UNIVERSITAT DE BARCELONA

FUNDAMENTAL PRINCIPLES OF DATA SCIENCE MASTER'S THESIS

Machine Translation evaluation metrics benchmarking: From traditional MT to LLMs

Author:
Álvaro López

Supervisor:
Prof. Jordi Vitrià

A thesis submitted in partial fulfillment of the requirements
for the degree of MSc in Fundamental Principles of Data Science

in the

Facultat de Matemàtiques i Informàtica

June 30, 2023

UNIVERSITAT DE BARCELONA

Abstract

Facultat de Matemàtiques i Informàtica

MSc

Machine Translation evaluation metrics benchmarking: From traditional MT to LLMs

by Álvaro López

This thesis endeavors to cast a spotlight on the evolution and applicability of machine translation (MT) evaluation metrics and models, mainly contrasting statistical methods against the more contemporary neural-based ones, where we also give special attention to the exciting modern Large Language Models (LLMs). MT, a significant area in Natural Language Processing (NLP), has seen a vast metamorphosis over the years, bringing into focus the critical need for thorough exploration of these evolving systems.

Our research is anchored on the Digital Corpus of the European Parliament (DCEP), a complex and multilingual corpus that makes it an ideal testbed to benchmark MT models given its comprehensive and diversified linguistic data. Through the use of this extensive corpus, we aim to present a comprehensive benchmarking of various selected MT models, encapsulating not just their evolution but also their performance dynamics across different tasks and contexts.

A vital facet of our study includes evaluating the relevance and reliability of various MT metrics, such as the old BLEU, METEOR, CHRF, along with newer neural-based metrics which promise to capture semantics more effectively. We aim to uncover the inherent strengths and limitations of these metrics, consequently guiding the choice of appropriate metrics for specific MT contexts for future practitioners and researchers.

In this holistic examination, we will also propose to analyze the interplay between model selection, evaluation metric, and translation quality. This thesis will provide a novel lens to understand the idiosyncrasies of various popular MT models and evaluation metrics, ultimately contributing to more effective and nuanced applications of MT.

In sum, this exploration promises to furnish a new perspective on MT evaluation, honing our understanding of both the models' and metrics' evolutionary paths, and providing insights into their contextual performance on the DCEP corpus, creating a benchmark that can serve the broader MT community. The insights derived aim to significantly contribute to the latter.

The reader can find all the code, used for the text pre/postprocessing and evaluation of the models and metrics at play along with other intermediate matters, published publicly in our *GitHub repository*.

Acknowledgements

I would like to express my heartfelt gratitude to my UB supervisor, Jordi Vitrià, for his invaluable support and guidance throughout my Master's Degree journey. His expertise and help have been instrumental in this academic journey.

I am deeply grateful to my parents, brother and girlfriend for their constant love, encouragement, and belief in my abilities. Their unwavering support has been the foundation of my success. The same goes for my friends in Barcelona and Granada for their companionship and all those memorable moments.

Finally, I would also like to thank ServiZurich and all the great and warm people that works there for providing me with the opportunity to delve into these exciting new technologies. I am grateful for the practical experience and knowledge gained during my time with the organization and for the amazing friendships I've been able to make there.

Contents

Chapter 1

Evolution and state of the art

1.1 Machine translation models

Machine translation (MT) has seen tremendous evolution since its inception. From rule-based systems to modern neural approaches, the journey of machine translation is a testament to the advances in AI and computational linguistics.

1.1.1 Traditional Machine Translation

The initial models for machine translation were rule-based systems, also known as **Rule-Based Machine Translation (RBMT)**. This approach began in the 1950s and extended until the 1980s. It relied heavily on the linguist to define grammar and vocabulary rules for the source and target languages. The system would then use these rules to translate from one language to another.

There were two main types of RBMT:

- Direct Translation: This was the simplest form, mainly used for closely related languages. It would translate word-by-word, often leading to inaccurate and unnatural translations due to differences in syntax and grammar.

- Transfer-based Translation: This method was more sophisticated, involving a three-step process of analyzing the source language, transferring the meaning, and then generating the target language. It required an extensive knowledge base and rules for each language pair, making it resource-intensive and limited to specific language pairs.

  Rule-Based Machine Translation (RBMT) systems, the first generation of MT systems, were heavily dependent on linguistic rules and dictionaries. While there are numerous contributions to this field, one can refer to a paper like "The first public demonstration of machine translation: the Georgetown-IBM system, 7th January 1954"(Hutchins, 2004) for historical perspective. This paper highlights one of the earliest endeavors in machine translation which was, indeed, largely rule-based.

1.1.2 Statistical Machine Translation (SMT)

From the late 1980s onwards, a new approach known as **Statistical Machine Translation (SMT)** started gaining attention. SMT used statistical models to generate translations, based on the analysis of bilingual text corpora and thanks to the advent of more computational power. Unlike RBMT, SMT did not need predefined grammatical rules and could learn them from the data.

Models such as **Bilingual Dictionary Induction (BLI)** were some of the most simple methods that are described as SMT. In this case, you're statistically inferring the alignment between words in two languages based on the proximity of their embeddings in the vector space (you can model the transformation from one language vector embedding space to the other as a linear map and then perform a KNN or approximated KNN). Obviously, this neglects any kind of context.

The most popular SMT approach was **Phrase-Based Machine Translation (PBMT)**, introduced in the paper "Statistical Phrase-Based Translation."(Koehn, Och, and Marcu, 2003). Instead of translating word-by-word, PBMT would translate whole phrases, which could be more than one word. This made translations more context-aware and natural-sounding.

However, SMT had its shortcomings, including difficulty dealing with long sentences due to its inability to retain long-term dependencies and its limited capacity to handle the nuances of natural language.

1.1.3 Neural Machine Translation (NMT)

Around the mid-2010s, the emergence of **Deep Learning (DL)** brought about a new booming era for many branches of Data Science and AI. To name a few: The invention of the dropout mechanism as a way to combat overfitting (Srivastava et al., 2014), the creation of the ADAM stochastic optimization algorithm (Kingma and Ba, 2014), the birth and success of (Deep) Reinforcement Learning (Mnih et al., 2013) and GANs (Goodfellow et al., 2014), as well as the advancement of Image Classification with the Deep CNNs (Krizhevsky, Sutskever, and Hinton, 2012).

Of course, the DL revolution led to a new era in machine translation: **Neural Machine Translation (NMT)**. NMT models use neural networks to predict the likelihood of a sequence of words, typically modeling entire sentences in a single integrated model. They are capable of learning complex patterns and capturing long-distance dependencies in the text.

- **Recurrent Neural Network (RNN)-based NMT**: The initial NMT models were based on recurrent neural networks, specifically the **Long Short-Term Memory (LSTM)** architecture. LSTM helped mitigate the vanishing gradient problem of traditional RNNs, allowing the model to learn longer sequences and dependencies.

  The **Sequence-to-Sequence (Seq2Seq)** model (Sutskever, Vinyals, and Le, 2014) was a popular application of RNNs in NMT, where one RNN (encoder) would encode the source sentence into a context vector, and another RNN (decoder) would decode that vector into the target sentence.

  The addition of an **attention mechanism** improved this approach by allowing the model to focus on different parts of the input sequence when generating each word in the output sequence, greatly improving the quality of translations.

- **Transformer-based NMT**: Transformers, introduced in the paper "Attention Is All You Need"(Vaswani et al., 2017), marked a significantly huge advance in NMT. They replaced recurrence with **self-attention** and positionally-encoded inputs, allowing parallelization and capturing dependencies irrespective of their distance in the sentence. In other words, they leverage self-attention mechanisms and avoid recurrence entirely, leading to significantly improved efficiency. This architecture is the basis of most state-of-the-art MT systems today and we will explain it in with more detail in the next chapter along with the specific models we will use.

Large language models (LLMs), mostly based on the Transformer architecture, have shown remarkable performance across a range of tasks, including translation. Pretraining a language model on a large corpus of text and then fine-tuning it on a specific task (including machine translation) has shown impressive results. This has led to some "early" successful examples such as:

- **BERT (Bidirectional Encoder Representations from Transformers)**: BERT (Devlin et al., 2019) marked a shift in NLP by using bidirectional transformers and a masked language model training objective. It learns to predict words in a sentence given the context from both sides.
- **GPT (Generative Pretrained Transformer)** models: OpenAI's GPT "Improving language understanding by generative pre-training"(Radford et al., 2018) and its follow-up paper on GPT-2 "Language models are unsupervised multitask learners"(Radford et al., 2019) outline the development and capabilities of the GPT models. These models are trained to predict the next word in a sentence, allowing them to generate remarkably human-like text. Unlike Bert, GPT is a left-to-right model (generates text from left to right, one token at a time). That is, a sequential model that processes the input text in a left-to-right fashion, predicting the next token based on the preceding context and it is primarily used for generation tasks.
- **T5 (Text-to-Text Transfer Transformer)**: The T5 model (Raffel et al., 2019), developed by Google, reframes all NLP tasks as a text generation problem and has been shown to perform well on a variety of tasks, including translation. The T5 architecture is also based on the Transformer model, specifically the "encoder-decoder" structure which is commonly used in many NLP tasks. However, unlike traditional usage, T5 uses the encoder-decoder structure even for tasks that are typically solved with an encoder or decoder alone (which is the case for the other LLMs mentioned).

1.1.4 State of the art

SOTA models/techniques

The current state-of-the-art in machine translation includes advanced transformer models and various techniques that enhance their capabilities. These include:

- **Multilingual NMT**: These models can handle multiple languages, often in a many-to-many fashion. They have been shown to improve translation quality, particularly for low-resource languages, by learning to share information between different languages. Facebook AI introduced the "M2M-100 model" (Fan et al., 2020), which can directly translate between 100 languages without using English as a pivot.

- **Zero-shot Translation**: This refers to translating between language pairs that the model has never seen during training. This is often achieved with multilingual models, and although it's not perfect, the results are surprisingly good and getting better.

- **Fine-tuning**: Techniques like transfer learning and fine-tuning are commonly used to adapt pre-trained models to specific translation tasks, often improving performance substantially. They are significant aspects of Large Language

Models (LLMs) like OpenAI's GPT-3. These models are first pre-trained on a large corpus of text data and then fine-tuned for a specific task such as translation. The paper "Language Models are Few-Shot Learners"(Brown et al., 2020) provides insights into the capabilities of GPT-3 and how it can be used for various NLP tasks, including translation, with minimal task-specific training data.

- **Domain-specific Models**: For certain professional fields like law, medicine, or engineering, domain-specific models are trained on specialized datasets to handle the complex jargon and structures used in these areas.
- **Quality Estimation**: An emerging area of research involves predicting the quality of machine translations automatically. This can help users understand the reliability of a translated text and has various applications.

Private Industry Models

Even though **ChatGPT** (one of the models we will use for the berchmarking), the successor of **InstructGPT** (Ouyang et al., 2022), can still be pretty competitive with commercial translation products on high-resource European languages such as German/English, it still falls behind on low-resource or distant languages (Liu et al., 2023; Hendy et al., 2023). Its zero-shot translation capabilities are still impressive, nonetheless. Other GPT-3.5 series perform similarly but at least are publicly available for fine-tuning under payment, unlike ChatGPT/gpt3.5-turbo (its engine).

Here, see "Findings of the WMT 2022 Shared Task on Translation Suggestion"(Yang et al., 2022), we encounter a huge paywall in the development of state-of-the-art NMT models due to the scale of the latter and the comercial interests at play: The Google translator, DeepL translator, Microsoft translator (Azure cognitive translator), Tencent translator...

For example, in this last paper of the World Machine Translation (WMT) 2022 event on the second edition of the shared task on chat translation: "The joint submission of Beijing Jiaotong University and WeChat achieved state of the art on the specific test set with an **ensemble of deep Transformer models** with 20 layers of encoder and 10 layers of decoder. Their models are firstly trained on the training corpora provided by the general track of WMT 2022. They are then fine-tuned on the training data of the chat translation track of WMT 2020 with several strategies to incorporate the potential context including the multi-encoder framework, speaker tag, and prompt-based fine-tuning (otherwise known as **prompt engineering**, useful for **few-shot learning**, where the model is provided with a few example inputs and outputs during inference to help it understand the task. By designing the prompts used in these examples carefully, we can often significantly improve performance). Regarding the size of their models, the authors report numbers that vary from 6.075 Billion to 6.881 Billion parameters."

The open-source and/or public versus private war in the LLM (and by extension, the NMT) world is being waged right now and no one is sure but there are two clear facts to consider:

1. Industry races ahead of academia, precisely since the DL boom. Thus, research tends to become private at least in the short-term (see Figure 1.1: "Until 2014, most significant machine learning models were released by academia. Since then, industry has taken over. In 2022, there were 32 significant industry-produced machine learning models compared to just three produced by academia.

Building state-of-the-art AI systems increasingly requires large amounts of data, computer power, and money—resources that industry actors inherently possess in greater amounts compared to nonprofits and academia.").

2. LLMs are getting bigger and more expensive and this is a huge threshold for many research teams without the appropiate resources (see Figure 1.2: "GPT-2, released in 2019, considered by many to be the first large language model, had 1.5 billion parameters and cost an estimated $50,000 USD to train. PaLM, one of the flagship large language models launched in 2022, had 540 billion parameters and cost an estimated $8 million USD—PaLM was around 360 times larger than GPT-2 and cost 160 times more. It's not just PaLM: Across the board, large language and multimodal models are becoming larger and pricier.").

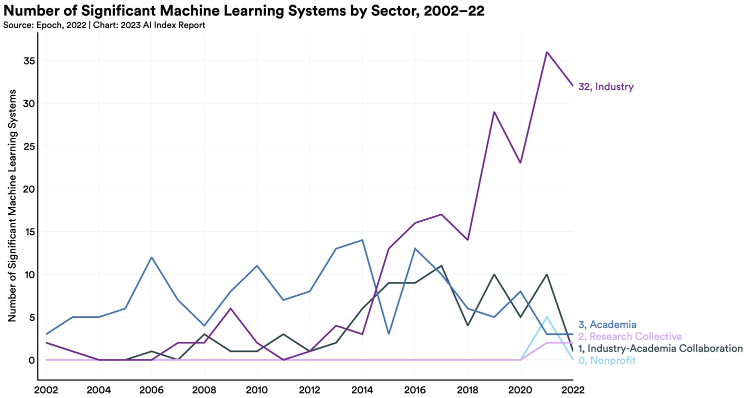

Number of Significant Machine Learning Systems by Sector, 2002–22
Source: Epoch, 2022 | Chart: 2023 AI Index Report

FIGURE 1.1: Extracted from *AI Index Report, Chapter 1: Research and Development*.

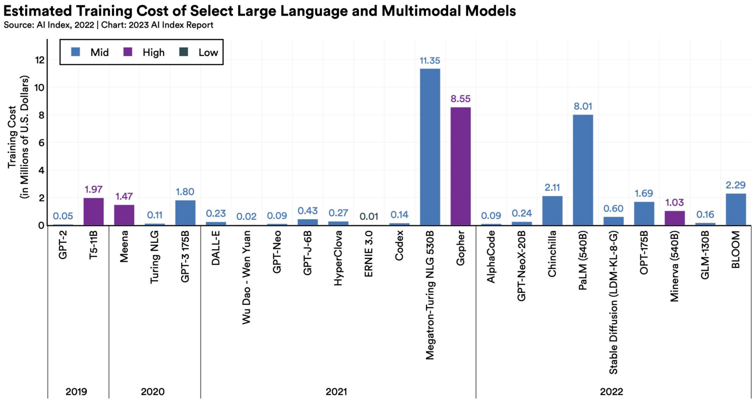

Estimated Training Cost of Select Large Language and Multimodal Models
Source: AI Index, 2022 | Chart: 2023 AI Index Report

FIGURE 1.2: Extracted from *AI Index Report, Chapter 1: Research and Development*.

1.2 Machine translation evaluation metrics

The evolution of machine translation (MT) evaluation metrics follows the evolution of MT systems themselves, transitioning from simple statistical models to complex neural models. The major goal has always been to correlate as much as possible with human judgment.

1.2.1 Automatic Metrics development

If we go back in time, an assessment using automated metrics offers several huge advantages over human-conducted evaluations, namely speed, reproducibility, and cost-effectiveness. This is particularly relevant when evaluating machine translation systems. For human evaluations, the ideal scenario would involve proficient translators, but sourcing such experts for many language pairs poses significant challenges due to their scarcity.

The requirement for large-scale and rapid manual evaluations, essential for assessing new systems in the rapidly evolving field of machine translation, often proves impractical. Consequently, automatic evaluation methods have gained prominence as a highly active and productive research area for over two decades.

Although BLEU (Papineni et al., 2002) remains the most widely used evaluation metric, numerous superior alternatives have emerged, rendering BLEU somewhat antiquated. Since 2010, researchers have proposed more than 100 automatic metrics aimed at enhancing machine translation evaluation.

I will present the most popular metrics that serve as viable alternatives or complementary measures to BLEU and are SOTA. I will however, reserve the in-depth explanation for some of them as I will be providing it in the next chapter since we will use them in our work.

These metrics are categorized as either traditional or neural, each offering distinct advantages.

The majority of automatic metrics used for evaluating machine translation typically require the following:

- The **translation hypothesis** generated by the machine translation system for evaluation.
- At least one **reference translation** produced by humans.
- Occasionally (mostly for some neural SOTA metrics as we will see), the **source text** translated by the machine translation system.

Both the translation hypothesis and the reference translation are translations of the same source text.

The objective of an automatic metric is to provide a score that represents the proximity between the translation hypothesis and the reference translation. A smaller distance indicates that the system's translation is closer to **human quality**.

Typically, the absolute score returned by a metric alone lacks interpretability. It is primarily utilized for ranking machine translation systems, where a better score denotes a superior system.

In the paper "Scientific Credibility of Machine Translation Research: A Meta-Evaluation of 769 Papers"(Marie, Fujita, and Rubino, 2021) (Marie et al., 2021), it was observed that "nearly **99% of research papers on machine translation rely on the automatic metric BLEU** to evaluate translation quality and rank systems (see Figure 1.3). However, over the past 12 years, more than 100 alternative metrics have been proposed." It's worth noting that this analysis focused only on research papers

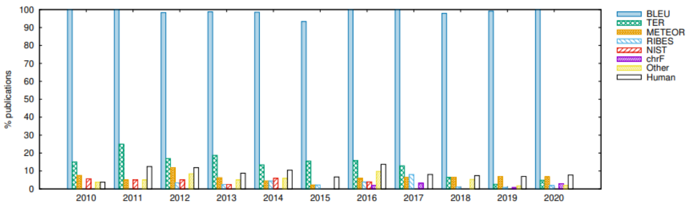


FIGURE 1.3: "Percentage of papers using each evaluation metric per year. Metrics displayed are used in more than five papers." Extracted from (Marie, Fujita, and Rubino, 2021).

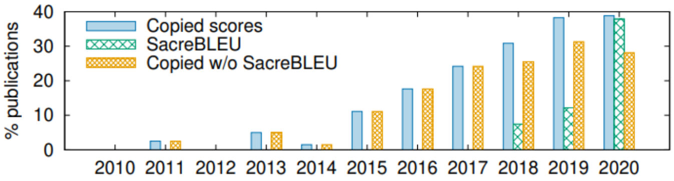


FIGURE 1.4: "Percentage of papers copying scores from previous work (Copied scores), using SacreBLEU (SacreBLEU), and copying scores without using SacreBLEU (“Copied w/o SacreBLEU”)" Extracted from (Marie, Fujita, and Rubino, 2021).

published from 2010 by the ACL, suggesting that even more metrics may exist for evaluating machine translation.

Many of those aforementioned metrics have demonstrated superior performance compared to BLEU, but they have not been widely utilized. In fact, only two of these metrics (RIBES and chrF) have been employed in more than two research publications, out of the 700+ publications analyzed. Despite the availability of newer metrics, the most frequently used metrics since 2010 are those proposed prior to 2010, namely BLEU, TER, and METEOR.

Moreover, as pointed out by Figure , "An MT paper may compare the automatic metric scores of proposed MT systems with the scores reported in previous work. This practice has the advantage to save the time and cost of reproducing competing methods ... **Copying scores (mostly BLEU)** from previous work was rarely done before 2015 but in 2019 and 2020 nearly 40% of the papers reported on comparisons with scores from other papers. While many papers copied and compared metric scores across papers, it is often unclear whether they are actually comparable."

Anyways, most of the metrics created after 2016 are **neural metrics** that rely on neural networks and the most recent ones even rely on the very popular pre-trained LLMs. In contrast, **traditional metrics** published earlier can be more simple and cheaper to run. They remain extremely popular for various reasons, and this popularity doesn’t seem to decline, at least in the research domain.

Let’s now present some of those metrics. Those being the most popular, original and showing of their correlation with human evaluation (don’t forget that we are reserving the detailed explanations for some of them in the next chapter).

1.2.2 Traditional metrics

Traditional metrics for evaluating machine translation are designed to measure the similarity between two strings based solely on the characters they comprise. These strings are typically the translation hypothesis and the reference translation. It is worth noting that traditional metrics do not take advantage of the source text that was translated by the system.

One widely used traditional metric, **WER (Word Error Rate)**, served as the precursor to BLEU, which eventually gained prominence in the early 2000s. WER is the minimum number of operations needed to transform a system output into a reference. **PER (Position-independent word Error Rate)** is a variant of WER which disregards word order. However, both WER and PER are not very suitable for MT evaluation because of their insensitivity to minor changes that can significantly alter translation quality.

However, traditional metrics do have some advantages:

- **Low computational cost**: Traditional metrics leverage efficient string-matching algorithms operating at the character and/or token levels. While certain metrics may require token shifting, which can be more computationally intensive, their calculations are easily parallelizable and do not necessitate GPU usage.
- **Explainable**: Computing scores for traditional metrics is generally straightforward, even for small segments, facilitating analysis. However, it is important to note that "explainable" does not equate to "interpretable." While we can precisely explain how a metric score is derived, the score alone usually does not provide meaningful insights into the translation quality.
- **Language independent**: With a few exceptions, most traditional metric algorithms can be applied regardless of the language being translated.

While the disadvantages are clear:

- **Poor correlation with human judgments**: The primary drawback of traditional metrics lies in their limited ability to align with human evaluations. To obtain the most accurate assessment of translation quality, traditional metrics should not be solely relied upon.
- **Require specific preprocessing**: Except for the chrF metric, all the traditional metrics discussed will necessitate tokenized evaluated segments and their corresponding reference translations. The tokenization process is not integrated into the metric itself and must be performed separately using external tools. Consequently, the obtained scores are dependent on a particular tokenization, which may not be reproducible.

Evolution and examples

1. **BLEU (Bilingual Evaluation Understudy)**: This is the most popular metric. It is used by almost 99% of machine translation research publications. Introduced in “BLEU: a Method for Automatic Evaluation of Machine Translation”(Papineni et al., 2002), it was a major advancement in MT evaluation. BLEU compares n-grams of the machine-generated text (translation hypothesis) to that of a human-generated reference text and returns a score between 0 and 1. We won’t analyze it any further for now.

2. **NIST**: The NIST metric is an extension of BLEU, developed around 2002. It uses a similar n-gram precision method but introduces a few improvements. It gives more weight to less frequent n-grams and includes a brevity penalty like BLEU but calculates it differently. NIST is more computationally intensive but sometimes preferred due to its more nuanced approach.

3. **METEOR**: METEOR, proposed in "METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments"(Lavie and Agarwal, 2007), extends upon BLEU and NIST. It introduces recall into the equation and includes various levels of linguistic analysis (stemming, synonymy, and paraphrasing). METEOR shows a stronger correlation with human judgment but is more computationally intensive.

4. **RIBES (Rank-based Intuitive Bilingual Evaluation Score):** It was introduced in "Automatic Evaluation of Translation Quality for Distant Language Pairs"(Isozaki et al., 2010) and it was designed for language pairs with different sentence structures, penalizing incorrect word order.

5. **TER (Translation Edit Rate)**: Introduced in the paper "A study of translation edit rate with targeted human annotation"(Snover et al., 2006). It measures the effort required for human translators to post-edit machine translations. **TERp** incorporates a paraphrase database, and **HTER** compares machine translations to their post-edited versions by humans.

6. **CharacTER**: It is a character-level variant of TER (Wang et al., 2016), normalizing the edit distance by the length of the translation hypothesis. It shows high correlation with human evaluation but is less commonly used as we can expect.

7. **chrF(++)**: "chrF: character n-gram F-score for automatic MT evaluation"(Popovic, 2015) is the second most popular metric for machine translation evaluation and has been shown to better correlate with human judgment than BLEU. chrF is tokenization independent and relies solely on characters, while chrF++ takes word order into account but is tokenization dependent (see next Chapter).

1.2.3 Neural metrics

Neural metrics differ significantly from traditional metrics as they employ neural networks to estimate translation quality scores. The first neural metric, ReVal (Gupta, Orasan, and Genabith, 2015), was introduced in 2015, and since then, new neural metrics have emerged regularly for machine translation evaluation. However, despite their superiority, neural metrics have not gained widespread popularity, with traditional metrics remaining dominant in the research community (although this is starting to change this last year, we must say).

Their advantages:

- **Better correlation with human evaluation**: Neural metrics are considered state-of-the-art in assessing machine translation quality, exhibiting strong alignment with human judgments.

- **No preprocessing is required**: Recent neural metrics, like COMET (Rei et al., 2020) and BLEURT (Sellam, Das, and Parikh, 2020), handle preprocessing internally, eliminating the need for additional tokenization or other preprocessing steps.

- **Better recall**: Neural metrics leverage embeddings to reward translations that closely match the reference, even if not exact. This flexibility allows for recognition of semantically similar translations, unlike traditional metrics that rely on exact matches.
- **Trainable**: Most neural metrics can be fine-tuned with specific training data, enhancing their ability to correlate with human judgments when tailored to a particular use case.

Their disadvantages:

- **High computational cost**: Neural metrics, while not necessarily requiring a GPU, exhibit significantly slower computation times compared to traditional metrics. Metrics relying on large language models may also demand substantial memory. Additionally, statistical significance testing becomes computationally expensive.
- **Unexplainable (interpretability)**: The complex nature of neural models, with millions or billions of parameters, makes it challenging to understand the reasons behind specific metric scores. Efforts to improve the explainability of neural models are actively pursued in research.
- **Difficult to maintain**: Older implementations of neural metrics may become incompatible due to changes in nVidia CUDA or frameworks like (py)Torch and TensorFlow. It raises concerns about the long-term viability and sustainability of neural metrics.
- **Lack of reproducibility**: Neural metrics typically involve numerous hyperparameters, often underspecified in scientific publications. Consequently, reproducing specific scores for a given dataset becomes challenging, hindering reproducibility efforts.

Overall, while neural metrics offer advantages in correlation with human evaluation and require minimal preprocessing, their drawbacks include high computational costs, limited explainability, maintenance challenges, and issues with reproducibility. However, their superiority is unmatched and, as we will discuss later, that's why the landscape is starting to change in this sense.

Evolution and examples

1. **REVAL (Recurrent Embeddings for Validation)**: Introduced in "ReVal: A Simple and Effective Machine Translation Evaluation Metric Based on Recurrent Neural Networks"(Gupta, Orasan, and Genabith, 2015), it is the first neural metric proposed to evaluate machine translation quality. It leveraged Recurrent Neural Networks (RNNs), **LTSM** to be specific, to encode source and target sentences into vector representations, considering the sequence of words rather than just n-grams. The encoded vectors were then compared to give a measure of the translation quality. This marked a major shift in MT evaluation by using neural networks to capture deeper linguistic features. It is now outperformed by more recent metrics.
2. **YiSi**: Proposed by (Lo, 2019), YiSi-1 is a semantic MT evaluation metric that uses **Word2Vec** embeddings to compute cosine similarity between source and

reference sentences. Unlike ReVal, YiSi-1 uses a universal multilingual semantic space, which means that it is not restricted to specific language pairs. It also introduces a novel alignment approach, which is more reliable in assessing semantic similarity.

3. **BERTScore**: Introduced by (Zhang et al., 2019), BERTScore leverages the power of **BERT**, a **transformer-based contextual embedding** model. It computes token-level F1-scores using BERT embeddings, capturing the context of words in sentences and providing more nuanced scoring. BERTScore has been shown to correlate well with human judgment across different tasks and datasets.

4. **BLEURT (BLEU plus a Usable Readability metric Tuned)**: Proposed by Google Research in "BLEURT: Learning Robust Metrics for Text Generation"(Sellam, Das, and Parikh, 2020), BLEURT uses BERT embeddings to learn a metric based on human judgments of translation quality. The model is **trained** to predict human ratings on a large dataset of sentence pairs, leading to a metric that is highly correlated with human judgment.

5. **PRISM (Predictive Ratings and Informed Metric of Semantics)**: Introduced by Facebook AI in "PRISM: Concept-preserving Summarization of Top-K Social Image Search Results"(Seah, S Bhowmick, and Sun, 2015), it highlights the similarity between machine translation and paraphrasing evaluation tasks. The authors argue that the only difference lies in the source language. Prism is trained on a large **multilingual** parallel dataset using a neural machine translation framework.

   During inference, Prism serves as a zero-shot paraphraser to score the similarity between a source text (translation hypothesis) and a target text (reference translation) in the same language. Notably, Prism does not require human evaluation training data or paraphrasing training data, making it advantageous in terms of simplicity and training convenience. The metric's effectiveness appears promising, surpassing many other metrics, including BLEURT.

6. **COMET (Crosslingual Optimized Metric for Evaluation of Translation)**: Proposed by Unbabel in "COMET: A Neural Framework for MT Evaluation"(Rei et al., 2020), COMET (Rei et al., 2020) presents a supervised approach to machine translation evaluation, utilizing a large language model (XLM-RoBERTa) but noting that other models like BERT could also be employed. Unlike many other metrics, COMET **leverages the source sentence**, fine-tuning the language model on a triplet of data comprising the translated source sentence, translation hypothesis, and reference translation (see Figure **??**))

   The metric is trained using human ratings, similar to those used by BLEURT. Notably, COMET offers a simpler training process compared to BLEURT, as it does not require the generation and scoring of synthetic data.

   COMET has been shown to correlate well with human evaluations and to capture semantic meaning, making it a promising tool for MT evaluation (we will be using its newest version COMET-22 in our evaluations).

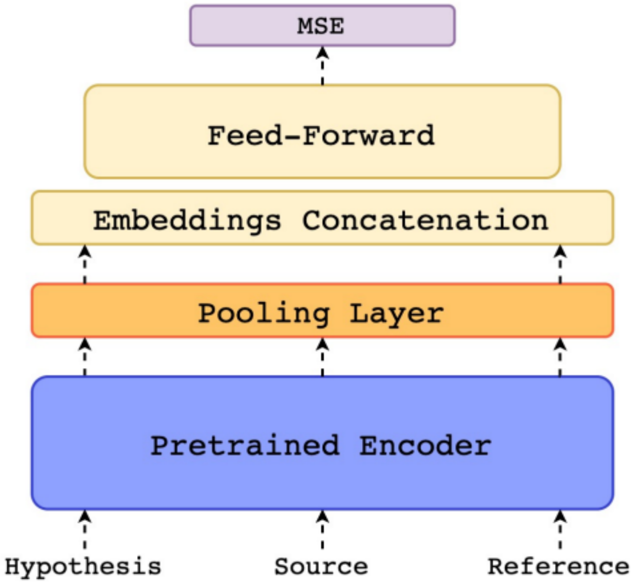


FIGURE 1.5: "Estimator model architecture. The source, hypothesis and reference are independently encoded using a pretrained cross-lingual encoder. The resulting word embeddings are then passed through a pooling layer to create a sentence embedding for each segment. Finally, the resulting sentence embeddings are combined and concatenated into one single vector that is passed to a feed-forward regressor. The entire model is trained by minimizing the Mean Squared Error (MSE)." Extracted from (Rei et al., 2020).

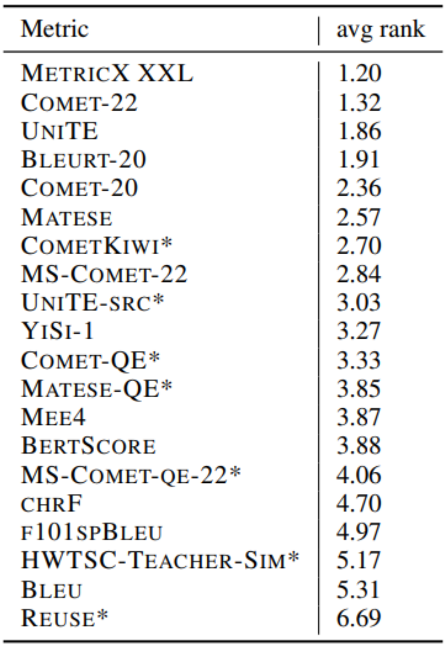
1.2.4 State of the art

Machine translation evaluation is a very active research area. Neural metrics are getting better and more efficient every yearas we've seen in combination of transformer-based powerful models. Yet, traditional metrics such as BLEU remain the favorites of machine translation practitioners, mainly by habits.

The title of this WMT says it all: "Results of WMT22 Metrics Shared Task: Stop Using BLEU - Neural Metrics Are Better and More Robust"(Freitag et al., 2022). It published a ranking of evaluation metrics according to their correlation with human evaluation that we can see in Figure 1.6.

COMET and BLEURT rank at the top while BLEU appears at the bottom and others like METRICX XXL seem to be poorly documented. We will be explaining and using COMET-22 as well as other SOTA metrics we've introduced in this long evolution and SOTA overview of both the models and metrics.

In Chapter 2, we will review the theoretical foundations for some of the concepts that have already come up, before we proceed with the set-up of our benchmarking.



Metric	avg rank
MetricX XXL	1.20
Comet-22	1.32
UniTE	1.86
Bleurt-20	1.91
Comet-20	2.36
Matese	2.57
CometKiwi*	2.70
MS-Comet-22	2.84
UniTE-src*	3.03
YiSi-1	3.27
Comet-QE*	3.33
Matese-QE*	3.85
Mee4	3.87
BERTScore	3.88
MS-Comet-qe-22*	4.06
chrF	4.70
f101spBleu	4.97
HWTSC-Teacher-Sim*	5.17
Bleu	5.31
Reuse*	6.69

Figure 1.6: "Official ranking of all primary submissions of the WMT22 Metric Task. The final score is the weighted average ranking over 201 different scenarios. Metrics with * are reference-free metrics." Extracted from (Freitag et al., 2022).

Chapter 2

Theoretical foundations of our models and metrics

2.1 Attention is all you need

2.1.1 History and Origin

The **Transformer** model was first introduced in the paper "Attention Is All You Need"(Vaswani et al., 2017), published at the Neural Information Processing Systems (NIPS) conference. The authors belonged to the Google Brain team. As we have already been able to seen, the model revolutionized the field of natural language processing (NLP) by introducing the idea that attention mechanisms could entirely replace recurrent networks for sequence modeling tasks.

2.1.2 Concept and Architecture

The main innovation in the Transformer model is the attention mechanism, which has primarily two types: **self-attention** (also known as intra-attention) and **multi-head attention**.

Self-attention allows the model to focus on different words in the input sequence when generating each word in the output sequence, giving it the ability to generate more contextually relevant translations. The model assigns more attention (i.e., weight) to the more important words and less to the less important ones.

Multi-head attention allows the model to focus on information from different positions at the same time, enabling it to capture various aspects of the sentence's structure, like syntax and semantics, at multiple levels of abstraction.

The Transformer model consists of an encoder and a decoder, both of which are composed of multiple identical layers. Each layer has two sub-layers in the encoder and three sub-layers in the decoder.

Here is a simplified diagrammatic representation (see Figure 2.1):

- **Encoder**:
  - Multi-head self-attention mechanism
  - Position-wise fully connected feed-forward network
- **Decoder**:
  - Multi-head self-attention mechanism
  - Multi-head attention over the output of the encoder stack
  - Position-wise fully connected feed-forward network

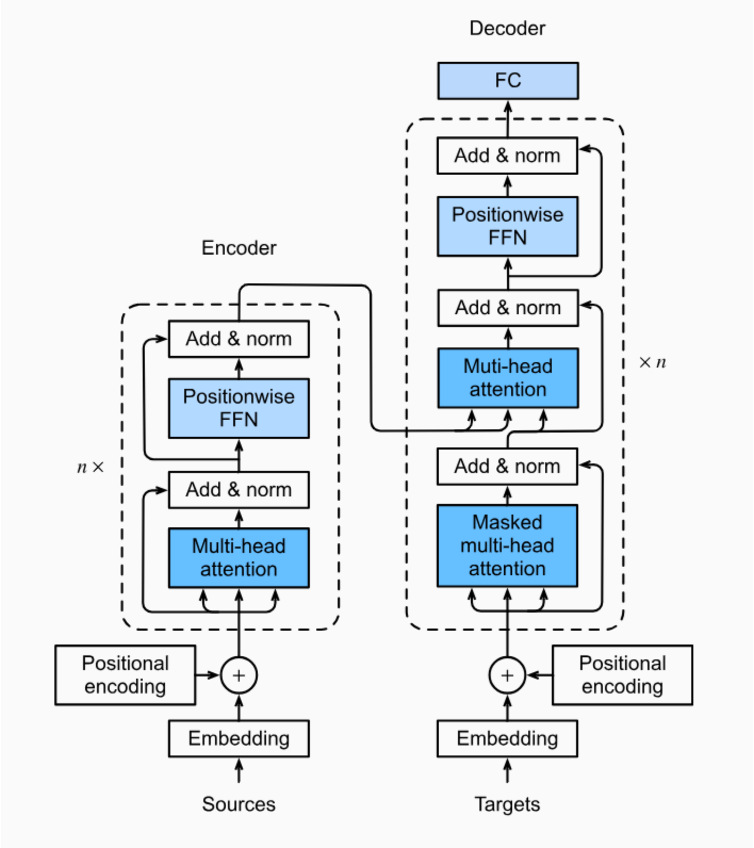


FIGURE 2.1: The transformer architecture. Extracted from *Deep Dive into DL*.

Before the sequences are fed into the encoder and decoder, they pass through an initial embedding layer. This layer transforms the discrete tokens into continuous embeddings, allowing the model to learn a more expressive representation of the inputs.

Both the encoder and decoder are supplemented with position-wise feed-forward networks and residual connections, followed by layer normalization. The feed-forward network consists of two linear transformations with a ReLU activation in between, while the residual connections help in avoiding the vanishing gradient problem.

2.1.3 The layers inside the Transformer

While we won't enter into a complete mathematical description or don't even have the need to in order to understand the key concepts behind Transformers, here are some further details:

1. **Input Embedding**: The input tokens are transformed into vectors through learned embeddings. The vectors are then scaled by a factor of $\sqrt{d_{model}}$, where d_{model} is the dimension of the model. A positional encoding is added to these embeddings to retain the order of the words in the sequence.

2. **Scaled Dot-Product Attention**: The attention score between a query and a key is computed as their dot product, scaled by $\sqrt{d_k}$, where d_k is the dimension of

the key. This score determines how much focus to place on different parts of the input sequence.

3. **Multi-Head Attention**: This concept allows the model to jointly attend to information at different positions from different representational spaces. This is done by performing the scaled dot-product attention in parallel (h times, where h is the number of heads), each with different learned linear transformations of the original queries, keys, and values.

   This way, the attention function can be described as:

   $$Attention(Q, K, V) = softmax(QK^T / \sqrt{d_k})V$$

   where Q is the matrix of queries, K is the matrix of keys, V is the matrix of values, and $'^T'$ denotes transpose. The softmax function ensures that the weights sum to 1, and the $\sqrt{d_k}$ in the denominator is a scaling factor that leads to more stable gradients.

   In practice, the model uses multiple attention heads. As we just commented, for each of these heads, the queries, keys, and values are independently linearly projected h times with learned parameters. This multi-head attention is what allows the model to focus on different types of information.

4. **Feed-Forward Networks**: Each position in the encoder and decoder's input and output is transformed using a separate feed-forward network.

5. **Output Linear layer and Softmax**: The decoder output is projected to the output vocabulary's dimension using a linear layer and softmax function to produce a probability distribution over the output vocabulary.

2.2 Selected Models

Now that we know how Transformers work, we will learn about the types of models we will be using later and their peculiarities.

2.2.1 T5

T5, which stands for "**Text-to-Text Transfer Transformer**," is a model introduced by Google Research in a paper titled *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*(Raffel et al., 2019).

The authors proposed to treat every NLP task as a "text-to-text" problem, and demonstrated that a unified approach can be used across different tasks with competitive performance. This means that tasks like translation, summarization, and question answering are all approached in a similar way: the model is provided with text as input and generates text as output.

Here's a more detailed look at the architecture and the specific components of the T5:

1. **Model Architecture**:

   The underlying architecture of T5 is similar to the original Transformer model we have just reviewed with a slight change: it employs a **denoising autoencoder** structure. The input sequence is corrupted by replacing some tokens

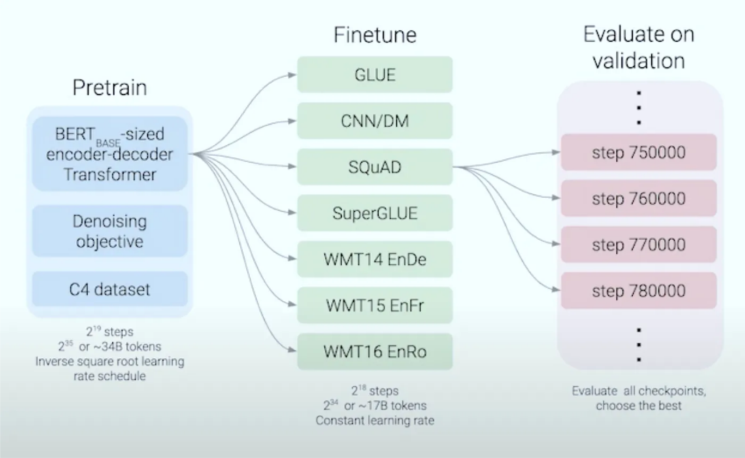


FIGURE 2.2: "The pretraining, fine-tuning, and evaluating steps" Extracted from *Collin Raffel video*.

(15% of the input tokens) with a mask token, and the model then learns to recover the original, uncorrupted text.

The architecture includes an encoder and a decoder, each consisting of a stack of identical layers. Each layer in the encoder has two sub-layers: a multi-head self-attention mechanism, and a position-wise fully connected feed-forward network. The decoder also has two sub-layers like the encoder, plus an additional third sub-layer which performs multi-head attention over the output of the encoder stack.

2. **Task-Specific Prefixes**:

   T5 uses a unique approach to handle different tasks. It introduces task-specific prefixes to the input sequences to guide the model's prediction. For example, if we are doing translation from English to French, the input might start with the text "translate English to French:" followed by the sentence to be translated. Similarly, for summarization, it could be "summarize:" followed by the text to be summarized.

3. **Training Objective**:

   The T5 model is trained with a **Causal Language Modeling (CLM) objective**, meaning it is trained to predict the next token in a sequence given the previous tokens. This objective encourages the model to learn contextual representations of the input tokens. It uses teacher forcing during training, i.e., it uses the true previous tokens in the output sequence as input to the decoder, rather than the tokens it predicted.

4. **Preprocessing and Tokenization**:

   T5 uses **SentencePiece** tokenization, which is a type of subword tokenization method. SentencePiece is a language-independent, data-driven tokenization method that enables the model to handle multiple languages.

5. **Variants**:

   T5 comes in different sizes, just like BERT and GPT-2. The "base" version has 220 million parameters, while the largest, T5-11B, has 11 billion parameters.

In conclusion, T5 represents a significant shift in the way NLP tasks are approached. By treating all tasks as a "text-to-text" problem, T5 simplifies the process of applying transformers to a wide variety of tasks. However, this also means that T5 might need to be fine-tuned more often than task-specific models, which could have implications for training time and computational resources.

In our case, we will be using **FLAN-T5** (base) as the first model in our benchmarking. It was released in “Scaling Instruction-Finetuned Language Models”(Chung et al., 2022) and it is an enhanced version of T5 that has been **finetuned** in a mixture of tasks such as MT and achieves incredible benchmarks without additional complications.

2.2.2 The GPT models

GPT-1

OpenAI introduced the first version of the GPT model (**GPT-1**) in the paper “Improving language understanding by generative pre-training”(Radford et al., 2018), designed to improve upon the fine-tuning approach for transfer learning.

GPT-1 utilizes a transformer model, but unlike the original transformer that uses both an encoder and a decoder, GPT-1 only uses the transformer’s **decoder** mechanism. The architecture (see Figure 2.3) has 12 self-attention layers (transformer blocks), each with 12 attention heads. The dimension of the input embeddings and the number of hidden units in the model are both 768.

The model is trained to predict the next token in a sequence, a task called language modeling. However, GPT-1 uses a version of language modeling called "causal language modeling" or "**autoregressive language modeling**" that we have already seen and where it’s trained to predict the next token given the preceding ones, learning an approximation of the probability distribution of a word given its prior context.

The training process involved two steps: unsupervised pre-training and supervised fine-tuning. During pre-training, GPT-1 learned to predict the next word in a sentence, allowing it to understand the syntax, context, and semantics of language. In the fine-tuning step, GPT-1 was further trained on task-specific datasets (BooksCorpus, containing about 7,000 unpublished books) to adapt it to various NLP tasks such as text classification, text generation, translation, and more.

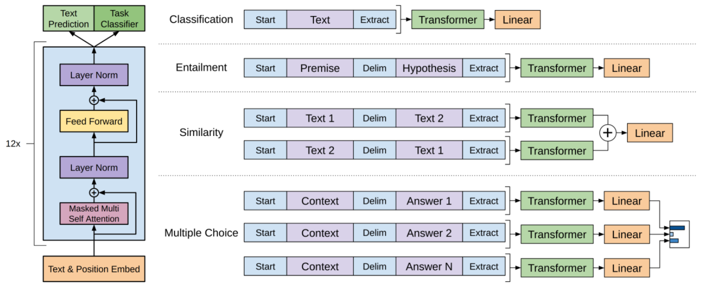


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

FIGURE 2.3: Extracted from Radford et al. (2018).

GPT-2

In the paper "Language models are unsupervised multitask learners"(Radford et al., 2019), OpenAI introduced **GPT-2**, a significant upgrade from GPT-1.

GPT-2 increased the **model's size** to 1.5 billion parameters and was trained on a larger and more diverse dataset, the WebText, which is a subset of the internet. With more layers (48 transformer blocks) and parameters, GPT-2 exhibited improved performance in various NLP tasks.

Importantly, OpenAI also showed that GPT-2 could perform several tasks without task-specific training data, relying solely on the prompts given at inference time. This **zero-shot learning capability** was a significant leap in NLP.

However, due to concerns about potential misuse, OpenAI initially refrained from releasing the full GPT-2 model, marking a critical moment in AI research and its intersection with ethics and society.

GPT-3

GPT-3 was introduced in the paper "Language Models are Few-Shot Learners"(Brown et al., 2020). This version dramatically scaled up the model size to 175 billion parameters, making it one of the **largest language models at the time**.

GPT-3 demonstrated that scaling up language models improves their performance across a wide range of tasks and languages. More interestingly, GPT-3 exhibited the ability to perform tasks in a f**ew-shot learning setup**, where the model can generalize from a handful of examples to perform a task.

The increased size and improved performance allowed GPT-3 to generate impressively coherent and contextually relevant passages of text, understand the sentiment of a text, answer questions, and even translate languages with remarkable accuracy.

The transition from GPT-1 to GPT-3 showcases the principle of **scaling in AI**: bigger models, trained on larger datasets, tend to perform better. However, the transition also highlights the increasing ethical, societal, and technical challenges posed by such large language models.

GPT-3.5 series

GPT-3.5 is based on GPT-3 but works within specific policies of human values and only 1.3 billion parameters fewer than the previous version by 100X. sometimes called **InstructGPT** that trained on the same datasets of GPT-3 but with additional fine tuning process that adds a concept called **'reinforcement learning with human feedback'** or RLHF to the GPT-3 model.

Models referred to as "GPT 3.5" are a series of models that were trained on a blend of text and code from before Q4 2021. The highly famous **ChatGPT** dialogue model is supported by IntructGPT and has very interesting use cases. In Chapter 1, we already saw its incredibles zero-shot capabilities in translation. That's why we will use this LLM later in our work.

2.2.3 Selected metrics

Finally, we are going to review some aspects of the metrics that we will actually choose for our benchmarking and that we already saw (or mentioned in the last case) in the last Chapter (1).

BLEU

It works by comparing a candidate translation to one or multiple reference translations. The BLEU score is then computed based on the **precision of n-grams** (contiguous sequences of n words) present in the candidate translation that also appear in the reference translation(s). The precision is calculated for different values of n (typically up to 4), and a geometric mean is taken of these precision scores to compute the final BLEU score.

Additionally, BLEU includes a **brevity penalty**. If the candidate translation is shorter than the reference(s), the score gets penalized. This penalty ensures that the translation doesn't achieve a high score by simply leaving out content.

Overall, we know how popular and simple it is. It is reproducible and correlates with human evaluation but it has a complete lack of semantic understanding and is too sensitive to exact word matches. Moreover, let's remember that although BLEU correlates reasonably with human evaluation at the corpus level, its performance at the sentence level is much worse.

METEOR

Unlike BLEU which relies solely on precision, METEOR considers both **precision and recall**, using a harmonic mean to combine them. This makes it a more balanced metric that takes into account both under-translation and over-translation.

METEOR also includes several additional features to deal with the limitations of BLEU. For instance, it uses WordNet to identify synonyms and considers them as matches. It also uses stemming to handle different forms of the same word, and includes a module to recognize paraphrases. Additionally, METEOR takes word order into account.

METEOR tends to perform better than BLEU at sentence level although it's more complex.

chrF(++)

The chrF metric calculates the n-gram F-score at the character level, rather than the word level, which makes it sensitive to morphological differences, such as inflections. This is particularly useful in languages with rich morphology. Additionally, it penalizes over- and under-predictions, leading to a more balanced evaluation. By working at the character level, chrF can better handle differences in morphology that word-level metrics might overlook. It is also tokenization-independent.

Similar to BLEU, chrF doesn't consider semantics or context (although we just mentioned in Chapter 1 that it is the second most popular traditional metric for good reasons). That is why we will include the following transformer-based metrics too (we already discussed their advantage, so we will just briefly explain how they function)

BERTScore

BERTScore leverages the pre-trained contextual embeddings from BERT (Bidirectional Encoder Representations from Transformers) and correlates them to human judgment on the system level. It computes the **cosine similarity** between BERT embeddings of the candidate and reference sentences, which allows it to capture meaning beyond simple n-gram overlap.

BLEURT

BLEURT is an evaluation metric that is designed to leverage pre-training, fine-tuning, and transfer learning.

To compute BLEURT scores, first, a base model is pre-trained on a large corpus of text. This pre-training step allows the model to learn a general understanding of language, including grammar, syntax, and semantics. This pre-trained model is similar to the Transformer models used in BERT, GPT-2, etc., but with a smaller size to reduce computational complexity.

The pre-trained model is then fine-tuned on a specific task, which is to predict human judgments on translation quality. For this, Google researchers created a new dataset consisting of millions of sentence-level **translation quality scores**. These scores were collected from human raters judging translations produced by various systems. The model learns to predict these human quality scores, and this learned model becomes the BLEURT scoring system.

In operation, given a candidate translation and a reference, BLEURT tokenizes the texts, computes a series of features such as the number of matching n-grams between the candidate and reference, then feeds these features, along with the tokenized texts, to the fine-tuned model, which outputs the final BLEURT score.

COMET(-22)

COMET, developed by Unbabel AI, also leverages transformer-based architectures to model the quality of translations. It is trained in three stages to provide a well-rounded evaluation of translations.

- **Pre-training**: Just like BLEURT, the base model in COMET is pre-trained on a large-scale multilingual corpus, to learn general language representations.
- **Ranking Fine-tuning**: The model is then fine-tuned on a translation ranking task. Given a source sentence and several candidate translations, the model learns to rank the translations in the order of quality. This helps the model understand what a ’good’ translation looks like in comparison to ’bad’ ones.
- **Quality Estimation Fine-tuning**: The final step is to fine-tune the model on a quality estimation task. For this, the model is trained to predict sentence-level and word-level quality scores collected from human evaluations. This helps the model estimate how ’good’ a translation is in absolute terms, not just relative to other translations.

However, we will be using SOTA **COMET-22** (Rei et al., 2022), presented by joint contribution of Unbabel and IST to the WMT 2022 Metrics Shared Task. Their primary submission – dubbed COMET-22 – is an **ensemble** between a COMET estimator model trained with Direct Assessments and a newly proposed multitask model trained to predict sentence-level scores along with OK/BAD word-level tags derived from Multidimensional Quality Metrics error annotations. These models are ensembled together using a **hyper-parameter search** that weights different features extracted from both evaluation models and combines them into a single score.

Chapter 3

Methodology and goals

3.1 The DCEP Corpus

Before we talk about the motivation for the models involved in the benchmarking or evaluation process and the metrics used, we need to talk about our corpus: **The Digital Corpus of the European Parliament (DCEP)**.

The Digital Corpus of the European Parliament (DCEP) is a multilingual dataset comprising all the documents and proceedings of the European Parliament, which are translated into 23 official languages of the European Union.

- **Content and Structure**:

  The corpus includes diverse types of documents, including agendas, written questions, resolutions, and verbatim reports of debates, among others. It's important to note that these texts are not only parallel in the sense that they are translations of the same content, but they are also **aligned** at the document level, meaning that the same sentence in different languages corresponds to the same sentence in the original language.

- **Applications**:

  DCEP is often used in machine translation, particularly in training and evaluating Statistical Machine Translation (SMT) and Neural Machine Translation (NMT) systems. The large size and multilingual nature of the dataset make it an excellent resource for these purposes.

- Advantages and Disadvantages for Machine Translation Benchmarking:

  Size and Diversity: The DCEP is one of the largest multilingual corpora available, making it ideal for training robust translation models.

  Quality of Translations: The translations in the DCEP are conducted by professional human translators, ensuring high quality.

  Domain Specificity: The DCEP is specific to the legislative and political domain. Therefore, models trained exclusively on this dataset might not generalize well to texts from other domains.

  Formality and Style: The language used in parliamentary documents tends to be **formal** and may not represent the variability of language use in less formal contexts. This could limit the ability of models trained on this data to handle more informal or colloquial language.

Although these two last facts will not affect our trained models, the style will play a role in the evaluation of said models.

The huge size of the dataset and since the preprocessing step was outdated, led us to choose a reduced version of a German-English dataset which was **numerically filtered** in order to do sentence-level translations (see the *GitHub repository* for a more thorough analysis).

3.2 Models and metrics selected

We will work with **ChatGPT** using its recently public API (with the gpt-3.5-turbo engine) and this way we leverage the fact that the translations are at sentence-level and that German is a **high resource language** so the capabilities of few-shot learning that we mentioned in Chapter 1, present in papers “Summary of ChatGPT/GPT-4 Research and Perspective Towards the Future of Large Language Models”(Liu et al., 2023) and “How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation”(Hendy et al., 2023)], are tested in this exciting research/comercial use case in the politics/law domain.

At the same time and since we can’t fine-tune ChatGPT. we will benchmark **FLAN-T5** (base) on the same corpus since it is one of the few open-source language models that has been finetuned in tasks such as translation and also presents huge performances and interesting use cases.

Finally we will use Microsoft’s **Azure Cognitive Services Translator**, which serves as the other NMT model and is expected to perform the best due to its commercial position. Of course, we know it has to be neural-based but we can’t really say which specific structure is behind the model.

For the traditional metrics, we will use BLEU, METEOR and chrF(++) since they are the most popular and interesting. For the transformer-based ones, we will use BERTScore and BLEURT first and COMET-22 which we saw is considered sota (Freitag et al., 2022). That way, we have a balanced set of metrics for the discussion afterwards.

Chapter 4

Results and discussion

4.1 Results

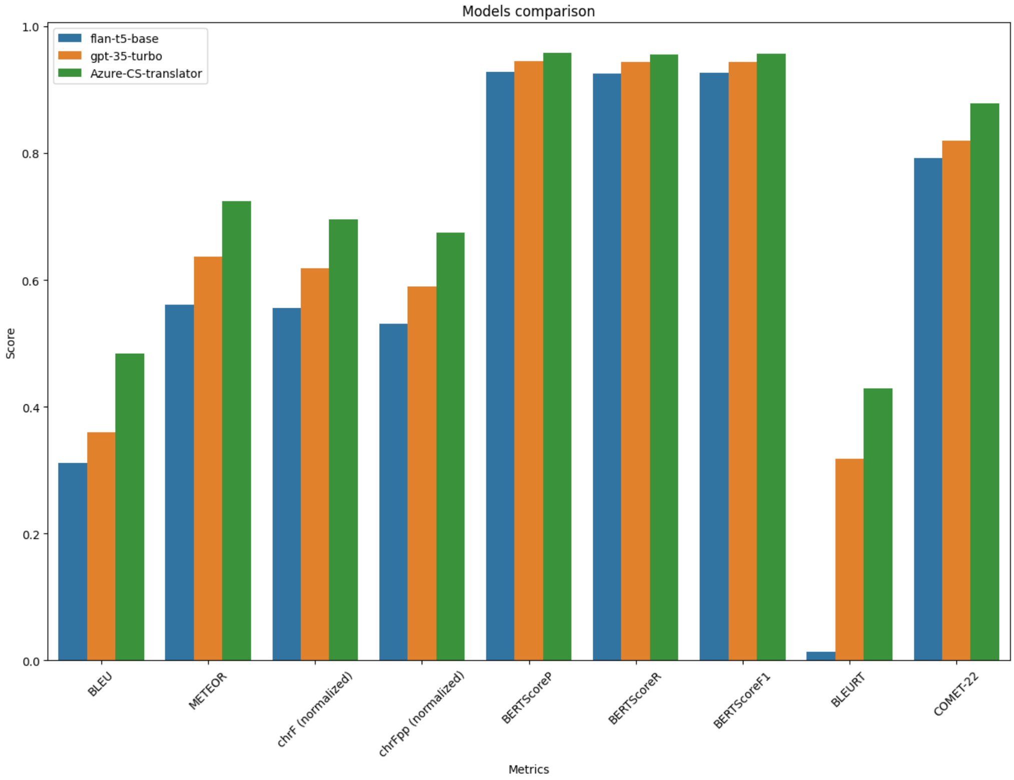


FIGURE 4.1: Evaluation results for the DCEP DE-EN Corpus

We can see the results of our benchmarking task above in Figure 4.1. The chrF and chrF++ metrics are normalized by 100 and BLEURT goes from $-\infty$ to $+\infty$ so even its results are good except for FLAN-T5 (which aren't that far from the other two models despite the distance in the graphic).

4.1.1 Discussion

Analyzing the results of our machine translation model and metrics comparison, a few compelling observations and hypotheses can be made:

The text and domain chosen for this general (not fine-tuned on this domain) sentence-level translations, makes it difficult to get high scores across the board for all metrics. However, **Azure C.S**. translator achieves that for even BLEU, which is something that we already were expecting.

ChatGPT is second and FLAN-T5 is left in the last position across all metrics. This shows how good **ChatGPT** is for a conversational-tuned LLM due to its sheer size and **few-shot capabilities** (see Chapter 2) we saw in other papers and use cases.

It is very clear that the only idea we can infer from the difference in scores between different metrics is that **neural-based metrics give all 3 models the highest scores by far**, which means a solid performance and the fact that they are fundamentally robust despite struggling with concrete translations.

Given that, it is clear how BLEU, a **precision-based** metric that relies heavily on exact n-gram matches, scores the lowest and may struggle to accurately evaluate the quality of translations in texts full of dates and uncommon succession of symbols (that shouldn't be preprocessed for the task at hand, translation) which is really bad when tokenizing in order to calculate BLEU score. Maybe a decisive point is that the translations aren't at the document level and BLEU is not fond of sentence-level translations so this adds up to a really bad result.

In general, **traditional metrics do not consider semantic similarity**, which might lead to a lower score even when a translation maintains the meaning of the original text and that is proved here.

It's worth noting that the type of text being translated—political/legal text—could be impacting the results. Such texts typically contain formal, domain-specific language and may have **complex sentence structures**. It not only clearly impacted the metrics but also the models and specially FLAN-T5 since sometimes it bugged repeating the same words in outputs when the sentence in question was hard.

In conclusion, the choice of evaluation metric is critical and should be dependent on the specifics of the text being translated and the aspects of translation quality that are most important for your use case. While traditional metrics like BLEU are widely used, they may not always be the most appropriate choice, particularly for complex, domain-specific texts. Neural-based metrics that consider semantic similarity, such as BERTScore and **COMET-22** (SOTA), could provide a more accurate evaluation in such cases. Apart from that, the outcome of our models was really predictable and yet ChatGPT was able to surprise us once again.

Chapter 5

Conclusions

5.1 Conclusions

The analysis undertaken in this study provides several compelling insights into the evaluation of machine translation models. As the results demonstrate, the Azure C.S. translator stands out as the leading performer across all metrics, which underlines the **power of commercial models** in the context of machine translation (*AI Index Report, Chapter 1: Research and Development* n.d.). Additionally, the **excellent performance of ChatGPT**, despite being a conversational-tuned LLM, testifies to the capabilities of large language models and their few-shot learning abilities as well as scalability potential.

The disparity observed across the scores of different metrics signals a **clear dominance of neural-based metrics**, which tend to provide higher scores compared to traditional metrics. This could be attributable to their focus on semantic similarity, a feature that seems particularly valuable for the complex, domain-specific texts used in this study and that seems hardly achievable for statistical MT metrics. It's thanks to great research and commercial interest that these huge transformer-based models are being made (Freitag et al., 2022).

The choice of text and domain, specifically political/legal texts, has evidently influenced the results, underscoring the importance of considering domain specificity and **fine-tuning** when evaluating machine translation models. In this context, traditional metrics like BLEU may struggle due to their focus on precision and exact n-gram matches, which does not fully cater to the complexity and semantic nuances of such texts.

In general, the conclusions found by our modest but huge results are really self-explainatory and served to confirm these last ideas, which is a success in our opinion.

Based on our findings, future researchers should be mindful of the type of texts they are working with and the particular characteristics of these texts. They should also be aware of the **limitations of traditional metrics** when dealing with complex, domain-specific texts and consider using neural-based metrics, which better account for semantic similarity.

Further work could also explore ways to enhance traditional metrics or develop new ones that offer a **balanced assessment of both semantic similarity and exact matches**. However, this might not be the case when these SOTA quality-assessment metrics are being produced. Future researchers should probably **include both** but they will have to decide.

Finally, given the dominance of commercial models like Azure C.S. translator, more research is needed to explore whether similar results could be asured in very different domains. Future studies could also investigate strategies for **improving the**

performance of open-source models to match, or even surpass, their commercial counterparts with the **appropiate strategies and quality assessments**.

Bibliography

AI Index Report, Chapter 1: Research and Development (n.d.). AI Index Report graphics. https://aiindex.stanford.edu/report/.

Brown, Tom et al. (May 2020). "Language Models are Few-Shot Learners". In: https://arxiv.org/abs/2005.14165.

Chung, Hyung Won et al. (2022). "Scaling Instruction-Finetuned Language Models". In: arXiv: 2210.11416. https://arxiv.org/abs/2210.11416.

Collin Raffel video (n.d.). T5 pretraining, fine-tuning, evaluation. https://www.youtube.com/watch?v=eKqWC577WlI&list=UUEqgmyWChwvt6MFGGlmUQCQ&index=5.

Deep Dive into DL (n.d.). The Transformer architecture. https://d2l.ai/chapter_attention-mechanisms-and-transformers/transformer.html.

Devlin, Jacob et al. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. arXiv: 1810.04805. https://arxiv.org/abs/1810.04805.

Fan, Angela et al. (Oct. 2020). "Beyond English-Centric Multilingual Machine Translation". In: https://arxiv.org/abs/2010.11125.

Freitag, Markus et al. (Dec. 2022). "Results of WMT22 Metrics Shared Task: Stop Using BLEU - Neural Metrics Are Better and More Robust". In: https://www.statmt.org/wmt22/pdf/2022.wmt-1.2.pdf.

GitHub repository (n.d.). GitHub repository with different Jupyter notebooks and Python scripts used in the benchmarking task. https://github.com/AlvLC/Machine-Translation-Evaluation-Metrics-Benchmarking.

Goodfellow, Ian et al. (June 2014). "Generative Adversarial Networks". In: *Advances in Neural Information Processing Systems* 3. DOI: 10.1145/3422622. https://arxiv.org/abs/1406.2661.

Gupta, Rohit, Constantin Orasan, and Josef Genabith (Sept. 2015). "ReVal: A Simple and Effective Machine Translation Evaluation Metric Based on Recurrent Neural Networks". In: DOI: 10.18653/v1/D15-1124. https://aclanthology.org/D15-1124.pdf.

Hendy, Amr et al. (Feb. 2023). "How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation". In: DOI: 10.48550/arXiv.2302.09210. https://arxiv.org/pdf/2302.09210.pdf.

Hutchins, John (Jan. 2004). "The first public demonstration of machine translation: the Georgetown-IBM system, 7th January 1954". In: https://aclanthology.org/www.mt-archive.info/00/AMTA-2004-Hutchins.pdf.

Isozaki, Hideki et al. (Jan. 2010). "Automatic Evaluation of Translation Quality for Distant Language Pairs". In: https://aclanthology.org/D10-1092/.

Kingma, Diederik and Jimmy Ba (Dec. 2014). "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations*. https://arxiv.org/abs/1412.6980.

Koehn, Philipp, Franz Och, and Daniel Marcu (Jan. 2003). "Statistical Phrase-Based Translation." In: DOI: 10.3115/1073445.1073462. https://aclanthology.org/N03-1017.pdf.

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey Hinton (Jan. 2012). "ImageNet Classification with Deep Convolutional Neural Networks". In: *Neural Information Processing Systems* 25. DOI: 10.1145/3065386. https://proceedings.neurips.cc/paper_files/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf.

Lavie, Alon and Abhaya Agarwal (July 2007). "METEOR: An automatic metric for MT evaluation with high levels of correlation with human judgments". In: https://aclanthology.org/W07-0734/.

Liu, Yiheng et al. (Apr. 2023). "Summary of ChatGPT/GPT-4 Research and Perspective Towards the Future of Large Language Models". In: https://arxiv.org/pdf/2304.01852.pdf.

Lo, Chi-kiu (Jan. 2019). "YiSi - a Unified Semantic MT Quality Evaluation and Estimation Metric for Languages with Different Levels of Available Resources". In: DOI: 10.18653/v1/W19-5358. https://aclanthology.org/W19-5358.pdf.

Marie, Benjamin, Atsushi Fujita, and Raphael Rubino (Jan. 2021). "Scientific Credibility of Machine Translation Research: A Meta-Evaluation of 769 Papers". In: DOI: 10.18653/v1/2021.acl-long.566. https://aclanthology.org/2021.acl-long.566/.

Mnih, Volodymyr et al. (Dec. 2013). "Playing Atari with Deep Reinforcement Learning". In: https://arxiv.org/abs/1312.5602.

Ouyang, Long et al. (2022). "Training language models to follow instructions with human feedback". In: *ArXiv* abs/2203.02155. https://arxiv.org/abs/2203.02155.

Papineni, Kishore et al. (Oct. 2002). "BLEU: a Method for Automatic Evaluation of Machine Translation". In: DOI: 10.3115/1073083.1073135. https://aclanthology.org/P02-1040.pdf.

Popovic, Maja (Sept. 2015). "chrF: character n-gram F-score for automatic MT evaluation". In: DOI: 10.18653/v1/W15-3049. https://aclanthology.org/W15-3049/.

Radford, Alec et al. (2018). "Improving language understanding by generative pre-training". In: https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.

Radford, Alec et al. (2019). "Language models are unsupervised multitask learners". In: *OpenAI blog* 1. https://d4mucfpksywv.cloudfront.net/better-language-models/language-models.pdf.

Raffel, Colin et al. (Oct. 2019). *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*. https://arxiv.org/abs/1910.10683.

Rei, Ricardo et al. (Jan. 2020). "COMET: A Neural Framework for MT Evaluation". In: DOI: 10.18653/v1/2020.emnlp-main.213. https://www.researchgate.net/publication/347234911_COMET_A_Neural_Framework_for_MT_Evaluation.

Rei, Ricardo et al. (Dec. 2022). "COMET-22: Unbabel-IST 2022 Submission for the Metrics Shared Task". In: https://aclanthology.org/2022.wmt-1.52.

Seah, Boon, Sourav S Bhowmick, and Aixin Sun (Aug. 2015). "PRISM: Concept-preserving Summarization of Top-K Social Image Search Results". In: *Proceedings of the VLDB Endowment* 8. DOI: 10.14778/2824032.2824088. https://www.researchgate.net/publication/283189799_PRISM_Concept-preserving_Summarization_of_Top-K_Social_Image_Search_Results.

Sellam, Thibault, Dipanjan Das, and Ankur Parikh (Apr. 2020). "BLEURT: Learning Robust Metrics for Text Generation". In: https://aclanthology.org/2020.acl-main.704/.

Snover, Matthew et al. (Jan. 2006). “A study of translation edit rate with targeted human annotation”. In: https://aclanthology.org/2006.amta-papers.25.pdf.

Srivastava, Nitish et al. (June 2014). “Dropout: A Simple Way to Prevent Neural Networks from Overfitting”. In: *Journal of Machine Learning Research* 15, pp. 1929–1958. https://jmlr.org/papers/volume15/srivastava14a/srivastava14a.pdf.

Sutskever, Ilya, Oriol Vinyals, and Quoc Le (Sept. 2014). “Sequence to Sequence Learning with Neural Networks”. In: *Advances in Neural Information Processing Systems* 4. https://arxiv.org/abs/1409.3215.

Vaswani, Ashish et al. (June 2017). “Attention Is All You Need”. In: https://arxiv.org/abs/1706.03762.

Wang, Weiyue et al. (Jan. 2016). “CharacTer: Translation Edit Rate on Character Level”. In: DOI: 10.18653/v1/W16-2342. https://aclanthology.org/W16-2342.pdf.

Yang, Zhen et al. (2022). “Findings of the WMT 2022 Shared Task on Translation Suggestion”. In: arXiv: 2211.16717. https://aclanthology.org/2022.wmt-1.70.pdf.

Zhang, Tianyi et al. (Apr. 2019). “BERTScore: Evaluating Text Generation with BERT”. In: https://arxiv.org/abs/1904.09675.