score to a translation, as it is much harder to diagnose why the evaluation model made a mistake, and identify and prevent similar mistakes in the future. In fact, reducing translation quality to a single score has proven problematic even for human annotators: asking raters to solely provide a single score can lead to rushed and noisy judgments (Freitag, Foster, et al., 2021) and the current standard for translation quality evaluation involving human annotators is instead based on methodologies like the MQM framework (see §8.2) , which provide richer feedback by identifying error spans, categorizing them, and evaluating their severity.

Interestingly, another emergent phenomenon in LLMs is the success of *chain-of-thought* prompting (Wei et al., 2022): when defining a prompt for a particular task, if we instruct the model to produce a series of intermediate reasoning steps (*"let's think step-by-step"*), it tends to generate a free-text *rationale* before generating an output, and this often improves the performance on the task at hand (Y. Liu et al., 2023). Furthermore, this *chain-of-thought* prompting can be used to obtain *structured* rationales from LLMs, and this can lead to better performance than with free-text rationales (Lu et al., 2023).

Motivated by these findings, we propose **AutoMQM**, a prompting technique for translation quality assessment that instructs LLMs to *identify* errors in a translation, and *categorize* the type of error according to the MQM framework (Lommel et al., 2014). Furthermore, we *don't* ask the model to produce a score, as the MQM framework provides an algorithmic procedure to obtain one from identified errors: the total score is the sum of penalties for all errors identified, where (roughly) *major* errors get penalized with -5 and *minors* with -1 .[2] Figure 8.3 shows the main AutoMQM prompt used in this chapter.

Importantly, obtaining meaningful AutoMQM results in a zero-shot setting is a substantially more challenging task compared to score prediction: we found that, without any in-context examples, LLMs tend to produce outputs that are either uninformative or difficult to parse. Thus we only consider the AutoMQM task in the *few-shot* scenario. Based

[2] This is similar to methods that leverage external *executors* to improve LLMs (Gao, Madaan, et al., 2022).

```
Based on the given source and reference, identify the major and minor errors in this
translation. Note that Major errors refer to actual translation or grammatical
    errors,
and Minor errors refer to smaller imperfections, and purely subjective opinions about
the translation.

{src_lang} source: "{source}"
{tgt_lang} human reference: "{reference}"
{tgt_lang} translation: "{candidate}"
Errors: {error1:span} - {error1:severity}/{error1:category}; {error2:span} - ...
```

Figure 8.3: The *AutoMQM* prompt used in this chapter. Parts in purple are only included for *reference-based* evaluation, while parts in orange represent slots for outputs, and are only included for in-context examples.

on the findings from §8.5.2, we explore the impact of in-context learning by sampling from the example pool using stratified sampling extended with a set of *rejection criteria* (??), which ensures that the example set has a balance between major and minor errors as well as diversity in the categories of errors.

8.5 Experiments

8.5.1 Experimental Setup

Data The metrics in this work are evaluated on three *high-resource* high-resource language pairs come from the WMT'22 Metrics Shared Task (Freitag et al., 2022): en→de, zh→en, and en→ru. The ground-truth translation quality scores are derived from MQM ratings in which expert annotators marked error spans in the translations with different severity levels which are automatically converted to a numeric score (see §8.2). See Table 8.1 for statistics about the number of MT systems and segments for each language pair.

LP	#Sys	#Seg
en→de	13	1315
zh→en	14	1875
en→ru	15	1315

Table 8.1: The number of systems and segments that have MQM scores, used as ground-truth human feedback.

Additionally, in our experiments, AutoMQM required in-context examples with MQM annotations to work, so we restrict our evaluation of AutoMQM to en→de and zh→en because there are available MQM ratings from the WMT'21 Metrics Shared Task (Freitag, Rei, Mathur, Lo, Stewart, et al., 2021) that we can use as in-context learning examples.

Models We base most of our experiments on the following LLMs:

- **PaLM**: A 540 billion parameter autoregressive Transformer model trained on 780 billion tokens of high-quality text (Chowdhery et al., 2022b). It showed remarkable performance on a wide-range of NLP tasks, including MT (Vilar et al., 2022).

- **PaLM-2**: The successor to PaLM, the PaLM-2 family of LLMs (Anil et al., 2023) builds upon recent research insights, such as compute-optimal scaling, a more multilingual and diverse pre-training mixture, and architectural/optimization improvements. We mainly use two model sizes in the family: PaLM-2 Bison and (the larger) PaLM-2-Unicorn.[3] In addition we explore the impact of instruction-tuning by using a Unicorn model finetuned on the FLAN dataset (Wei et al., 2021).

For *score prediction*, we compare PaLM and PaLM-2 against the GPT family of LLMs (Brown et al., 2020a; OpenAI, 2023) by leveraging the results and outputs from the GEMBA evaluator (Kocmi & Federmann, 2023). We then evaluate the performance of AutoMQM with only PaLM-2 models (which performed best in score prediction).

Additionally, for the high-resource languages, we compare to a set of strong baseline evaluation metrics, MetricX-XXL and COMET-22, which were the two top-performing metrics in the WMT'22 Metrics Shared Task. MetricX-XXL and COMET-22 are both finetuned regression models trained on DA data from WMT that are initialized with mT5 (Xue et al., 2021) and XLM-R (Conneau et al., 2020), respectively.

For the AutoMQM experiments, we also compare against MaTESe, a comparable submission to the WMT'22 Metrics Shared task that finetuned a XLM-R model to identify major and minor errors, and computed a score automatically. Since we were unable to obtain the span-level predictions for the MaTESe submission, we also compare against the top submission to the WMT'22 Word-Level Quality Estimation Shared Task (Zerva et al., 2021b): word-level CometKiwi (COMET-WL) (Rei, Treviso, et al., 2022), also based on an XLM-R model trained on a combination of sentence- and word-level data. To do so, we re-run this model on the WMT'22 Metrics Shared Task data, and convert the predicted *word-level* `OK`/`BAD` tags into spans.[4]

Finetuning For *regression* finetuning, we use a real-valued logit, extracted from a fixed index in the first target token's logit vector, as the quality signal. (In particular, we leverage a special, *unused*, vocabulary token.) This was the technique used to train MetricX-XXL in the WMT 2022 Shared Task submission (Freitag et al., 2022). The regression-based model was trained on WMT direct assessment (DA) data from the years 2015 through 2020.

For *generative* classification, we bucket the scores in the training data into five classes, where class boundaries are assigned so that each class contains an equal number of training examples. We then map labels to verbal ratings from the following set, based on their bucket: [*"very bad"*, *"bad"*, *"ok"*, *"good"*, *"very good"*]. To evaluate the model, predictions are

[3]Information about exact number of parameters of PaLM-2 models is not publicly available.

[4]We consider a span as any maximal consecutive sequence of words marked as `BAD`, assigning every span the *major* severity.

mapped back to integer labels from 1 to 5. Any predictions not containing a substring in the label set are considered invalid and are mapped to 0. We experimented with finetuning on both DA and MQM 2020 (Freitag, Foster, et al., 2021) data, and found that the latter performed slightly better.

To assess the impact of *model size*, we also finetune two additional (smaller) PaLM-2 models, which we call S and M, comparing their finetuned and zero-shot performance.[5]

Metric Meta-Evaluation The quality of an automatic evaluation metric is estimated by comparing the agreement between the metric scores and ground-truth quality scores on a large number of translations from different MT systems, a process known as metric meta-evaluation. This work reports three different agreement scores, as follows.

The first is system-level accuracy, which calculates the percent of system pairs that are ranked the same by the metric and ground-truth scores, micro-averaged over a set of language pairs (Kocmi et al., 2021b). System-level scores are defined as the average score across all segments.

At the segment-level, the standard correlation that is reported by WMT is Kendall's τ. However, recent work pointed out problems with Kendall's τ with respect to ties (Deutsch et al., 2023). In short, different variants of τ are inconsistent with respect to ties and even biased against metrics that predict ties, as our metrics do in this work. Deutsch et al. (2023) recommend reporting a pairwise accuracy score, which rewards metrics for correctly ranking translations as well as correctly predicting ties, in combination with a tie calibration procedure that automatically introduces ties into metric scores so that the meta-evaluation is fairer. This accuracy score, denoted acc*, ranges between 0 and 1, and a random metric would achieve 33% accuracy. We report the "group-by-item" variant of the pairwise accuracy score from Deutsch et al. (2023) in addition to Pearson's ρ, a complementary signal to rank-based correlations that measure the strength of the linear relationship between two variables (and one of the standard correlations reported in WMT).

Span Meta-Evaluation Since AutoMQM provides not only scores but also the identified error spans, we can compare the predicted spans with the errors marked by annotators in the MQM annotations. We evaluate quality of predicted spans using: (1) *Span Precision* (SP), which measures the overlap of predicted spans and gold (annotated) spans; and (2) *Major recall* (MR), which captures the percentage of gold major errors that were predicted as errors (either minor or major).

More formally, consider the set of ground truth spans $S^\star$, where each span consists of a sequence of words, i.e., $s_i = (w_{(a)}, w_{(a+1)}, \cdots)$. Let $S^\star_{\text{maj}} \subseteq S^\star$ be the subset containing only the major errors. Given a span set S, we define its positional set $P(S)$ as the set

[5] We use a small variation of the *zero-shot* prompt, asking models for scores from the same 5 buckets used in finetuning.

containing the positions of all the words in every span in S. For example, assuming a span $s_i = (w_{(n)}, w_{(n+1)}, \cdots)$ in S starts at the nth position in the text, its corresponding positional set will include the positions $\{n, n+1, ..., n+\mathrm{len}(s_i)-1\}$. Then for a set of *predicted* spans $\hat{S}$, SP and MR are defined as:

$$\mathrm{SP}(\hat{S}) = \frac{|P(\hat{S}) \cap P(S^\star)|}{|P(\hat{S})|} \tag{8.1}$$

$$\mathrm{MR}(\hat{S}) = \frac{|P(\hat{S}) \cap P(S^\star_{\mathrm{maj}})|}{|P(S^\star_{\mathrm{maj}})|} \tag{8.2}$$

Intuitively, we care for overall precision (regardless of severity) since we want to make sure predicted errors tend to be marked by annotators as well, but for recall we care mostly for *major* errors, as these have a larger impact on translation quality and are more critical to identify. Additionally, we also report the (3) *Matthews Correlation Coefficient* (MCC), one of the official metrics in the word-level quality estimation tasks (Zerva et al., 2022).

8.5.2 Results

Score Prediction

Table 8.2 summarizes the meta-evaluation results, at the *system* and *segment* level, for both the *zero-shot prompting* and *finetuning* settings.

		System-Level	Segment-Level					
		All (3 LPs)	EN-DE		ZH-EN		EN-RU	
Model	Ref?	Accuracy	ρ	acc*	ρ	acc*	ρ	acc*
Baselines								
MetricX-XXL	✓	85.0%	0.549	61.1%	0.581	54.6%	0.495	60.6%
COMET-22	✓	83.9%	0.512	60.2%	0.585	54.1%	0.469	57.7%
COMET-QE	✗	78.1%	0.419	56.3%	0.505	48.8%	0.439	53.4%
Prompting								
PaLM 540B	✓	90.1%	0.247	55.4%	0.255	48.5%	0.180	48.6%
PaLM-2 Bison	✓	88.7%	0.394	56.8%	0.322	49.3%	0.322	52.8%
PaLM-2 Unicorn	✓	90.1%	0.401	56.3%	0.349	51.1%	0.352	55.3%
FLAN-PaLM-2 Unicorn	✓	75.9%	0.197	55.6%	0.139	46.1%	0.198	52.0%
PaLM 540B	✗	84.3%	0.239	56.1%	0.270	43.1%	0.300	51.8%
PaLM-2 Bison	✗	85.0%	0.355	57.0%	0.299	48.6%	0.303	53.1%
PaLM-2 Unicorn	✗	84.3%	0.275	56.1%	0.252	48.3%	0.209	49.8%
FLAN-PaLM-2 Unicorn	✗	69.7%	0.116	54.6%	0.112	43.8%	0.156	47.8%
Finetune								
PaLM-2 Bison (R)	✓	88.0%	0.511	61.0%	0.459	51.5%	0.458	59.5%
PaLM-2 Bison (GC)	✓	86.1%	0.400	59.2%	0.444	49.3%	0.365	56.0%
PaLM-2 Unicorn (R)	✓	87.6%	0.508	61.1%	0.412	52.6%	0.460	60.4%
PaLM 2 Bison (R)	✗	87.6%	0.490	59.9%	0.439	53.4%	0.437	59.2%
PaLM 2 Bison (GC)	✗	86.1%	0.368	57.5%	0.420	47.3%	0.390	54.9%
PaLM 2 Unicorn (GC)	✗	86.1%	0.407	57.9%	0.402	45.6%	0.411	55.3%

Table 8.2: Meta-evaluation results at system and segment-level for the *high-resource* language pairs. Finetuned **(R)** and **(GC)** represent the *regression* and *generative classification* objectives (§8.3.2). ✓and ✗ represent *reference-based* and *reference-less* metrics, respectively.

Prompting A first observation is almost all zero-shot LLM evaluators have higher *system-level* performance than learned metrics (with and without references), with PaLM 540B and PaLM-2 Unicorn achieving the best performance. At the segment level, the story is more complicated: similarly to Kocmi et al. (2022), we find that none of the LLMs we explored was able to consistently outperform the baseline learned metrics. We see that PaLM-540B is a particularly poor reference-based evaluator, which is surprising given its system-level performance. Unexpectedly, instruction-tuning with FLAN seems to *degrade* performance, with FLAN-PaLM-2 Unicorn achieving poor performance at both the system and segment levels.[6]

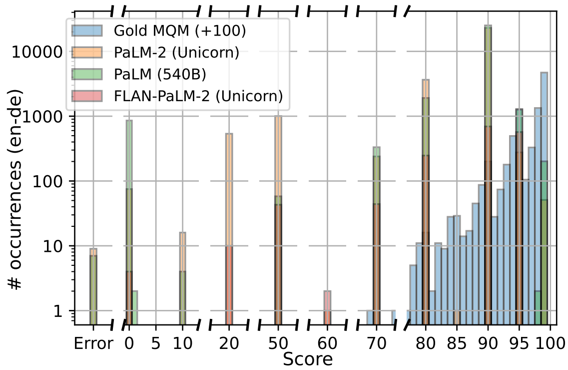

Figure 8.4: Distribution of scores for various LLM *reference-based* evaluators, on the EN-DE test set. Note that the y axis is in *log-scale*.

Nevertheless, PaLM-2 models achieve high correlations with human judgments, and the *reference-less* PaLM-2 Bison is competitive with the *learned* baselines, particularly at assessing alternative translations of the same sentence (acc*). Figure 8.4 shows the distribution of scores produced by PaLM- and PaLM-2-based evaluators. We find that, despite being prompted to give a score in the 0-100 range, these models almost always output one of a very limited set of scores (e.g. 0, 50, 90, 95). Given Kocmi and Federmann (2023)'s similar findings with GPT models, it seems that this is a consequence of the pretraining objective and the overall prevalence of certain *"round"* scores on the training data.

Finetuning Despite their already-great performance in the zero-shot setting, we find that finetuning LLMs can further improve LLM evaluators' segment-level scores. This is particularly obvious for the *reference-less* evaluators, where a finetuned PaLM-2 Bison achieves state-of-the-art performance in segment-level correlations and comparable system-level accuracy across all language pairs. Moreover, when we look at how performance *scales* with parameter count (Figure 8.5), we observe an interesting trend: while smaller models are not capable of being effective zero-shot evaluators, finetuning them leads to competitive

[6]Note that this might be a problem with the FLAN dataset and not instruction-tuning in general, as the GPT models are also instruction-tuned and perform well.

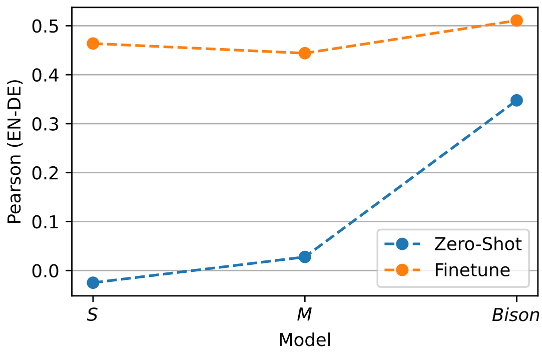

Figure 8.5: Behavior of *Pearson* as we scale the LLM's parameter count. Note that the x axis is not to-scale with regard to parameter count.

performance, and only a slight decrease when compared to their larger finetuned counterparts.

In-context Learning Figure 8.6 shows the mean and interquartile range (IQR) of the performance as we increase the number of in-context examples k (with 100 example sets per k) sampled with *stratified* sampling Surprisingly, despite evidence of the benefits of in-context learning for many tasks, we found that including in-context examples during evaluation (almost) never led to better performance, either with *uniform* or *stratified* sampling.

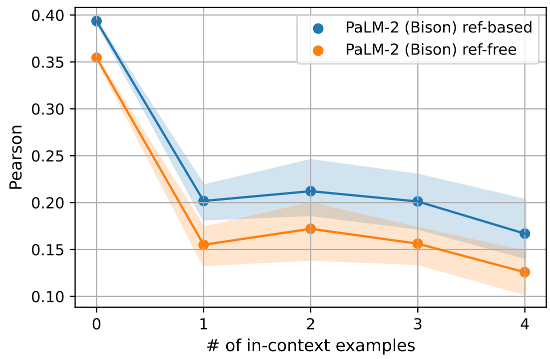

Figure 8.6: Mean *Pearson* and its interquartile range (IQR) in the WMT22 EN-DE test set, as we increase the number of in-context examples with *stratified* sampling

AutoMQM

Figure 8.7 shows the mean and interquartile range (IQR) of the performance of PaLM-2 Bison with AutoMQM, as we increase the number of in-context examples (again, with 100 example sets per k). Contrary to the performance with score prediction, we find that performance with AutoMQM seems to (mostly) scale with the number of in-context examples: performance increases monotonically with up to 4 in-context examples and plateaus thereafter. Additionally, the variance across the in-context learning sets seems

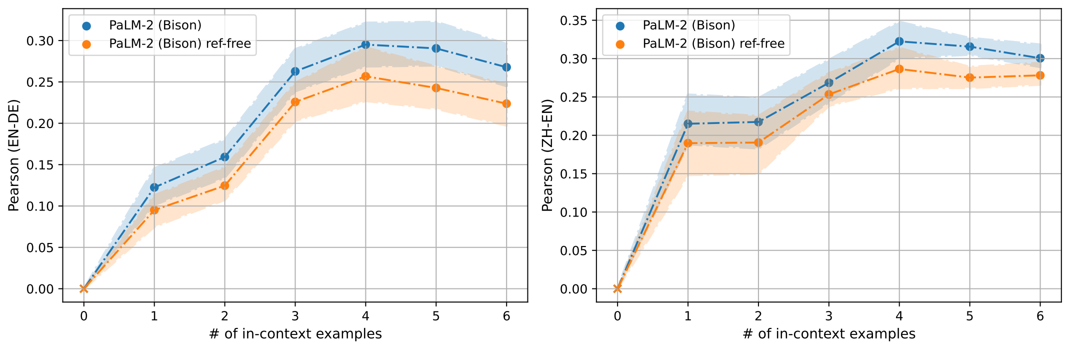

Figure 8.7: Mean *Pearson* and its interquartile range (IQR), as we increase the number of in-context examples in the AutoMQM prompt, for EN-DE (left) and ZH-EN (right).

to be lower, with most example sets exhibiting less than 0.05 *Pearson* difference from the best-performing sets. All this suggests that LLM evaluators are much more robust to the choice of in-context examples when prompted for AutoMQM rather than for score prediction. We also find that the behavior of in-context learning is similar for reference-based and reference-less evaluation.

		System-Level	Segment-Level			
		All (2 LPs)	EN-DE		ZH-EN	
Model	Ref?	Accuracy	ρ	acc*	ρ	acc*
Baselines						
MetricX-XXL	✓	81.1%	0.549	61.1%	0.581	54.6%
MaTESe	✓	79.9%	0.391	58.8%	0.528	51.5%
COMET-QE	✗	76.9%	0.419	56.3%	0.505	48.8%
MaTESe-QE	✗	73.4%	0.298	57.9%	0.468	50.1%
COMET-WL	✗	71.6%	0.418	57.1%	0.406	51.5%
Score Prediction						
PaLM-2 Bison	✓	86.4%	0.394	56.8%	0.322	49.3%
PaLM-2 Unicorn	✓	86.4%	0.401	56.3%	0.349	51.1%
PaLM-2 Bison	✗	84.0%	0.355	57.0%	0.299	48.6%
PaLM-2 Unicorn	✗	80.5%	0.275	56.1%	0.252	48.3%
AutoMQM						
PaLM-2 Bison	✓	84.0%	0.369	59.2%	0.355	48.4%
PaLM-2 Unicorn	✓	87.6%	0.432	59.1%	0.442	51.8%
PaLM 2 Bison	✗	87.6%	0.297	55.2%	0.331	48.0%
PaLM 2 Unicorn	✗	83.4%	0.368	56.4%	0.429	50.2%

Table 8.3: Meta-evaluation results for PaLM-2 models using *AutoMQM* and score prediction, at the system and segment levels for multiple language pairs.

Table 8.3 shows the meta-evaluation results for PaLM-2 Bison and Unicorn prompted with AutoMQM (using the best-performing in-context learning sets in Figure 8.7). For ease of comparison, we also report their performance when prompted for *score prediction*, as well as the performance of the baselines. Overall, prompting LLMs with AutoMQM seems to lead to significant improvements in evaluating machine translation quality, particularly for larger models: Unicorn achieves better performance (across all meta evaluations) with it than when prompted for *score prediction*, and its reference-less version is competitive with the best learned metric even at the segment level. However, for the smaller Bison, the benefits of AutoMQM are less clear, with both techniques performing comparably.

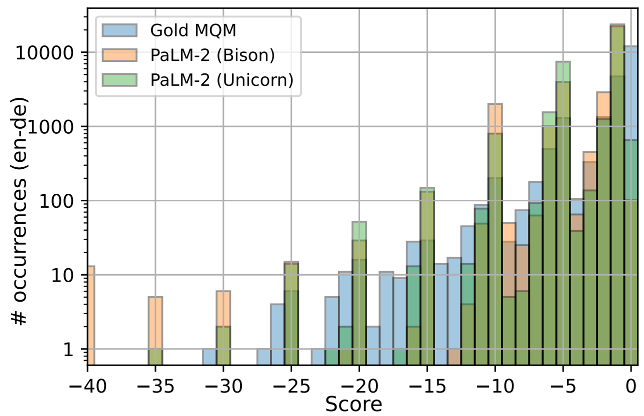

Figure 8.8: Distribution of scores for PaLM-2 models using AutoMQM, on the WMT22 EN-DE test set

This hints that *scale* is necessary for *zero*- and *few*- shot fine-grained evaluation (like with AutoMQM). We also find that the *distribution* of scores produced by LLMs prompted with AutoMQM is much closer to the gold MQM distribution, with models outputting a much larger set of scores, and in the same ranges as annotators do (see Figure 8.8).

		EN-DE			ZH-EN		
Model	R?	SP	MR	MCC	SP	MR	MCC
Baselines							
COMET-WL	✗	0.267	0.250	0.161	0.364	0.178	0.152
AutoMQM							
Bison	✓	0.095	0.749	0.060	0.252	0.255	0.109
Unicorn	✓	0.175	0.628	0.193	0.238	0.476	0.143
Bison	✗	0.119	0.520	0.092	0.224	0.311	0.091
Unicorn	✗	0.150	0.580	0.150	0.229	0.488	0.133

Table 8.4: Span-level meta-evaluation on WMT22 for PaLM-2 models using *AutoMQM*. **SR** and **MR** represent *span precision* and *major recall*, respectively.

Finally, when evaluating the error spans produced by LLMs prompted with AutoMQM (Table 8.4), we find that PaLM-2 models are able to identify most of the *major* errors. However, it does seem to *over-predict* errors (with errors predicted by Unicorn having on average ∼5 words per span vs ∼2 words in the ground truth) and have overall low span precision. Similarly to overall *score* correlations, *scale* also seems to be important for the quality of spans produced by AutoMQM, with Unicorn outperforming Bison at most metrics. Additionally, Unicorn prompted with AutoMQM predicts spans of comparable quality to the ones produced by current state-of-the-art *learned* word-level evaluators (trained on a large number of fine-grained annotations derived from MQM): while word-level models are more precise, their overall span correlation (MCC) is comparable, and they miss considerably more *major* errors than LLMs (despite only leveraging a handful of annotations).

8.6 Related Work

The success of *learned* machine translation metrics (Freitag et al., 2022; Qin et al., 2022; Rei, C. de Souza, et al., 2022; Sellam et al., 2020), which finetune neural network models pretrained on large amounts of (unsupervised) data, highlighted the importance of leveraging *transfer learning* to achieve metrics with better correlation with human judgments. More recently, *generative* LLMs (Anil et al., 2023; OpenAI, 2023) have consistently demonstrated impressive results in natural language understanding and *zero-* and *few-shot* transfer and, naturally, interest in employing these models for (translation) evaluation has increased. Kocmi and Federmann (2023) first explored the use of GPT models for evaluating machine translation tasks, showing their potential as *zero-shot* evaluators, and others have since extended GPT-based evaluation to other generation problems (Jain, Keshava, Sathyendra, et al., 2023; Y. Liu et al., 2023).

Perrella et al. (2022) first highlighted that MQM annotations could be leveraged to allow pretrained models to predict major and minor errors and, similarly to AutoMQM, used the identified errors to automatically score translations. However, their approach relied on weaker encoder-only or encoder-decoder language models, required *supervised* data to work, and overall underperformed other top metrics. We compare against their *MaTASe* metric in our experiments. Lu et al. (2023) showed that doing *error analysis*, a prompting technique similar to AutoMQM, could lead to better ChatGPT-based evaluators. However, they still relied on the LLM to provide a score once it identified errors (rather than do it automatically using something like the MQM framework). Furthermore, they provided a very limited meta-evaluation using only 40 examples per language pair. Concurrently with our work, Xu, Wang, et al. (2023) proposed InstructScore, a LLaMA-based evaluator that asks models to identify and categorize errors in translation (as well as providing a natural language explanation for each error). However, the authors only explore a 7B parameter model and don't leverage zero- and few-shot capabilities of models as in this work. Instead, they rely on a more complex approach of distilling the knowledge of a more capable GPT-4 LLM.

Additionally, WMT Word-Level Quality Estimation shared tasks (Fonseca et al., 2019; Zerva et al., 2022) leverage MQM data by converting span-level annotations of errors (normally of *major* severity) to word-level tags and Task 2 in the WMT19 Quality Estimation shared task evaluation explicitly evaluated submissions of span-level annotations (although most submissions still consisted of models that predicted word-level tags which were converted to spans). We also compare against state-of-the-art word-level quality estimation models.

8.7 Conclusion & Subsequent Work

In this study, we have systematically investigated the capabilities of large language models for machine translation evaluation through *score prediction*, and proposed AutoMQM, a novel prompting technique that leverages the Multidimensional Quality Metrics (MQM) framework for interpretable MT evaluation using LLMs. We demonstrated that just prompting LLMs for score prediction leads to state-of-the-art system-level evaluators, but still falls short of the best *learned* metrics at the segment-level (with finetuning being necessary to close this gap). Then we showed that AutoMQM can further improve the performance of LLMs without finetuning while providing interpretability through error spans that align with human annotations.

Our findings surrounding finetuning LLMs for *score prediction* hint that LLMs' performance in machine translation evaluation could be further improved by finetuning these models on fine-grained human judgment data (like MQM). In fact, subsequent work has since shown that training evaluation models on fine-grained MQM can lead to state-of-the-art metrics that surpass ones trained only on sentence-level scores (Guerreiro et al., 2023), and that fine-grained feedback produced by these models can be used to improve the quality of translations produced by LLMs (Xu, Deutsch, et al., 2023).

Epilogue

Chapter 9

Conclusion

In this chapter, we provide a high-level summary of our contributions, highlight some open problems & potential directions for future research, and offer some concluding thoughts.

9.1 Summary

This thesis investigates three fundamental challenges that have emerged alongside the remarkable success of large-scale neural language models: understanding their global behavior, extracting meaningful explanations for their outputs, and aligning their decisions with human intentions. We focused primarily and extensively on machine translation as our testbed, which historically has served as a breeding ground for broader innovations in natural language processing. However, we occasionally explored other tasks, and we believe the insights and methods developed throughout this work have wider implications for the development of reliable, interpretable, and effective language technologies.

In Part I, we studied the global behavioral patterns in neural machine translation models. We approached this challenge from two complementary perspectives: scaling dynamics and information utilization. In Chapter 3, we provided a comprehensive analysis of how multilingual machine translation models scale across different language pairs, discovering that individual task performance follows predictable patterns that are invariant to cross-lingual interactions. This insight enabled us to propose joint scaling laws that can predict the full Pareto frontier of translation quality as models grow in capacity. In Chapter 4 we shifted our focus to document-level translation, investigating how much contextual information beyond sentence boundaries these models actually leverage. Through the development of information-theoretic metrics, we discovered that models exhibit a preference for target-language context over source-language context, and proposed methods to enhance the utilization of extra-sentential information.

In Part II, we explored methods for extracting interpretable, local explanations from neural network decisions. Rather than relying on intuitive but potentially unreliable notions of explainability, we grounded our approach in a quantitatively measurable definition: good explanations should help other models learn. In Chapter 5, we introduced a novel framework leveraging bilevel optimization and meta-learning to train parametrized explainers that maximize their teaching effectiveness. Our approach demonstrated superior simulability compared to existing methods while producing explanations that align more closely with human intuitions about model behavior. In Chapter 6 we showed that *frontier* reasoning models optimized to solve complex tasks are also state-of-the-art low-resource MT systems, and that their *chain-of-thought* (CoT) rationales are good explanations of this task as they help student models learn more effectively than without explanations, but only through *post-hoc* regularization (rather than the standard CoT distillation).

Finally, in Part III we tackled the challenge of ensuring that machine translation is aligned with the goal of human-like translation. We began in Chapter 7 by developing quality-aware decoding algorithms that incorporate neural metrics of translation quality directly into the inference process. Through techniques such as N-best reranking and minimum Bayes risk decoding, we demonstrated that models can be steered toward more human-like translations without requiring retraining. Subsequently, in Chapter 8 we showed that large language models can serve as effective fine-grained translation quality metrics with minimal supervision, requiring orders of magnitude less training data than traditional neural metrics while achieving state-of-the-art performance. This development opens new possibilities for leveraging LLMs in the alignment of machine translation systems. Throughout these investigations, we released a series of research artifacts for public use, including datasets, code implementations, and trained models, contributing to the broader research community's ability to build upon and extend our work.

9.2 Future Directions

The research presented in this thesis spans a period of rapid transformation in natural language processing, from the era of specialized (machine translation) models to the current dominance of large language models. Although much of our foundational work was conducted in the pre-LLM era, we believe and have evidence that our findings extend to this new age (Flemings et al., 2025; I. Wu et al., 2025), and there are still innumerable opportunities to explore and extend our ideas. Below, we highlight some particularly promising research directions.

Understanding Multi-task Scaling in LLMs Current *state-of-the-art* language models' training can be framed as a massive multitask optimization problem (both in *pretraining* and *posttraining*). Therefore, understanding if our findings in Chapter 3 surrounding *interference* and *effective capacity* generalize to this setting could have large implications in guiding future trainings of new LLMs: for example, if models share very little capacity across languages, as we reach the limits of parameter scaling, it might turn out that it is more feasible to move towards *monolingual* LLMs. Our findings might also explain the move towards *mixture-of-experts* models (Cai et al., 2025) that allow scaling the number of *total* parameters while keeping the computation fixed.

Optimizing *Chains-of-Thoughts* for Teaching In Chapter 6 we saw that optimizing *reasoning* models for task performance alone seems to naturally lead to *chains-of-thoughts* that are good at teaching. However, how to best optimize these explanations for this later goal is still an open problem: while bilevel optimization approaches such as SMaT (Chapter 6) might be intractable with current very large models, *reinforcement learning* could be a viable alternative (Cetin et al., 2025). This could lead not only to more efficient distillation of student models, but also inadvertently to better, more trustworthy *frontier* reasoning models. Studying also how these explanations help *humans* predict the model outputs (Arora et al., 2022) or even learn the task itself (for example, teaching new language *à la* Chapter 6) could further cement their usefulness outside of distillation itself.

Beyond Human Feedback: *Extrinsically-Aware* Machine Translation Our work with quality-aware decoding showed that it is possible to optimize machine translation models for (models of) human feedback and translation quality. However, for more complex open-ended tasks (such as *document-level* machine translation), trying to capture quality as a single *intrinsic* score might be too reductive. In recent work (Fernandes et al., 2025) we show that *extrinsically* evaluating translations by how well they enable downstream tasks, such as question-answering, is a promising direction (and that we can use LLMs to automate this) but how to optimize models for this type of evaluation (either in training or in decoding) is still an open problem.

9.3 Concluding Thoughts

Throughout this thesis, it was possible to feel the interconnected nature of the challenges we addressed. Being able to explain models' decisions inexorably helps us understand their behaviour and, in turn, both helps align and improve their performance. Our work thus offers practical tools and insights to advance large-scale, neural network-based machine translation models (LLM-based or not). The focus on machine translation throughout this thesis was also not merely a matter of convenience or specialization. Translation represents a particularly rich domain for studying these challenges and an important problem as it sits at the intersection of multiple languages, cultures, and communication needs. Thus, we believe our research was meaningful if it helps, even if in the slightest, the millions of users who depend on these systems for meaningful communication.

Bibliography

This bibliography contains 308 references.

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., & Kim, B. (2018). Sanity checks for saliency maps. *Advances in neural information processing systems*, *31*.

Agrawal, S., De Souza, J. G. C., Rei, R., Farinhas, A., Faria, G., Fernandes, P., Guerreiro, N. M., & Martins, A. (2024, November). Modeling user preferences with automatic metrics: Creating a high-quality preference dataset for machine translation. In Y. Al-Onaizan, M. Bansal, & Y.-N. Chen (Eds.), *Proceedings of the 2024 conference on empirical methods in natural language processing* (pp. 14503–14519). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.emnlp-main.803

Agrawal, S., Farajian, A., Fernandes, P., Rei, R., & Martins, A. F. T. (2024). Assessing the role of context in chat translation evaluation: Is context helpful and under what conditions? *Transactions of the Association for Computational Linguistics*, *12*, 1250–1267. https://doi.org/10.1162/tacl_a_00700

Akyürek, A. F., Akyürek, E., Madaan, A., Kalyan, A., Clark, P., Wijaya, D., & Tandon, N. (2023). Rl4f: Generating natural language feedback with reinforcement learning for repairing model outputs.

Alves, D. M., Pombal, J., Guerreiro, N. M., Martins, P. H., Alves, J., Farajian, A., Peters, B., Rei, R., Fernandes, P., Agrawal, S., Colombo, P., de Souza, J. G. C., & Martins, A. F. T. (2024). Tower: An open multilingual large language model for translation-related tasks. *Conference on Language Modeling COLM*. https://openreview.net/forum?id=EHPns3hVkj

Amodei, D., Olah, C., Steinhardt, J., Christiano, P. F., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. *CoRR*, *abs/1606.06565*. http://arxiv.org/abs/1606.06565

Amrhein, C., & Sennrich, R. (2022). Identifying weaknesses in machine translation metrics through minimum bayes risk decoding: A case study for comet. https://doi.org/10.48550/ARXIV.2202.05148

Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., Chu, E., Clark, J. H., Shafey, L. E., Huang, Y., Meier-Hellstern, K., Mishra, G., Moreira, E., Omernick, M., Robinson, K., … Wu, Y. (2023). Palm 2 technical report.

Arivazhagan, N., Bapna, A., Firat, O., Lepikhin, D., Johnson, M., Krikun, M., Chen, M. X., Cao, Y., Foster, G. F., Cherry, C., Macherey, W., Chen, Z., & Wu, Y. (2019a). Massively multilingual neural machine translation in the wild: Findings and challenges. *ArXiv*, *abs/1907.05019*.

Arivazhagan, N., Bapna, A., Firat, O., Lepikhin, D., Johnson, M., Krikun, M., Chen, M. X., Cao, Y., Foster, G. F., Cherry, C., Macherey, W., Chen, Z., & Wu, Y. (2019b). Massively multilingual neural machine translation in the wild: Findings and challenges. *CoRR*, *abs/1907.05019*. http://arxiv.org/abs/1907.05019

Arora, S., Pruthi, D., Sadeh, N., Cohen, W., Lipton, Z., & Neubig, G. (2022). Explain, edit, and understand: Rethinking user study design for evaluating model explanations. *Thirty-Sixth AAAI Conference on Artificial Intelligence (AAAI)*. https://arxiv.org/abs/2112.09669

Arras, L., Montavon, G., Müller, K.-R., & Samek, W. (2017). Explaining recurrent neural network predictions in sentiment analysis. *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, 159–168. https://doi.org/10.18653/v1/W17-5221

Atanasova, P., Simonsen, J. G., Lioma, C., & Augenstein, I. (2020). A diagnostic study of explainability techniques for text classification. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3256–3274. https://doi.org/10.18653/v1/2020.emnlp-main.263

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, *10*.

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Y. Bengio & Y. LeCun (Eds.), *3rd international conference on learning representations, ICLR 2015, san diego, ca, usa, may 7-9, 2015, conference track proceedings*. http://arxiv.org/abs/1409.0473

Bai, Y., Jones, A., Ndousse, K., Askell, A., Chen, A., DasSarma, N., Drain, D., Fort, S., Ganguli, D., Henighan, T., Joseph, N., Kadavath, S., Kernion, J., Conerly, T., El-Showk, S., Elhage, N., Hatfield-Dodds, Z., Hernandez, D., Hume, T., . . . Kaplan, J. (2022). Training a helpful and harmless assistant with reinforcement learning from human feedback.

Bai, Y., Kadavath, S., Kundu, S., Askell, A., Kernion, J., Jones, A., Chen, A., Goldie, A., Mirhoseini, A., McKinnon, C., Chen, C., Olsson, C., Olah, C., Hernandez, D., Drain, D., Ganguli, D., Li, D., Tran-Johnson, E., Perez, E., . . . Kaplan, J. (2022). Constitutional ai: Harmlessness from ai feedback.

Bansal, Y., Ghorbani, B., Garg, A., Zhang, B., Krikun, M., Cherry, C., Neyshabur, B., & Firat, O. (2022). Data scaling laws in nmt: The effect of noise and architecture. *ICML*.

Bapna, A., Caswell, I., Kreutzer, J., Firat, O., van Esch, D., Siddhant, A., Niu, M., Baljekar, P., Garcia, X., Macherey, W., Breiner, T., Axelrod, V., Riesa, J., Cao, Y., Chen, M. X., Macherey, K., Krikun, M., Wang, P., Gutkin, A., … Hughes, M. (2022). Building machine translation systems for the next thousand languages. https://doi.org/10.48550/ARXIV.2205.03983

Barrault, L., Bojar, O., Costa-jussà, M. R., Federmann, C., Fishel, M., Graham, Y., Haddow, B., Huck, M., Koehn, P., Malmasi, S., Monz, C., Müller, M., Pal, S., Post, M., & Zampieri, M. (2019). Findings of the 2019 conference on machine translation (WMT19). *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 1–61. https://doi.org/10.18653/v1/W19-5301

Bastings, J., Aziz, W., & Titov, I. (2019). Interpretable neural predictions with differentiable binary variables. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2963–2977. https://doi.org/10.18653/v1/P19-1284

Bastings, J., Ebert, S., Zablotskaia, P., Sandholm, A., & Filippova, K. (2021). "will you find these shortcuts?" A protocol for evaluating the faithfulness of input salience methods for text classification. *CoRR, abs/2111.07367*. https://arxiv.org/abs/2111.07367

Bawden, R., Sennrich, R., Birch, A., & Haddow, B. (2018). Evaluating discourse phenomena in neural machine translation. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1304–1313. https://doi.org/10.18653/v1/N18-1118

Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, 610–623. https://doi.org/10.1145/3442188.3445922

Bengio, Y., Ducharme, R., Vincent, P., & Janvin, C. (2003). A neural probabilistic language model. *J. Mach. Learn. Res.*, *3*(null), 1137–1155.

Bhattacharyya, S., Rooshenas, A., Naskar, S., Sun, S., Iyyer, M., & McCallum, A. (2021). Energy-based reranking: Improving neural machine translation using energy-based models. *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)*, 4528–4537. https://doi.org/10.18653/v1/2021.acl-long.349

Blum, C., Filippova, K., Yuan, A., Ghandeharioun, A., Zimmert, J., Zhang, F., Hoffmann, J., Linzen, T., Wattenberg, M., Dixon, L., & Geva, M. (2025). Beyond the rosetta stone: Unification forces in generalization dynamics. https://arxiv.org/abs/2508.11017

Böhm, F., Gao, Y., Meyer, C. M., Shapira, O., Dagan, I., & Gurevych, I. (2019). Better rewards yield better summaries: Learning to summarise without references. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* *and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3110–3120. https://doi.org/10.18653/v1/D19-1307

Boizard, N., Gisserot-Boukhlef, H., Alves, D. M., Martins, A., Hammal, A., Corro, C., Hudelot, C., Malherbe, E., Malaboeuf, E., Jourdan, F., Hautreux, G., Alves, J., El-Haddad, K., Faysse, M., Peyrard, M., Guerreiro, N. M., Fernandes, P., Rei, R., & Colombo, P. (2025). Eurobert: Scaling multilingual encoders for european languages. *Proceedings of the Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2503.05500

Bostrom, N. (2014). *Superintelligence: Paths, dangers, strategies* (1st). Oxford University Press, Inc.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press. https://doi.org/10.1017/CBO9780511804441

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., & Zhang, Q. (2018). JAX: Composable transformations of Python+NumPy programs. http://github.com/google/jax

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020a). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 1877–1901, Vol. 33). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020b). Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in neural information processing systems* (pp. 1877–1901, Vol. 33). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

Bugliarello, E., Mielke, S. J., Anastasopoulos, A., Cotterell, R., & Okazaki, N. (2020). It's easier to translate out of English than into it: Measuring neural translation difficulty by cross-mutual information. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 1640–1649. https://doi.org/10.18653/v1/2020.acl-main.149

Cai, W., Jiang, J., Wang, F., Tang, J., Kim, S., & Huang, J. (2025). A survey on mixture of experts in large language models. *IEEE Transactions on Knowledge and Data Engineering*, *37*(7), 3896–3915. https://doi.org/10.1109/TKDE.2025.3554028

Callison-Burch, C., Koehn, P., Monz, C., Schroeder, J., & Fordyce, C. S. (Eds.). (2008, June). *Proceedings of the third workshop on statistical machine translation*. Association for Computational Linguistics. https://aclanthology.org/W08-0300

Caswell, I., Nielsen, E., Luo, J., Cherry, C., Kovacs, G., Shemtov, H., Talukdar, P., Tewari, D., Diane, B. M., Doumbouya, K. M., Diane, D., & Cissé, S. F. (2025). Smol: Professionally translated parallel data for 115 under-represented languages. https://arxiv.org/abs/2502.12301

Cetin, E., Zhao, T., & Tang, Y. (2025). Reinforcement learning teachers of test time scaling. https://arxiv.org/abs/2506.08388

Cettolo, M., Girardi, C., & Federico, M. (2012). WIT3: Web inventory of transcribed and translated talks. *Proceedings of the 16th Annual conference of the European Association for Machine Translation*, 261–268. https://www.aclweb.org/anthology/2012.eamt-1.60

Chandrasekaran, A., Prabhu, V., Yadav, D., Chattopadhyay, P., & Parikh, D. (2018). Do explanations make VQA models more predictable to a human? *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1036–1042. https://doi.org/10.18653/v1/D18-1128

Chen, J., Song, L., Wainwright, M., & Jordan, M. (2018). Learning to explain: An information-theoretic perspective on model interpretation. *International Conference on Machine Learning*, 883–892.

Chen, J., Li, X., Zhang, J., Zhou, C., Cui, J., Wang, B., & Su, J. (2020). Modeling discourse structure for document-level neural machine translation. *Proceedings of the First Workshop on Automatic Simultaneous Translation*, 30–36. https://doi.org/10.18653/v1/2020.autosimtrans-1.5

Chen, L., Ma, S., Zhang, D., Wei, F., & Chang, B. (2023). On the pareto front of multilingual neural machine translation. *ArXiv*, *abs/2304.03216*. https://api.semanticscholar.org/CorpusID:257985324

Cheng, J., & Vlachos, A. (2023, December). Faster minimum Bayes risk decoding with confidence-based pruning. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 12473–12480). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.767

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014, October). Learning phrase representations using RNN encoder–decoder for statistical machine translation. In A. Moschitti, B. Pang, & W. Daelemans (Eds.), *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1724–1734). Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1179

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., . . . Fiedel,

N. (2022a). Palm: Scaling language modeling with pathways. https://doi.org/10.48550/ARXIV.2204.02311

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., … Fiedel, N. (2022b). Palm: Scaling language modeling with pathways.

Chung, H. W., Fevry, T., Tsai, H., Johnson, M., & Ruder, S. (2021). Rethinking embedding coupling in pre-trained language models. *Tenth International Conference on Learning Representations, ICLR*. https://openreview.net/forum?id=xpFFI_NtgpW

Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020). ELECTRA: Pre-training text encoders as discriminators rather than generators. *ICLR*. https://openreview.net/pdf?id=r1xMH1BtvB

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8440–8451. https://doi.org/10.18653/v1/2020.acl-main.747

Costa-jussà, M. R., Cross, J., Çelebi, O., Elbayad, M., Heafield, K., Heffernan, K., Kalbassi, E., Lam, J., Licht, D., Maillard, J., et al. (2022). No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Dabre, R., Cromieres, F., & Kurohashi, S. (2017). Enabling multi-source neural machine translation by concatenating source sentences in multiple languages. *Proceedings of Machine Translation Summit XVI: Research Track*, 96–107. https://aclanthology.org/2017.mtsummit-papers.8

DeepSeek-AI. (2025). Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. https://arxiv.org/abs/2501.12948

Denil, M., Demiraj, A., & de Freitas, N. (2014). Extraction of salient sentences from labelled documents. *ArXiv, abs/1412.6815*. https://api.semanticscholar.org/CorpusID:9121062

Dery, L. M., Michel, P., Talwalkar, A. S., & Neubig, G. (2021). Should we be pre-training? an argument for end-task aware training as an alternative. *ArXiv, abs/2109.07437*.

Deutsch, D., Briakou, E., Caswell, I., Finkelstein, M., Galor, R., Juraska, J., Kovacs, G., Lui, A., Rei, R., Riesa, J., Rijhwani, S., Riley, P., Salesky, E., Trabelsi, F., Winkler, S., Zhang, B., & Freitag, M. (2025). WMT24++: Expanding the Language Coverage of WMT24 to 55 Languages And Dialects. https://arxiv.org/abs/2502.12404

Deutsch, D., Foster, G., & Freitag, M. (2023, December). Ties matter: Meta-evaluating modern metrics with pairwise accuracy and tie calibration. In H. Bouamor, J. Pino,

& K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 12914–12929). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.798

DeYoung, J., Jain, S., Rajani, N. F., Lehman, E., Xiong, C., Socher, R., & Wallace, B. C. (2020, July). ERASER: A benchmark to evaluate rationalized NLP models. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 4443–4458). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.408

Ding, S., Xu, H., & Koehn, P. (2019). Saliency-driven word alignment interpretation for neural machine translation. *Proceedings of the Fourth Conference on Machine Translation (Volume 1: Research Papers)*, 1–12. https://doi.org/10.18653/v1/W19-5201

Dong, D., Wu, H., He, W., Yu, D., & Wang, H. (2015). Multi-task learning for multiple language translation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1723–1732.

Doshi-Velez, F., & Kim, B. (2017). Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Duh, K., & Kirchhoff, K. (2008). Beyond log-linear models: Boosted minimum error rate training for n-best re-ranking. *Proceedings of ACL-08: HLT, Short Papers*, 37–40. https://aclanthology.org/P08-2010

Edunov, S., Ott, M., Auli, M., Grangier, D., & Ranzato, M. (2018). Classical structured prediction losses for sequence to sequence learning. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 355–364. https://doi.org/10.18653/v1/N18-1033

Eikema, B., & Aziz, W. (2020). Is MAP decoding all you need? the inadequacy of the mode in neural machine translation. *Proceedings of the 28th International Conference on Computational Linguistics*, 4506–4520. https://doi.org/10.18653/v1/2020.coling-main.398

Eikema, B., & Aziz, W. (2021). Sampling-based minimum bayes risk decoding for neural machine translation.

Esplà, M., Forcada, M., Ramírez-Sánchez, G., & Hoang, H. (2019). ParaCrawl: Web-scale parallel corpora for the languages of the EU. *Proceedings of Machine Translation Summit XVII Volume 2: Translator, Project and User Tracks*, 118–119. https://www.aclweb.org/anthology/W19-6721

Fan, A., Lewis, M., & Dauphin, Y. (2018). Hierarchical neural story generation. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 889–898. https://doi.org/10.18653/v1/P18-1082

Farinha, A. C., Farajian, M. A., Buchicchio, M., Fernandes, P., C. de Souza, J. G., Moniz, H., & Martins, A. F. T. (2022, December). Findings of the WMT 2022 shared task on chat translation. In P. Koehn, L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussà, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. Jimeno Yepes, T. Kocmi, A. Martins, M. Morishita, . . . M. Zampieri (Eds.), *Proceedings of the seventh conference on machine translation (wmt)* (pp. 724–743). Association for Computational Linguistics. https://aclanthology.org/2022.wmt-1.70

Farinhas, A., de Souza, J., & Martins, A. (2023, December). An empirical study of translation hypothesis ensembling with large language models. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 11956–11970). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.733

Faysse, M., Fernandes, P., Guerreiro, N. M., Loison, A., Alves, D. M., Corro, C., Boizard, N., Alves, J., Rei, R., Martins, P. H., Casademunt, A. B., Yvon, F., Martins, A. F. T., Viaud, G., Hudelot, C., & Colombo, P. (2025). Croissantllm: A truly bilingual french-english language model. *Transactions on Machine Learning Research*. https://openreview.net/forum?id=uA19Xo1o31

Feng, S., Wallace, E., Grissom II, A., Iyyer, M., Rodriguez, P., & Boyd-Graber, J. (2018). Pathologies of neural models make interpretations difficult. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3719–3728. https://doi.org/10.18653/v1/D18-1407

Fernandes, P., Agrawal, S., Zaranis, E., Martins, A. F. T., & Neubig, G. (2025). Do llms understand your translations? evaluating paragraph-level mt with question answering. *Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2504.07583

Fernandes, P., Deutsch, D., Finkelstein, M., Riley, P., Martins, A. F. T., Neubig, G., Garg, A., Clark, J. H., Freitag, M., & Firat, O. (2023). The devil is in the errors: Leveraging large language models for fine-grained machine translation evaluation. *Proceedings of the Seventh Conference on Machine Translation (WMT)*.

Fernandes, P., Farinhas, A., Rei, R., C. de Souza, J. G., Ogayo, P., Neubig, G., & Martins, A. (2022, July). Quality-aware decoding for neural machine translation. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Proceedings of the 2022 conference of the north american chapter of the association for computational linguistics: Human language technologies* (pp. 1396–1412). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.naacl-main.100

Fernandes, P., Ghorbani, B., Garcia, X., Freitag, M., & Firat, O. (2023, 23–29 Jul). Scaling laws for multilingual neural machine translation. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th international conference on machine learning* (pp. 10053–10071, Vol. 202). PMLR. https://proceedings.mlr.press/v202/fernandes23a.html

Fernandes, P., Madaan, A., Liu, E., Farinhas, A., Martins, P. H., Bertsch, A., de Souza, J. G. C., Zhou, S., Wu, T., Neubig, G., & Martins, A. F. T. (2023). Bridging the gap: A survey on integrating (human) feedback for natural language generation. *Transactions of the Association for Computational Linguistics*. https://arxiv.org/abs/2305.00955

Fernandes, P., Treviso, M. V., Pruthi, D., Martins, A. F. T., & Neubig, G. (2022). Learning to scaffold: Optimizing model explanations for teaching. *Advances in Neural Information Processing Systems*. https://openreview.net/pdf?id=V5rlSPsHpkf

Fernandes, P., Yin, K., Liu, E., Martins, A., & Neubig, G. (2023, July). When does translation require context? a data-driven, multilingual exploration [Awarded **Best Resource Paper**]. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.36

Fernandes, P., Yin, K., Neubig, G., & Martins, A. F. T. (2021, August). Measuring and increasing context usage in context-aware machine translation. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *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)* (pp. 6467–6478). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.505

Finn, C., Abbeel, P., & Levine, S. (2017, June). Model-agnostic meta-learning for fast adaptation of deep networks. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th international conference on machine learning* (pp. 1126–1135, Vol. 70). PMLR. https://proceedings.mlr.press/v70/finn17a.html

Flemings, J., Jiang, B., Zhang, W., Takhirov, Z., & Annavaram, M. (2025). Estimating privacy leakage of augmented contextual knowledge in language models. https://arxiv.org/abs/2410.03026

Fomicheva, M., Lertvittayakumjorn, P., Zhao, W., Eger, S., & Gao, Y. (2021). The Eval4NLP shared task on explainable quality estimation: Overview and results. *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, 165–178. https://doi.org/10.18653/v1/2021.eval4nlp-1.17

Fomicheva, M., Specia, L., & Aletras, N. (2021). Translation error detection as rationale extraction. *arXiv preprint arXiv:2108.12197*.

Fomicheva, M., Sun, S., Yankovskaya, L., Blain, F., Guzmán, F., Fishel, M., Aletras, N., Chaudhary, V., & Specia, L. (2020). Unsupervised quality estimation for neural

machine translation. *Transactions of the Association for Computational Linguistics*, *8*, 539–555.
Fonseca, E., Yankovskaya, L., Martins, A. F. T., Fishel, M., & Federmann, C. (2019). Findings of the wmt 2019 shared tasks on quality estimation. *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, 1–12. http://www.aclweb.org/anthology/W19-5401
Frank, S., Bugliarello, E., & Elliott, D. (2021, November). Vision-and-language or vision-for-language? on cross-modal influence in multimodal transformers. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Proceedings of the 2021 conference on empirical methods in natural language processing* (pp. 9847–9857). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.775
Freitag, M., Foster, G., Grangier, D., Ratnakar, V., Tan, Q., & Macherey, W. (2021). Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, *9*, 1460–1474. https://doi.org/10.1162/tacl_a_00437
Freitag, M., Grangier, D., Tan, Q., & Liang, B. (2021). Minimum bayes risk decoding with neural metrics of translation quality.
Freitag, M., Rei, R., Mathur, N., Lo, C.-k., Stewart, C., Avramidis, E., Kocmi, T., Foster, G., Lavie, A., & Martins, A. F. T. (2022). Results of WMT22 metrics shared task: Stop using BLEU – neural metrics are better and more robust. *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 46–68. https://aclanthology.org/2022.wmt-1.2
Freitag, M., Rei, R., Mathur, N., Lo, C.-k., Stewart, C., Foster, G., Lavie, A., & Bojar, O. (2021). Results of the WMT21 metrics shared task: Evaluating metrics with expert-based human evaluations on TED and news domain. *Proceedings of the Sixth Conference on Machine Translation*, 733–774. https://aclanthology.org/2021.wmt-1.73
Fu, Y., Peng, H., Ou, L., Sabharwal, A., & Khot, T. (2023, 23–29 Jul). Specializing smaller language models towards multi-step reasoning. In A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, & J. Scarlett (Eds.), *Proceedings of the 40th international conference on machine learning* (pp. 10421–10430, Vol. 202). PMLR. https://proceedings.mlr.press/v202/fu23d.html
Gao, L., Schulman, J., & Hilton, J. (2022). Scaling laws for reward model overoptimization.
Gao, L., Madaan, A., Zhou, S., Alon, U., Liu, P., Yang, Y., Callan, J., & Neubig, G. (2022). Pal: Program-aided language models. *arXiv preprint arXiv:2211.10435*.
Ghorbani, B., Firat, O., Freitag, M., Bapna, A., Krikun, M., García, X., Chelba, C., & Cherry, C. (2022). Scaling laws for neural machine translation. *ArXiv*, *abs/2109.07740*.
Ghosal, G. R., Zurek, M., Brown, D. S., & Dragan, A. D. (2023). The effect of modeling human rationality level on learning rewards from multiple feedback types.

Gordon, M. A., Duh, K., & Kaplan, J. (2021). Data and parameter scaling laws for neural machine translation. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 5915–5922. https://doi.org/10.18653/v1/2021.emnlp-main.478

Graham, Y., Baldwin, T., Moffat, A., & Zobel, J. (2013). Continuous measurement scales in human evaluation of machine translation. *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, 33–41. https://aclanthology.org/W13-2305

Graves, A. (2012a). Sequence transduction with recurrent neural networks.

Graves, A. (2012b). Sequence transduction with recurrent neural networks. *ArXiv, abs/1211.3711*. https://api.semanticscholar.org/CorpusID:17194112

Grefenstette, E., Amos, B., Yarats, D., Htut, P. M., Molchanov, A., Meier, F., Kiela, D., Cho, K., & Chintala, S. (2019). Generalized inner loop meta-learning. *arXiv preprint arXiv:1910.01727*.

Guerreiro, N. M., & Martins, A. F. T. (2021). SPECTRA: Sparse structured text rationalization. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6534–6550. https://doi.org/10.18653/v1/2021.emnlp-main.525

Guerreiro, N. M., Rei, R., van Stigt, D., Coheur, L., Colombo, P., & Martins, A. F. T. (2023). Xcomet: Transparent machine translation evaluation through fine-grained error detection.

Gulcehre, C., Paine, T. L., Srinivasan, S., Konyushkova, K., Weerts, L., Sharma, A., Siddhant, A., Ahern, A., Wang, M., Gu, C., Macherey, W., Doucet, A., Firat, O., & de Freitas, N. (2023). Reinforced self-training (rest) for language modeling.

Hase, P., & Bansal, M. (2020). Evaluating explainable AI: Which algorithmic explanations help users predict model behavior? *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5540–5552. https://doi.org/10.18653/v1/2020.acl-main.491

Hassan, H., Aue, A., Chen, C., Chowdhary, V., Clark, J., Federmann, C., Huang, X., Junczys-Dowmunt, M., Lewis, W., Li, M., Liu, S., Liu, T., Luo, R., Menezes, A., Qin, T., Seide, F., Tan, X., Tian, F., Wu, L., . . . Zhou, M. (2018). Achieving human parity on automatic chinese to english news translation. *CoRR*, *abs/1803.05567*. http://arxiv.org/abs/1803.05567

He, Y., Benhaim, A., Patra, B., Vaddamanu, P., Ahuja, S., Chopra, P., Chaudhary, V., Zhao, H., & Song, X. (2024). Scaling laws for multilingual language models. *ArXiv*, *abs/2410.12883*. https://api.semanticscholar.org/CorpusID:273404359

Heek, J., Levskaya, A., Oliver, A., Ritter, M., Rondepierre, B., Steiner, A., & van Zee, M. (2020). *Flax: A neural network library and ecosystem for JAX* (Version 0.3.4). http://github.com/google/flax

Henighan, T., Kaplan, J., Katz, M., Chen, M., Hesse, C., Jackson, J., Jun, H., Brown, T. B., Dhariwal, P., Gray, S., et al. (2020). Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*.

Herbrich, R., Minka, T., & Graepel, T. (2007). Trueskill(tm): A bayesian skill rating system (Advances in Neural Information Processing Systems 20). *Advances in Neural Information Processing Systems 20*, 569–576. https://www.microsoft.com/en-us/research/publication/trueskilltm-a-bayesian-skill-rating-system/

Hernandez, D., Kaplan, J., Henighan, T. J., & McCandlish, S. (2021). Scaling laws for transfer. *ArXiv*, *abs/2102.01293*.

Hestness, J., Narang, S., Ardalani, N., Diamos, G., Jun, H., Kianinejad, H., Patwary, M., Ali, M., Yang, Y., & Zhou, Y. (2017). Deep learning scaling is predictable, empirically. *arXiv preprint arXiv:1712.00409*.

Ho, N., Schmid, L., & Yun, S.-Y. (2023, July). Large language models are reasoning teachers. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 14852–14882). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.830

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, *9*, 1735–80. https://doi.org/10.1162/neco.1997.9.8.1735

Hoffmann, J., Borgeaud, S., Mensch, A., Buchatskaya, E., Cai, T., Rutherford, E., Casas, D. d. L., Hendricks, L. A., Welbl, J., Clark, A., Hennigan, T., Noland, E., Millican, K., Driessche, G. v. d., Damoc, B., Guy, A., Osindero, S., Simonyan, K., Elsen, E., … Sifre, L. (2022). Training compute-optimal large language models. https://doi.org/10.48550/ARXIV.2203.15556

Holtzman, A., Buys, J., Du, L., Forbes, M., & Choi, Y. (2020). The curious case of neural text degeneration. *Eighth International Conference on Learning Representations, ICLR*. https://openreview.net/forum?id=rygGQyrFvH

Huang, Y., Cheng, Y., Bapna, A., Firat, O., Chen, D., Chen, M., Lee, H., Ngiam, J., Le, Q. V., Wu, Y., et al. (2019). Gpipe: Efficient training of giant neural networks using pipeline parallelism. *Advances in neural information processing systems*, *32*.

Jacovi, A., & Goldberg, Y. (2020). Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4198–4205. https://doi.org/10.18653/v1/2020.acl-main.386

Jacovi, A., & Goldberg, Y. (2021). Aligning Faithful Interpretations with their Social Attribution. *Transactions of the Association for Computational Linguistics*, *9*, 294–310. https://doi.org/10.1162/tacl_a_00367

Jain, S., Keshava, V., Mysore Sathyendra, S., Fernandes, P., Liu, P., Neubig, G., & Zhou, C. (2023, July). Multi-dimensional evaluation of text summarization with in-context learning. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Findings of the associ-*

ation for computational linguistics: Acl 2023* (pp. 8487–8495). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.findings-acl.537

Jain, S., Keshava, V., Sathyendra, S. M., Fernandes, P., Liu, P., Neubig, G., & Zhou, C. (2023). Multi-dimensional evaluation of text summarization with in-context learning.

Jain, S., & Wallace, B. C. (2019). Attention is not Explanation. *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)*, 3543–3556. https://doi.org/10.18653/v1/N19-1357

Jean, S., Lauly, S., Firat, O., & Cho, K. (2017). Does neural machine translation benefit from larger context?

Jean, S., & Cho, K. (2019). Context-aware learning for neural machine translation. *CoRR, abs/1903.04715*. http://arxiv.org/abs/1903.04715

Kaplan, J., McCandlish, S., Henighan, T. J., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models. *ArXiv, abs/2001.08361*. https://api.semanticscholar.org/CorpusID:210861095

Kaplan, J., McCandlish, S., Henighan, T., Brown, T. B., Chess, B., Child, R., Gray, S., Radford, A., Wu, J., & Amodei, D. (2020). Scaling laws for neural language models. *CoRR, abs/2001.08361*. https://arxiv.org/abs/2001.08361

Kenton, Z., Everitt, T., Weidinger, L., Gabriel, I., Mikulik, V., & Irving, G. (2021). Alignment of language agents. *CoRR, abs/2103.14659*. https://arxiv.org/abs/2103.14659

Kepler, F., Trénous, J., Treviso, M., Vera, M., & Martins, A. F. T. (2019). OpenKiwi: An open source framework for quality estimation. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 117–122. https://doi.org/10.18653/v1/P19-3020

Kiegeland, S., & Kreutzer, J. (2021). Revisiting the weaknesses of reinforcement learning for neural machine translation. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1673–1681. https://doi.org/10.18653/v1/2021.naacl-main.133

Kim, S., Yi, J., Kim, E., & Yoon, S. (2020). Interpretation of NLP models through input marginalization. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 3154–3167. https://doi.org/10.18653/v1/2020.emnlp-main.255

Kim, Y., & Rush, A. M. (2016, November). Sequence-level knowledge distillation. In J. Su, K. Duh, & X. Carreras (Eds.), *Proceedings of the 2016 conference on empirical methods in natural language processing* (pp. 1317–1327). Association for Computational Linguistics. https://doi.org/10.18653/v1/D16-1139

Kim, Y., Tran, D. T., & Ney, H. (2019). When and why is document-level context useful in neural machine translation? *Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019)*, 24–34. https://doi.org/10.18653/v1/D19-6503

Kobayashi, G., Kuribayashi, T., Yokoi, S., & Inui, K. (2020). Attention is not only a weight: Analyzing transformers with vector norms. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 7057–7075. https://doi.org/10.18653/v1/2020.emnlp-main.574

Kocmi, T., Avramidis, E., Bawden, R., Bojar, O., Dvorkovich, A., Federmann, C., Fishel, M., Freitag, M., Gowda, T., Grundkiewicz, R., Haddow, B., Karpinska, M., Koehn, P., Marie, B., Monz, C., Murray, K., Nagata, M., Popel, M., Popović, M., . . . Zouhar, V. (2024, November). Findings of the WMT24 general machine translation shared task: The LLM era is here but MT is not solved yet. In B. Haddow, T. Kocmi, P. Koehn, & C. Monz (Eds.), *Proceedings of the ninth conference on machine translation* (pp. 1–46). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.wmt-1.1

Kocmi, T., Bawden, R., Bojar, O., Dvorkovich, A., Federmann, C., Fishel, M., Gowda, T., Graham, Y., Grundkiewicz, R., Haddow, B., Knowles, R., Koehn, P., Monz, C., Morishita, M., Nagata, M., Nakazawa, T., Novák, M., Popel, M., & Popović, M. (2022). Findings of the 2022 conference on machine translation (WMT22). *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 1–45. https://aclanthology.org/2022.wmt-1.1

Kocmi, T., & Federmann, C. (2023). Large language models are state-of-the-art evaluators of translation quality.

Kocmi, T., Federmann, C., Grundkiewicz, R., Junczys-Dowmunt, M., Matsushita, H., & Menezes, A. (2021a). To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. *Proceedings of the Sixth Conference on Machine Translation*, 478–494. https://aclanthology.org/2021.wmt-1.57

Kocmi, T., Federmann, C., Grundkiewicz, R., Junczys-Dowmunt, M., Matsushita, H., & Menezes, A. (2021b, November). To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussa, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. J. Yepes, P. Koehn, T. Kocmi, A. Martins, M. Morishita, & C. Monz (Eds.), *Proceedings of the sixth conference on machine translation* (pp. 478–494). Association for Computational Linguistics. https://aclanthology.org/2021.wmt-1.57

Koehn, P., & Knowles, R. (2017). Six challenges for neural machine translation. *Proceedings of the First Workshop on Neural Machine Translation*, 28–39. https://doi.org/10.18653/v1/W17-3204

Korbak, T., Shi, K., Chen, A., Bhalerao, R., Buckley, C. L., Phang, J., Bowman, S. R., & Perez, E. (2023). Pretraining language models with human preferences. *Proceedings of the 40th International Conference on Machine Learning*.

Krizhevsky, A. (2009). *Learning multiple layers of features from tiny images* (tech. rep.). University of Toronto.

Kudo, T. (2018). Subword regularization: Improving neural network translation models with multiple subword candidates. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 66–75. https://doi.org/10.18653/v1/P18-1007

Kudo, T., & Richardson, J. (2018). SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 66–71. https://doi.org/10.18653/v1/D18-2012

Kumar, S., & Byrne, W. (2002). Minimum bayes-risk word alignments of bilingual texts. *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing - Volume 10*, 140–147. https://doi.org/10.3115/1118693.1118712

Kumar, S., & Byrne, W. (2004). Minimum Bayes-risk decoding for statistical machine translation. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, 169–176. https://aclanthology.org/N04-1022

Kurin, V., De Palma, A., Kostrikov, I., Whiteson, S., & Kumar, M. P. (2022). In defense of the unitary scalarization for deep multi-task learning. *arXiv preprint arXiv:2201.04122*.

Lanham, T., Chen, A., Radhakrishnan, A., Steiner, B., Denison, C., Hernandez, D., Li, D., Durmus, E., Hubinger, E., Kernion, J., Lukošiūtė, K., Nguyen, K., Cheng, N., Joseph, N., Schiefer, N., Rausch, O., Larson, R., McCandlish, S., Kundu, S., … Perez, E. (2023). Measuring faithfulness in chain-of-thought reasoning.

Läubli, S., Sennrich, R., & Volk, M. (2018). Has machine translation achieved human parity? a case for document-level evaluation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4791–4796. https://doi.org/10.18653/v1/D18-1512

Lavie, A., & Denkowski, M. J. (2009). The meteor metric for automatic evaluation of machine translation. *Machine Translation*, *23*(2-3), 105–115. https://doi.org/10.1007/s10590-009-9059-4

Leblond, R., Alayrac, J.-B., Sifre, L., Pislar, M., Jean-Baptiste, L., Antonoglou, I., Simonyan, K., & Vinyals, O. (2021). Machine translation decoding beyond beam search. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 8410–8434. https://aclanthology.org/2021.emnlp-main.662

Lee, A., Auli, M., & Ranzato, M. (2021). Discriminative reranking for neural machine translation. *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)*, 7250–7264. https://doi.org/10.18653/v1/2021.acl-long.563

Lei, T., Barzilay, R., & Jaakkola, T. (2016). Rationalizing neural predictions. *roceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 107–117.

Lepikhin, D., Lee, H., Xu, Y., Chen, D., Firat, O., Huang, Y., Krikun, M., Shazeer, N., & Chen, Z. (2020). Gshard: Scaling giant models with conditional computation and automatic sharding. *arXiv preprint arXiv:2006.16668*.

Lertvittayakumjorn, P., & Toni, F. (2019). Human-grounded evaluations of explanation methods for text classification. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5195–5205. https://doi.org/10.18653/v1/D19-1523

Li, J., Miller, A. H., Chopra, S., Ranzato, M., & Weston, J. (2017). Dialogue learning with human-in-the-loop. *International Conference on Learning Representations*. https://openreview.net/forum?id=HJgXCV9xx

Li, L. H., Hessel, J., Yu, Y., Ren, X., Chang, K.-W., & Choi, Y. (2023, July). Symbolic chain-of-thought distillation: Small models can also "think" step-by-step. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)* (pp. 2665–2679). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.150

Li, S., Chen, J., Shen, Y., Chen, Z., Zhang, X., Li, Z., Wang, H., Qian, J., Peng, B., Mao, Y., Chen, W., & Yan, X. (2022). Explanations from large language models make small reasoners better. https://arxiv.org/abs/2210.06726

Lipton, Z. C. (2016a). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, *16*(3), 31–57. https://doi.org/10.1145/3236386.3241340

Lipton, Z. C. (2016b). The mythos of model interpretability. *Communications of the ACM*, *61*, 36–43. https://api.semanticscholar.org/CorpusID:5981909

Liu, E., Bertsch, A., Sutawika, L., Tjuatja, L., Fernandes, P., Marinov, L., Chen, M., Singhal, S., Lawrence, C., Raghunathan, A., Gashteovski, K., & Neubig, G. (2025). Not-just-scaling laws: Towards a better understanding of the downstream impact of language model design decisions. *The 2025 Conference on Empirical Methods in Natural Language Processing*. https://openreview.net/forum?id=XgGX19LJ2e

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.*, *55*(9). https://doi.org/10.1145/3560815

Liu, Y., Iter, D., Xu, Y., Wang, S., Xu, R., & Zhu, C. (2023). G-eval: Nlg evaluation using gpt-4 with better human alignment.

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., & Zettlemoyer, L. (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, *8*, 726–742. https://doi.org/10.1162/tacl_a_00343

Lommel, A., Uszkoreit, H., & Burchardt, A. (2014). Multidimensional quality metrics (MQM): A framework for declaring and describing translation quality metrics. *Revista Tradumàtica: tecnologies de la traducció*.

Lopes, A., Farajian, M. A., Bawden, R., Zhang, M., & Martins, A. F. T. (2020). Document-level neural MT: A systematic comparison. *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 225–234. https://www.aclweb.org/anthology/2020.eamt-1.24

Loshchilov, I., & Hutter, F. (2019). Decoupled weight decay regularization. *International Conference on Learning Representations*. https://openreview.net/forum?id=Bkg6RiCqY7

Lu, Q., Qiu, B., Ding, L., Xie, L., & Tao, D. (2023). Error analysis prompting enables human-like translation evaluation in large language models: A case study on chatgpt. *arXiv preprint*. https://arxiv.org/pdf/2303.13809.pdf

Luong, M.-T., Le, Q. V., Sutskever, I., Vinyals, O., & Kaiser, L. (2015). Multi-task sequence to sequence learning. https://doi.org/10.48550/ARXIV.1511.06114

Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 142–150. http://www.aclweb.org/anthology/P11-1015

Madaan, A., Tandon, N., Rajagopal, D., Clark, P., Yang, Y., & Hovy, E. (2021). Think about it! improving defeasible reasoning by first modeling the question scenario. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6291–6310. https://doi.org/10.18653/v1/2021.emnlp-main.508

Martins, A., & Astudillo, R. (2016, 20–22 Jun). From softmax to sparsemax: A sparse model of attention and multi-label classification. In M. F. Balcan & K. Q. Weinberger (Eds.), *Proceedings of the 33rd international conference on machine learning* (pp. 1614–1623, Vol. 48). PMLR. https://proceedings.mlr.press/v48/martins16.html

Martins, P., Fernandes, P., Alves, J., Guerreiro, N., Rei, R., Alves, D., Pombal, J., Farajian, A., Faysse, M., Klimaszewski, M., Colombo, P., Haddow, B., De Souza, J., Birch, A., & Martins, A. (2025). Eurollm: Multilingual language models for europe [Published by Elsevier B.V.; 2nd EuroHPC user day, EuroHPC 2024]. *Procedia Computer Science*, *255*, 53–62. https://doi.org/10.1016/j.procs.2025.02.260

Maruf, S., & Haffari, G. (2018). Document context neural machine translation with memory networks. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1275–1284. https://doi.org/10.18653/v1/P18-1118

Maruf, S., Martins, A. F. T., & Haffari, G. (2019). Selective attention for context-aware neural machine translation. *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)*, 3092–3102. https://doi.org/10.18653/v1/N19-1313

Maruf, S., Saleh, F., & Haffari, G. (2019). A survey on document-level machine translation: Methods and evaluation. *ArXiv*, *abs/1912.08494*.

Mathur, N., Baldwin, T., & Cohn, T. (2020). Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 4984–4997. https://doi.org/10.18653/v1/2020.acl-main.448

Mathur, N., Wei, J., Freitag, M., Ma, Q., & Bojar, O. (2020). Results of the WMT20 metrics shared task. *Proceedings of the Fifth Conference on Machine Translation*, 688–725. https://aclanthology.org/2020.wmt-1.77

McKenzie, I. R., Lyzhov, A., Pieler, M., Parrish, A., Mueller, A., Prabhu, A., McLean, E., Kirtland, A., Ross, A., Liu, A., Gritsevskiy, A., Wurgaft, D., Kauffman, D., Recchia, G., Liu, J., Cavanagh, J., Weiss, M., Huang, S., Droid, T. F., … Perez, E. (2024). Inverse scaling: When bigger isn't better. https://arxiv.org/abs/2306.09479

Mehandru, N., Agrawal, S., Xiao, Y., Gao, G., Khoong, E., Carpuat, M., & Salehi, N. (2023, December). Physician detection of clinical harm in machine translation: Quality estimation aids in reliance and backtranslation identifies critical errors. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 11633–11647). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.712

Meister, C., Cotterell, R., & Vieira, T. (2020). If beam search is the answer, what was the question? *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2173–2185. https://doi.org/10.18653/v1/2020.emnlp-main.170

Miculicich, L., Ram, D., Pappas, N., & Henderson, J. (2018). Document-level neural machine translation with hierarchical attention networks. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2947–2954. https://doi.org/10.18653/v1/D18-1325

Miller, T. (2019). Explanation in artificial intelligence: Insights from the social sciences. *Artificial intelligence*, *267*, 1–38.

Mu, J., & Andreas, J. (2020). Compositional explanations of neurons. *ArXiv*, *abs/2006.14032*.

Müller, M., Rios, A., Voita, E., & Sennrich, R. (2018). A large-scale test set for the evaluation of context-aware pronoun translation in neural machine translation. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 61–72. https://doi.org/10.18653/v1/W18-6307

Müller, M., & Sennrich, R. (2021). Understanding the properties of minimum Bayes risk decoding in neural machine translation. *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)*, 259–272. https://doi.org/10.18653/v1/2021.acl-long.22

Murray, K., & Chiang, D. (2018). Correcting length bias in neural machine translation. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 212–223. https://doi.org/10.18653/v1/W18-6322

Neely, M., Schouten, S. F., Bleeker, M. J. R., & Lucic, A. (2021). Order in the court: Explainable ai methods prone to disagreement.

Neubig, G. (2013). Travatar: A forest-to-string machine translation engine based on tree transducers. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 91–96. https://aclanthology.org/P13-4016

Ng, N., Yee, K., Baevski, A., Ott, M., Auli, M., & Edunov, S. (2019). Facebook FAIR's WMT19 news translation task submission. *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, 314–319. https://doi.org/10.18653/v1/W19-5333

Och, F. J. (2003). Minimum error rate training in statistical machine translation. *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, 160–167. https://doi.org/10.3115/1075096.1075117

O'Connor, J., & Andreas, J. (2021, August). What context features can transformer language models use? In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *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)* (pp. 851–864). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.70

OpenAI. (2023). Gpt-4 technical report.

Ott, M., Auli, M., Grangier, D., & Ranzato, M. (2018, October). Analyzing uncertainty in neural machine translation. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th international conference on machine learning* (pp. 3956–3965, Vol. 80). PMLR. https://proceedings.mlr.press/v80/ott18a.html

Ott, M., Edunov, S., Baevski, A., Fan, A., Gross, S., Ng, N., Grangier, D., & Auli, M. (2019). Fairseq: A fast, extensible toolkit for sequence modeling. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, 48–53. https://doi.org/10.18653/v1/N19-4009

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002a). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 311–318. https://doi.org/10.3115/1073083.1073135

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002b). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.

Papineni, K., Roukos, S., Ward, T., & Zhu, W.-J. (2002c). Bleu: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the As-*

sociation for Computational Linguistics, 311–318. https://doi.org/10.3115/1073083.1073135

Perrella, S., Proietti, L., Scirè, A., Campolungo, N., & Navigli, R. (2022, December). MaTESe: Machine translation evaluation as a sequence tagging problem. In P. Koehn, L. Barrault, O. Bojar, F. Bougares, R. Chatterjee, M. R. Costa-jussà, C. Federmann, M. Fishel, A. Fraser, M. Freitag, Y. Graham, R. Grundkiewicz, P. Guzman, B. Haddow, M. Huck, A. Jimeno Yepes, T. Kocmi, A. Martins, M. Morishita, … M. Zampieri (Eds.), *Proceedings of the seventh conference on machine translation (wmt)* (pp. 569–577). Association for Computational Linguistics. https://aclanthology.org/2022.wmt-1.51

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., & Zettlemoyer, L. (2018). Deep contextualized word representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2227–2237. https://doi.org/10.18653/v1/N18-1202

Pombal, J., Yoon, D., Fernandes, P., Wu, I., Kim, S., Rei, R., Neubig, G., & Martins, A. F. T. (2025). M-prometheus: A suite of open multilingual llm judges. *Proceedings of the Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2504.04953

Popović, M. (2015). ChrF: Character n-gram F-score for automatic MT evaluation. *Proceedings of the Tenth Workshop on Statistical Machine Translation*, 392–395. https://doi.org/10.18653/v1/W15-3049

Post, M. (2018). A call for clarity in reporting BLEU scores. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 186–191. https://doi.org/10.18653/v1/W18-6319

Pruthi, D., Bansal, R., Dhingra, B., Baldini Soares, L., Collins, M., Lipton, Z. C., Neubig, G., & Cohen, W. W. (2022). Evaluating explanations: How much do explanations from the teacher aid students? (B. Roark & A. Nenkova, Eds.). *Transactions of the Association for Computational Linguistics*, *10*, 359–375. https://doi.org/10.1162/tacl_a_00465

Pruthi, D., Dhingra, B., Soares, L. B., Collins, M., Lipton, Z. C., Neubig, G., & Cohen, W. W. (2020). Evaluating explanations: How much do explanations from the teacher aid students? *CoRR*, *abs/2012.00893*. https://arxiv.org/abs/2012.00893

Pruthi, D., Gupta, M., Dhingra, B., Neubig, G., & Lipton, Z. C. (2020, July). Learning to deceive with attention-based explanations. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th annual meeting of the association for computational linguistics* (pp. 4782–4793). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.432

Pu, A., Chung, H. W., Parikh, A., Gehrmann, S., & Sellam, T. (2021). Learning compact metrics for MT. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 751–762. https://aclanthology.org/2021.emnlp-main.58

Qi, J., Fernández, R., & Bisazza, A. (2023). Cross-lingual consistency of factual knowledge in multilingual language models. *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 10650–10666. https://doi.org/10.18653/v1/2023.emnlp-main.658

Qin, Y., Yuan, W., Neubig, G., & Liu, P. (2022). T5score: Discriminative fine-tuning of generative evaluation metrics. *ArXiv*, *abs/2212.05726*.

Qwen, : Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., … Qiu, Z. (2025). Qwen2.5 technical report. https://arxiv.org/abs/2412.15115

Radford, A., & Narasimhan, K. (2018). Improving language understanding by generative pre-training. https://api.semanticscholar.org/CorpusID:49313245

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020a). Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, *21*(140), 1–67. http://jmlr.org/papers/v21/20-074.html

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., & Liu, P. J. (2020b). Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, *21*(1).

Raghu, A., Raghu, M., Kornblith, S., Duvenaud, D., & Hinton, G. (2021). Teaching with commentaries. *International Conference on Learning Representations*. https://openreview.net/forum?id=4RbdgBh9gE

Ramos, M., Fernandes, P., Agrawal, S., & Martins, A. F. T. (2025). Multilingual contextualization of large language models for document-level machine translation. *Proceedings of the Second Conference on Language Modeling (COLM)*. https://arxiv.org/abs/2504.12140

Ramos, M., Fernandes, P., Farinhas, A., & Martins, A. (2024, June). Aligning neural machine translation models: Human feedback in training and inference. In C. Scarton, C. Prescott, C. Bayliss, C. Oakley, J. Wright, S. Wrigley, X. Song, E. Gow-Smith, R. Bawden, V. M. Sánchez-Cartagena, P. Cadwell, E. Lapshinova-Koltunski, V. Cabarrão, K. Chatzitheodorou, M. Nurminen, D. Kanojia, & H. Moniz (Eds.), *Proceedings of the 25th annual conference of the european association for machine translation (volume 1)* (pp. 258–274). European Association for Machine Translation (EAMT). https://aclanthology.org/2024.eamt-1.22/

Ranasinghe, T., Orasan, C., & Mitkov, R. (2020). TransQuest: Translation quality estimation with cross-lingual transformers. *Proceedings of the 28th International Conference on Computational Linguistics*, 5070–5081. https://doi.org/10.18653/v1/2020.coling-main.445

Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S. G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J. T., et al. (2022). A generalist agent. *arXiv preprint arXiv:2205.06175*.

Rei, R., C. de Souza, J. G., Alves, D., Zerva, C., Farinha, A. C., Glushkova, T., Lavie, A., Coheur, L., & Martins, A. F. T. (2022). COMET-22: Unbabel-IST 2022 submission for the metrics shared task. *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 578–585. https://aclanthology.org/2022.wmt-1.52

Rei, R., Farinha, A. C., Zerva, C., van Stigt, D., Stewart, C., Ramos, P., Glushkova, T., Martins, A. F. T., & Lavie, A. (2021). Are references really needed? unbabel-IST 2021 submission for the metrics shared task. *Proceedings of the Sixth Conference on Machine Translation*, 1030–1040. https://aclanthology.org/2021.wmt-1.111

Rei, R., Pombal, J., Guerreiro, N. M., Alves, J., Martins, P. H., Fernandes, P., Wu, H., Vaz, T., Alves, D., Farajian, A., Agrawal, S., Farinhas, A., C. De Souza, J. G., & Martins, A. (2024, November). Tower v2: Unbabel-IST 2024 submission for the general MT shared task. In B. Haddow, T. Kocmi, P. Koehn, & C. Monz (Eds.), *Proceedings of the ninth conference on machine translation* (pp. 185–204). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.wmt-1.12

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020a). COMET: A neural framework for MT evaluation. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2685–2702. https://doi.org/10.18653/v1/2020.emnlp-main.213

Rei, R., Stewart, C., Farinha, A. C., & Lavie, A. (2020b). Unbabel's participation in the WMT20 metrics shared task. *Proceedings of the Fifth Conference on Machine Translation*, 911–920. https://aclanthology.org/2020.wmt-1.101

Rei, R., Treviso, M., Guerreiro, N. M., Zerva, C., Farinha, A. C., Maroti, C., C. de Souza, J. G., Glushkova, T., Alves, D., Coheur, L., Lavie, A., & Martins, A. F. T. (2022). CometKiwi: IST-unbabel 2022 submission for the quality estimation shared task. *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 634–645. https://aclanthology.org/2022.wmt-1.60

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "why should I trust you?": Explaining the predictions of any classifier. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 1135–1144.

Ridnik, T., Ben-Baruch, E., Noy, A., & Zelnik-Manor, L. (2021). Imagenet-21k pretraining for the masses. *ArXiv, abs/2104.10972*.

Rosenfeld, J. S., Rosenfeld, A., Belinkov, Y., & Shavit, N. (2019). A constructive prediction of the generalization error across scales. *International Conference on Learning Representations*.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, *323*, 533–536. https://api.semanticscholar.org/CorpusID:205001834

Sanh, V., Webson, A., Raffel, C., Bach, S., Sutawika, L., Alyafeai, Z., Chaffin, A., Stiegler, A., Raja, A., Dey, M., Bari, M. S., Xu, C., Thakker, U., Sharma, S. S., Szczechla, E., Kim, T., Chhablani, G., Nayak, N., Datta, D., . . . Rush, A. M. (2022). Multitask prompted training enables zero-shot task generalization. *International Conference on Learning Representations*. https://openreview.net/forum?id=9Vrb9D0WI4

Schäfer, A., Ravfogel, S., Hofmann, T., Pimentel, T., & Schlag, I. (2024). The role of language imbalance in cross-lingual generalisation: Insights from cloned language experiments. https://arxiv.org/abs/2404.07982

*Schmidhuber, J. (1987, 14 5). *Evolutionary principles in self-referential learning. on learning now to learn: The meta-meta-meta...-hook* [Diploma Thesis]. Technische Universitat Munchen, Germany. http://www.idsia.ch/~juergen/diploma.html

Sellam, T., Das, D., & Parikh, A. (2020). BLEURT: Learning robust metrics for text generation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7881–7892. https://doi.org/10.18653/v1/2020.acl-main.704

Sennrich, R., Haddow, B., & Birch, A. (2016a). Edinburgh neural machine translation systems for WMT 16. *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, 371–376. https://doi.org/10.18653/v1/W16-2323

Sennrich, R., Haddow, B., & Birch, A. (2016b). Neural machine translation of rare words with subword units. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1715–1725. https://doi.org/10.18653/v1/P16-1162

Serrano, S., & Smith, N. A. (2019). Is attention interpretable? *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2931–2951. https://doi.org/10.18653/v1/P19-1282

Shaban, A., Cheng, C.-A., Hatch, N., & Boots, B. (2019, 16–18 Apr). Truncated back-propagation for bilevel optimization. In K. Chaudhuri & M. Sugiyama (Eds.), *Proceedings of the twenty-second international conference on artificial intelligence and statistics* (pp. 1723–1732, Vol. 89). PMLR. https://proceedings.mlr.press/v89/shaban19a.html

Shaham, U., & Levy, O. (2021). What do you get when you cross beam search with nucleus sampling?

Shazeer, N. M., & Stern, M. (2018). Adafactor: Adaptive learning rates with sublinear memory cost. *ArXiv*, *abs/1804.04235*.

Shen, L., Sarkar, A., & Och, F. J. (2004). Discriminative reranking for machine translation. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, 177–184. https://aclanthology.org/N04-1023

Shen, S., Cheng, Y., He, Z., He, W., Wu, H., Sun, M., & Liu, Y. (2016). Minimum risk training for neural machine translation. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1683–1692. https://doi.org/10.18653/v1/P16-1159

Sheng, E., Chang, K.-W., Natarajan, P., & Peng, N. (2019). The woman worked as a babysitter: On biases in language generation. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3407–3412. https://doi.org/10.18653/v1/D19-1339

Shoeybi, M., Patwary, M., Puri, R., LeGresley, P., Casper, J., & Catanzaro, B. (2019). Megatron-lm: Training multi-billion parameter language models using model parallelism. https://doi.org/10.48550/ARXIV.1909.08053

Shu, R., & Nakayama, H. (2017). Later-stage minimum bayes-risk decoding for neural machine translation.

Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, *abs/1312.6034*.

Smith, D. A., & Eisner, J. (2006). Minimum risk annealing for training log-linear models. *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, 787–794. https://aclanthology.org/P06-2101

Specia, L., Blain, F., Fomicheva, M., Fonseca, E., Chaudhary, V., Guzmán, F., & Martins, A. F. T. (2020). Findings of the WMT 2020 shared task on quality estimation. *Proceedings of the Fifth Conference on Machine Translation*, 743–764. https://aclanthology.org/2020.wmt-1.79

Specia, L., Blain, F., Fomicheva, M., Zerva, C., Li, Z., Chaudhary, V., & Martins, A. F. T. (2021). Findings of the WMT 2021 shared task on quality estimation. *Proceedings of the Sixth Conference on Machine Translation*, 684–725. https://aclanthology.org/2021.wmt-1.71

Stahlberg, F., & Byrne, B. (2019). On NMT search errors and model errors: Cat got your tongue? *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3356–3362. https://doi.org/10.18653/v1/D19-1331

Stahlberg, F., de Gispert, A., Hasler, E., & Byrne, B. (2017). Neural machine translation by minimising the Bayes-risk with respect to syntactic translation lattices. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, 362–368. https://aclanthology.org/E17-2058

Stojanovski, D., Krojer, B., Peskov, D., & Fraser, A. (2020). ContraCAT: Contrastive coreference analytical templates for machine translation. *Proceedings of the 28th International Conference on Computational Linguistics*, 4732–4749. https://www.aclweb.org/anthology/2020.coling-main.417

Sun, J., Fernandes, P., Wang, X., & Neubig, G. (2023, May). A multi-dimensional evaluation of tokenizer-free multilingual pretrained models. In A. Vlachos & I. Augenstein (Eds.), *Findings of the association for computational linguistics: Eacl 2023* (pp. 1725–1735). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.findings-eacl.128

Sun, Z., Wang, M., Zhou, H., Zhao, C., Huang, S., Chen, J., & Li, L. (2020). Capturing longer context for document-level neural machine translation: A multi-resolutional approach. *arXiv, abs/2010.08961*.

Sundararajan, M., Taly, A., & Yan, Q. (2017a). Axiomatic attribution for deep networks. *International Conference on Machine Learning*. https://api.semanticscholar.org/CorpusID:16747630

Sundararajan, M., Taly, A., & Yan, Q. (2017b, June). Axiomatic attribution for deep networks. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th international conference on machine learning* (pp. 3319–3328, Vol. 70). PMLR. https://proceedings.mlr.press/v70/sundararajan17a.html

Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems* (Vol. 27). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2014/file/a14ac55a4f27472c5d894ec1c3c743d2-Paper.pdf

Team, G. (2025). Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. https://arxiv.org/abs/2507.06261

Tiedemann, J., & Scherrer, Y. (2017). Neural machine translation with extended context. *Proceedings of the Third Workshop on Discourse in Machine Translation*, 82–92. https://doi.org/10.18653/v1/W17-4811

Toral, A., Castilho, S., Hu, K., & Way, A. (2018). Attaining the unattainable? reassessing claims of human parity in neural machine translation. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 113–123. https://doi.org/10.18653/v1/W18-6312

Treviso, M., Góis, A., Fernandes, P., Fonseca, E., & Martins, A. (2022, May). Predicting attention sparsity in transformers. In A. Vlachos, P. Agrawal, A. Martins, G. Lampouras, & C. Lyu (Eds.), *Proceedings of the sixth workshop on structured prediction for nlp* (pp. 67–81). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.spnlp-1.7

Treviso, M., Guerreiro, N. M., Rei, R., & Martins, A. F. T. (2021). IST-unbabel 2021 submission for the explainable quality estimation shared task. *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, 133–145. https://doi.org/10.18653/v1/2021.eval4nlp-1.14

Treviso, M., & Martins, A. F. T. (2020, November). The explanation game: Towards prediction explainability through sparse communication. In A. Alishahi, Y. Belinkov, G. Chrupała, D. Hupkes, Y. Pinter, & H. Sajjad (Eds.), *Proceedings of the third blackboxnlp workshop on analyzing and interpreting neural networks for nlp* (pp. 107–118). Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.blackboxnlp-1.10

Tu, Z., Liu, Y., Shi, S., & Zhang, T. (2018a). Learning to remember translation history with a continuous cache. *Transactions of the Association for Computational Linguistics*, *6*, 407–420. https://doi.org/10.1162/tacl_a_00029

Tu, Z., Liu, Y., Shi, S., & Zhang, T. (2018b). Learning to remember translation history with a continuous cache. *Transactions of the Association for Computational Linguistics*, *6*, 407–420. https://doi.org/10.1162/tacl_a_00029

Turpin, M., Michael, J., Perez, E., & Bowman, S. R. (2023). Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Thirty-seventh Conference on Neural Information Processing Systems*. https://openreview.net/forum?id=bzs4uPLXvi

Vafa, K., Deng, Y., Blei, D., & Rush, A. (2021). Rationales for sequential predictions. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 10314–10332. https://doi.org/10.18653/v1/2021.emnlp-main.807

Van de Pol, J., Volman, M., & Beishuizen, J. (2010). Scaffolding in teacher–student interaction: A decade of research. *Educational psychology review*, *22*(3), 271–296.

Vashishth, S., Upadhyay, S., Tomar, G. S., & Faruqui, M. (2019). Attention interpretability across nlp tasks. *ArXiv*, *abs/1909.11218*. https://api.semanticscholar.org/CorpusID:202750077

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017a). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 30). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017b). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in neural information processing systems* (Vol. 30). Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Vilar, D., Freitag, M., Cherry, C., Luo, J., Ratnakar, V., & Foster, G. (2022). Prompting palm for translation: Assessing strategies and performance.

Voita, E., Sennrich, R., & Titov, I. (2019). When a good translation is wrong in context: Context-aware machine translation improves on deixis, ellipsis, and lexical cohesion. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1198–1212. https://doi.org/10.18653/v1/P19-1116

Wang, C., & Sennrich, R. (2020). On exposure bias, hallucination and domain shift in neural machine translation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 3544–3552. https://doi.org/10.18653/v1/2020.acl-main.326

Wang, L., Tu, Z., Way, A., & Liu, Q. (2017). Exploiting cross-sentence context for neural machine translation. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 2826–2831. https://doi.org/10.18653/v1/D17-1301

Wang, W., Watanabe, T., Hughes, M., Nakagawa, T., & Chelba, C. (2018). Denoising neural machine translation training with trusted data and online data selection. *Proceedings of the Third Conference on Machine Translation: Research Papers*, 133–143. https://doi.org/10.18653/v1/W18-6314

Wang, X., Tsvetkov, Y., & Neubig, G. (2020). Balancing training for multilingual neural machine translation. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8526–8537. https://doi.org/10.18653/v1/2020.acl-main.754

Wang, Z., Tsvetkov, Y., Firat, O., & Cao, Y. (2021). Gradient vaccine: Investigating and improving multi-task optimization in massively multilingual models. *International Conference on Learning Representations*. https://openreview.net/forum?id=F1vEjWK-lH_

Wei, J., Bosma, M., Zhao, V., Guu, K., Yu, A. W., Lester, B., Du, N., Dai, A. M., & Le, Q. V. (2021). Finetuned language models are zero-shot learners. *International Conference on Learning Representations*.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., ichter brian, b., Xia, F., Chi, E., Le, Q. V., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, & A. Oh (Eds.), *Advances in neural information processing systems* (pp. 24824–24837, Vol. 35). Curran Associates, Inc. https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf

Wiegreffe, S., & Pinter, Y. (2019, November). Attention is not not explanation. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (emnlp-ijcnlp)* (pp. 11–20). Association for Computational Linguistics. https://doi.org/10.18653/v1/D19-1002

Wieting, J., Berg-Kirkpatrick, T., Gimpel, K., & Neubig, G. (2019). Beyond BLEU:training neural machine translation with semantic similarity. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4344–4355. https://doi.org/10.18653/v1/P19-1427

Williams, R. J. (1992). Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, *8*(3–4), 229–256. https://doi.org/10.1007/BF00992696

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., … Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 38–45. https://www.aclweb.org/anthology/2020.emnlp-demos.6

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., & Schmidt, L. (2022). Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. https://doi.org/10.48550/ARXIV.2203.05482

Wu, I., Fernandes, P., Bertsch, A., Kim, S., Pakazad, S., & Neubig, G. (2025). Better instruction-following through minimum bayes risk. *International Conference on Learning Representations ICLR*. https://openreview.net/forum?id=7xCSK9BLPy

Wu, T., Ribeiro, M. T., Heer, J., & Weld, D. (2021). Polyjuice: Generating counterfactuals for explaining, evaluating, and improving models. *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)*, 6707–6723. https://doi.org/10.18653/v1/2021.acl-long.523

Xin, D., Ghorbani, B., Garg, A., Firat, O., & Gilmer, J. (2022). Do current multi-task optimization methods in deep learning even help? *Advances in neural information processing systems*.

Xiong, R., Yang, Y., He, D., Zheng, K., Zheng, S., Xing, C., Zhang, H., Lan, Y., Wang, L., & Liu, T. (2020). On layer normalization in the transformer architecture. *International Conference on Machine Learning*, 10524–10533.

Xu, W., Deutsch, D., Finkelstein, M., Juraska, J., Zhang, B., Liu, Z., Wang, W. Y., Li, L., & Freitag, M. (2023). Pinpoint, not criticize: Refining large language models via fine-grained actionable feedback.

Xu, W., Wang, D., Pan, L., Song, Z., Freitag, M., Wang, W., & Li, L. (2023, December). INSTRUCTSCORE: Towards explainable text generation evaluation with automatic feedback. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 conference on empirical methods in natural language processing* (pp. 5967–5994). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.365

Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., & Raffel, C. (2021, June). MT5: A massively multilingual pre-trained text-to-text transformer. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, & Y. Zhou (Eds.), *Proceedings of the*

2021 conference of the north american chapter of the association for computational linguistics: Human language technologies (pp. 483–498). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.41

Yan, B., Fernandes, P., Dalmia, S., Shi, J., Peng, Y., Berrebbi, D., Wang, X., Neubig, G., & Watanabe, S. (2022, May). CMU's IWSLT 2022 dialect speech translation system. In E. Salesky, M. Federico, & M. Costa-jussà (Eds.), *Proceedings of the 19th international conference on spoken language translation (iwslt 2022)* (pp. 298–307). Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.iwslt-1.27

Yan, B., Shi, J., Tang, Y., Inaguma, H., Peng, Y., Dalmia, S., Polák, P., Fernandes, P., Berrebbi, D., Hayashi, T., Zhang, X., Ni, Z., Hira, M., Maiti, S., Pino, J., & Watanabe, S. (2023, July). ESPnet-ST-v2: Multipurpose spoken language translation toolkit. In D. Bollegala, R. Huang, & A. Ritter (Eds.), *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 3: System demonstrations)* (pp. 400–411). Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-demo.38

Yang, Y., Huang, L., & Ma, M. (2018). Breaking the beam search curse: A study of (re-)scoring methods and stopping criteria for neural machine translation. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3054–3059. https://doi.org/10.18653/v1/D18-1342

Yee, K., Dauphin, Y., & Auli, M. (2019). Simple and effective noisy channel modeling for neural machine translation. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5696–5701. https://doi.org/10.18653/v1/D19-1571

Yin, K., Fernandes, P., Pruthi, D., Chaudhary, A., Martins, A. F. T., & Neubig, G. (2021, August). Do context-aware translation models pay the right attention? In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *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)* (pp. 788–801). Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.acl-long.65

Yu, L., Blunsom, P., Dyer, C., Grefenstette, E., & Kociský, T. (2017). The neural noisy channel. *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. https://openreview.net/forum?id=SJ25-B5eg

Zerva, C., Blain, F., Rei, R., Lertvittayakumjorn, P., C. de Souza, J. G., Eger, S., Kanojia, D., Alves, D., Orăsan, C., Fomicheva, M., Martins, A. F. T., & Specia, L. (2022). Findings of the WMT 2022 shared task on quality estimation. *Proceedings of the Seventh Conference on Machine Translation (WMT)*, 69–99. https://aclanthology.org/2022.wmt-1.3

Zerva, C., van Stigt, D., Rei, R., Farinha, A. C., Ramos, P., C. de Souza, J. G., Glushkova, T., Vera, M., Kepler, F., & Martins, A. F. T. (2021a). IST-unbabel 2021 submission for the quality estimation shared task. *Proceedings of the Sixth Conference on Machine Translation*, 961–972. https://aclanthology.org/2021.wmt-1.102

Zerva, C., van Stigt, D., Rei, R., Farinha, A. C., Ramos, P., C. de Souza, J. G., Glushkova, T., Vera, M., Kepler, F., & Martins, A. F. T. (2021b). IST-unbabel 2021 submission for the quality estimation shared task. *Proceedings of the Sixth Conference on Machine Translation*, 961–972. https://aclanthology.org/2021.wmt-1.102

Zhai, X., Kolesnikov, A., Houlsby, N., & Beyer, L. (2022). Scaling vision transformers. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 12104–12113.

Zhang, B., Ghorbani, B., Bapna, A., Cheng, Y., Garcia, X., Shen, J., & Firat, O. (2022). Examining scaling and transfer of language model architectures for machine translation. *arXiv preprint arXiv:2202.00528*.

Zhang, D., Crego, J., & Senellart, J. (2018, October). Analyzing knowledge distillation in neural machine translation. In M. Turchi, J. Niehues, & M. Frederico (Eds.), *Proceedings of the 15th international conference on spoken language translation* (pp. 23–30). International Conference on Spoken Language Translation. https://aclanthology.org/2018.iwslt-1.4/

Zhang, J., Luan, H., Sun, M., Zhai, F., Xu, J., Zhang, M., & Liu, Y. (2018). Improving the transformer translation model with document-level context. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 533–542. https://doi.org/10.18653/v1/D18-1049

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., & Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. *International Conference on Learning Representations*. https://openreview.net/forum?id=SkeHuCVFDr

Zhou, W., Xu, C., & McAuley, J. (2021). Meta learning for knowledge distillation. *arXiv preprint arXiv:2106.04570*.

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., Christiano, P. F., & Irving, G. (2019). Fine-tuning language models from human preferences. *CoRR*, *abs/1909.08593*. http://arxiv.org/abs/1909.08593

Zouhar, V., Chen, P., Lam, T. K., Moghe, N., & Haddow, B. (2024, November). Pitfalls and outlooks in using COMET. In B. Haddow, T. Kocmi, P. Koehn, & C. Monz (Eds.), *Proceedings of the ninth conference on machine translation* (pp. 1272–1288). Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.wmt-1.121