FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

DOCTORAL THESIS

Ivana Kvapilíková

Towards Machine Translation Based on Monolingual Texts

Institute of Formal and Applied Linguistics

Supervisor of the doctoral thesis: doc. RNDr. Ondřej Bojar, Ph.D.

Study programme: Computer Science

Study branch: Mathematical Linguistics

Prague 2023

I declare that I carried out this doctoral thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

Prague, November 12, 2023 Ivana Kvapilíková

Acknowledgements

I would like to express my gratitude to the people who supported me throughout my doctoral studies. Above all, I would like to thank my supervisor doc. RNDr. Ondřej Bojar, Ph.D. both for his expertise and for his kind support. I am grateful that he believed in me and gave me the opportunity to set out on this journey. He was always there to point me in the right direction, to encourage me, and to push my limits. This accomplishment would not have been possible without his mentorship.

I would also like to thank my friends and colleagues at the Institute of Formal and Applied Linguistics who have been a source of inspiration, advice, and also fun during the last five years.

My special thanks goes to my great friends and fellow researchers PhDr. Pavlína Honsová, Ph.D. and Anna Houštecká, Ph.D. for our inspirational talks (not only) about research and for their endless moral support.

Finally, I would like to send my love to my family, especially my son, and thank them for their patience during the final weeks of writing this thesis.

The work on this thesis was supported by the Charles University Grant Agency (GAUK 1050119, SVV 260 453, SVV 260 575, and SVV 250 698), the Czech Science Foundation 19-26934X (NEUREM3), and the project "Grant Schemes at CU" (reg. no. CZ.02.2.69/0.0/0.0/19_073/0016935). This work has been using language resources and tools stored and distributed by the LINDAT/CLARIN project of the Ministry of Education, Youth and Sports of the Czech Republic (LM2015071).

Title: Towards Machine Translation Based on Monolingual Texts

Author: Ivana Kvapilíková

Institute: Institute of Formal and Applied Linguistics

Supervisor: doc. RNDr. Ondřej Bojar, Ph.D., Institute of Formal and Applied Linguistics

Abstract: The current state of the art in machine translation (MT) heavily relies on parallel data, i.e. texts that have been previously translated by humans. This type of resource is expensive and only available for several language pairs in limited domains. A new line of research has emerged to design models capable of learning to translate from monolingual texts which are significantly easier to obtain, e.g. by web-crawling. While it is impressive that such models achieve translation capabilities, the translation quality of the output they produce is still low for practical applications. This dissertation thesis strives to improve their performance. We explore the existing approaches of using monolingual resources to train translation models and propose a new technique to generate pseudo-parallel training data artificially without expensive human input. We automatically select similar sentences from monolingual corpora in different languages and we show that using them in the initial stages of MT training leads to a significant enhancement in translation quality. We also point out the limitations of existing MT models based on monolingual texts which often struggle with the translation of named entities and generally produce low-quality translations, especially in truly low-resource conditions where monolingual training data is limited and often suffers from a domain mismatch.

Keywords: machine translation, unsupervised learning, deep neural networks, low-resource languages, natural language processing

Název: Strojový překlad na základě jednojazyčných textů

Autor: Ivana Kvapilíková

Ústav: Ústav aplikované a formální lingvistiky

Vedoucí: doc. RNDr. Ondřej Bojar, Ph.D., Ústav aplikované a formální lingvistiky

Abstrakt: Současné systémy strojového překladu (SP) jsou závislé na existenci paralelních dat, tedy textů, které byly dříve přeloženy lidmi. Tento typ dat je drahý a je dostupný pouze pro několik jazykových párů v omezených doménách. Vznikl tedy nový výzkumný směr zaměřený na navrhování modelů schopných naučit se překládat z jednojazyčných textů, které jsou výrazně dostupnější než texty paralelní, např. z internetu. I když je působivé, že takové modely překládat skutečně dokáží, kvalita jimi vyprodukovaných výstupů je pro praktické aplikace stále nedostatečná. Tato disertační práce se snaží vylepšit jejich výkonnost. Zkoumáme stávající přístupy používání jednojazyčných zdrojů k trénování překladových modelů a navrhujeme novou techniku generování pseudo-paralelních trénovacích dat uměle, bez drahého lidského vstupu. Automaticky hledáme podobné věty v jednojazyčný korpusech v různých jazycích a ukazujeme, že jejich použití v počátečních fázích trénování SP vede k výraznému zvýšení kvality překladu. Poukazujeme také na omezení stávajících modelů SP založených na jednojazyčných textech, které si často nedokáží poradit s překladem pojmenovaných entit a obecně produkují nekvalitní překlady, zejména v podmínkách s opravdu omezenými zdroji, kde je k dispozici pouze malé množství jednojazyčných textů, které navíc patří do odlišných domén.

Klíčová slova: strojový překlad, neřízené učení, hluboké neuronové sítě, nízkozdrojové jazyky, zpracování přirozeného jazyka

Contents

1. Introduction

Modern machine translation (MT) systems are trained on large parallel corpora, i.e. collections of sentence-aligned text documents translated by humans, ideally professional translators. While there are public sources of parallel data for several dominant languages (e.g. EU legislation, public domain books, movie subtitles), the only parallel corpus available for many other language pairs is the Bible. There are more than 7,000 [Eberhard et al., 2023] languages spoken in the world and only a small fraction of them is covered by large data sets, others are considered low-resource for most natural language processing (NLP) tasks, including MT.

The scarcity of parallel data motivated researchers to devise a new training strategy where the MT model can learn from monolingual texts which are significantly easier to obtain (e.g. by web crawling) than parallel texts. Monolingual corpora were then used in combination with existing parallel resources to increase translation quality and it was an open question whether an MT system can be trained purely from monolingual data.

The problem of learning to translate without ever seeing a translation was first tackled as a deciphering problem [Ravi and Knight, 2011] where foreign text was viewed merely as an unknown cipher of the English text. The idea seemed intriguing but quite unrealistic, until the pioneering work of Artetxe et al. [2018d] and Lample et al. [2018a]. It was shown that minimal supervision suffices to teach a neural model to align monolingual word representations (embeddings) and find translation equivalents. Unsupervised training of MT systems became a hot topic both for the curiosity of a seemingly unsolvable task as well as for its relevance for low-resource language pairs.

The initial attempts at unsupervised machine translation (UMT) applied the newest advancements in deep neural models. However, it was quickly realized that statistical phrase-based machine translation (PBMT) offered a valuable toolkit for unsupervised scenarios, and the performance of phrase-based systems even surpassed that of the initial unsupervised neural models [Lample et al., 2018a, Artetxe et al., 2018b]. It was only when the benefits of cross-lingual pre-training were discovered [Conneau and Lample, 2019] that the performance of unsupervised PBMT models started to lag behind. Until today, using a hybrid approach [Artetxe et al., 2019a] where translations from a PBMT system are used to pre-train a deep neural system is still a relevant strategy which, in some settings, can supersede purely neural systems.

Although the translation quality achieved by a completely unsupervised system did not reach the level of supervised MT, the initial attempts showed that training of machine translation exclusively on monolingual texts is feasible. New advances significantly increased the performance, leaving the question of the maximum attainable translation quality for an MT system trained exclusively on monolingual corpora unanswered. In this thesis, we strive to move towards that limit by proposing new components of the unsupervised training schedule.

Several authors [Marchisio et al., 2020, Søgaard et al., 2018] have pointed out limitations of UMT, especially in the context of translation of truly low-resource languages where we do not have gigabytes of monolingual texts to use for training and where the training data likely covers only limited domains. To be able to draw robust conclusions, we evaluate our approach on authentic low-resource language pairs with a presence of monolingual data but limited or non-existent parallel texts. Aside from low-resource languages, unsupervised methods can also be useful for domain-specific translation between languages which are generally considered high-resource but lack parallel data in particular domains. However, we do not explore this direction in our thesis.

This thesis investigates unsupervised learning strategies to find the most efficient way to exploit monolingual data for cross-lingual signal. There are two main directions this work will explore: (1) methods for obtaining parallel data when authentic parallel resources are unavailable, and (2) UMT models, their architecture, and training strategies. The two directions are closely intertwined since UMT models are always trained using a form of synthetic parallel data. Moreover, the underlying problem behind the UMT task as well as the unsupervised parallel corpus mining (PCM) task is the building of a cross-lingual space which we can either use to initialize an MT system or to search for similar sentences. In our analysis, we will focus on various techniques to induce the cross-lingual space and enhance the alignment of parallel word and sentence representations. We will explore the effect of multilingual training on the quality of the representations and on the performance of UMT systems.

We first introduce our background and motivation in Chapter 2. We present the theoretical foundations behind our work in Chapter 3, followed by the literature review of unsupervised methods in MT in Chapter 4. We give details on how we obtain the training data for our experiments in Chapter 5. We outline the unsupervised training methodology in Chapter 6. We describe our translation experiments and comment on the results in Chapter 7. We discuss the translation quality and point out the limitations of unsupervised techniques in Chapter 8. We conclude by summarizing our main findings and uncovering potential directions for future research.

2. Background

2.1 Language Data Resources

Language data resources refer to the various sources of information that are used to study, process, and analyze language. In the context of machine translation, the most relevant data resources are written text corpora, pre-translated texts (parallel corpora), word lexicons and pre-trained models. In other areas of linguistic research, useful resources include treebanks (for syntactic and morphological analysis), speech corpora (for automatic speech recognition) or other annotated corpora (for sentiment analysis, sentence similarity search, named entity recognition etc.).

2.1.1 Monolingual Corpora

A monolingual corpus is a collection of texts in a single language. For the purposes of this thesis, a monolingual corpus is understood as a collection of texts in a single language in plain text with no additional annotations. Out of all NLP resource types, monolingual corpora are the easiest to obtain. Even in many low-resource languages, it is possible to gather significant amounts of text by automatic web crawling. The CommonCrawl[1] project carries out periodic web crawls and publishes the crawled data in an open repository with public access. The repository contains petabytes of data collected since 2008. The quality of web-crawled corpora is dubious even after filtering [Kreutzer et al., 2022] but for low-resource languages, it is often the only data source available. Artetxe et al. [2022] demonstrate that in cases where there is not a sufficient amount of high-quality curated data, the benefits of having a larger and a more diverse corpus are worth the potential data quality issues.

The majority of monolingual corpora used in MT papers is derived by automatic filtering of the CommonCrawl corpus. For example, the open source OSCAR[2] project compiled a large multilingual corpus by language classification and filtering of the CommonCrawl with the goal of providing large quantities of raw text to be used mostly for pre-training of large deep learning models in 151 languages.

Monolingual corpora can come from different domains. The popularity of online newspapers warrants a high representation of the news domain in the crawled corpora. Newspaper articles are collected in the NewsCrawl[3] corpus which is released every year for the WMT series of shared tasks. Similarly, the legal domain is strong due to the online legal codes, European regulations and international treaties which are publicly available.[4]

---

[1] `https://commoncrawl.org/`
[2] `https://oscar-project.org/`
[3] `https://data.statmt.org/news-crawl/`
[4] `https://www.clarin.eu/resource-families/legal-corpora`

2.1.2 Parallel Corpora

A parallel corpus is a primary resource for standard MT training. It is a collection of texts in different languages that are aligned at the sentence level. In a parallel corpus, each sentence or phrase in one language corresponds to its translation equivalent in another language. Parallel corpora are typically created by professional translators or by collecting documents that have been translated for various purposes, such as multilingual websites, official documents, or bilingual books. While most parallel corpora are bilingual, some have multiway alignments between all covered languages (e.g. the eBible[5] corpus).

Most publicly available parallel corpora are gathered on the OPUS[6] website for anybody to download. The largest corpora come from the mixed domain, but there are significant resources of specialized texts as well. Legal texts often naturally originate in multiple languages in parallel, e.g. the extensive EuroParl[7] corpus of proceedings of the European Parliament covers all EU languages. Similarly, the EMEA[8] corpus comprises documents from the European Medicines Agency in all EU languages.

As far as low-resource languages are concerned, Tatoeba[9] is a collaborative online project that aims to create a multilingual corpus of sentences and translations for underrepresented languages. It allows users to contribute sentences in various languages along with their translations into other languages. The corpus is continuously expanded and improved through the collaborative efforts of volunteers from around the world. The number of translated sentences in each language varies from only a couple to several thousand. Besides Tatoeba, the only parallel datasets for truly low-resource languages are often the Bible [Akerman et al., 2023] or the Ubuntu localization files which are small and narrowly specialized [Tiedemann, 2012]. Costa-jussà et al. [2022] compiled a multiway parallel corpus FLORES-200 of 3k sentences curated by professional translators in 200 low-resource languages.

The lack of parallel data faced by many language pairs is the reason why researchers explore the options of utilizing monolingual data for MT training.

2.1.3 Comparable Corpora

A comparable corpus is a collection of texts in different languages that are comparable in terms of genre, content and purpose. Unlike parallel corpora, they are not sentence-aligned but they can be aligned at the paragraph or document level. A popular example of a comparable corpus is Wikipedia,[10] where articles on the same topic in different languages are linked but they vary in their content as well as their length. The Wikipedia size of each language is a good proxy of the online presence of a language and the strength of the community supporting its preservation.

[5] `https://github.com/BibleNLP/ebible`
[6] `[https://opus.nlpl.eu/`
[7] `https://www.statmt.org/europarl/`
[8] `https://inventory.clarin.gr/corpus/747`
[9] `https://tatoeba.org/`
[10] `https://lindat.mff.cuni.cz/repository/xmlui/handle/11234/1-2735`

In our work, we use comparable corpora to search for translation equivalents to build a pseudo-parallel corpus.

2.1.4 Pseudo-Parallel Corpora

A pseudo-parallel corpus is a collection of text data that is not perfectly aligned or parallel, but still provides useful information for machine translation and other language processing tasks. Unlike a true parallel corpus, where the paired sentences fully correspond to each other, a pseudo-parallel corpus consists of similar but not necessarily identical texts in two or more languages. In the context of this work, a pseudo-parallel corpus is created by automatic search for parallel sentences in two monolingual and preferably comparable corpora.

2.1.5 Synthetic Parallel Corpora

Synthetic parallel corpora arise by a process called back-translation [Sennrich et al., 2016] when a trained MT system is used to translate a monolingual corpus and the original sentences are coupled with their synthetic translations. The source side of the resulting parallel corpus is usually the synthetic one while the target side has the original authentic sentences. Using translations from a phrase-based system to train a neural system in the opposite translation direction is an effective approach to unsupervised MT which we explore in Section 7.2.

2.1.6 Pre-Trained Models

Pre-trained models refer to machine learning models that have been trained on large amounts of text data and made available for general use, e.g. in the HuggingFace Model Hub.[11] The training process involves exposing the model to vast amounts of text data and optimizing its parameters to learn patterns, relationships, and representations of language. This allows the model to capture various linguistic properties, contextual information, and semantic relationships between words and sentences.

Pre-trained models have become one of the most powerful resources for NLP applications as they allow researchers to reach state-of-the-art results with limited computation capacity. However, their performance for underrepresented languages is usually subpar and many languages are not supported at all. In spite of that, utilizing the knowledge learned from high-resource languages is an effective strategy when training a model for a low-resource language [Zoph et al., 2016, Nguyen and Chiang, 2017, Kocmi and Bojar, 2018]. In this work, we use large-scale multilingual models from the BERT family [Devlin et al., 2018] as sentence encoders and we fine-tune them for better performance on the languages of our interest. More details on pre-trained language models will be given in Chapter 3 and later in Chapter 5.

[11] `https://huggingface.co/`

Class	#Langs	#Speakers	% of Total Langs	Example Languages
Dominant	7	2.5B	0.28%	English, German
High-resource	18	2.2B	1.07%	Czech
Low-resource I.	28	1.8B	4.42%	Kazakh, Ukrainian, Georgian
Low-resource II.	241	36M	5.85%	Upper Sorbian, Inuktitut, Assamese, Khasi, Manipuri, Mizo
No-resource	2,191	1.2B	88.38%	–

Table 2.1: Taxonomy of languages originally by Joshi et al. [2020] with a number of languages per group, a number of speakers per group, and a percentage of total languages. We use it for classification of the languages we focus on in this work.

2.2 Cross-lingual Information in Monolingual Data

Collections of texts in multiple languages inherently contain a translation signal, even if the texts are not explicitly matched. It is possible that equivalent sentences are concealed within the corpora, and these can be automatically identified before the translation training starts. In such cases, we refer to the process as the creation of a *pseudo-parallel corpus* in advance.

In other cases, especially when the monolingual corpora are of limited size and the likelihood of discovering matching sentences is low, we can explore semantic correspondences at the level of individual words or short phrases, considering their context. The core concept here is that across languages, similar words tend to occur in similar contexts. While this principle may not be universally applicable across distinct cultural, climatic, or socioeconomic backgrounds, when the corpora share a common domain, it becomes possible to leverage this similarity to extract a word or phrase dictionary, often referred to as a **lexicon**.

Such a lexicon serves as a valuable resource for generating a *synthetic parallel corpus*. This can be achieved through word-by-word translation or by employing a phrase-based machine translation system. Although these initial translations are far from perfect, they represent a potential source of cross-lingual signal when true parallel data is not readily available.

It came as a surprise that multilingual language-representation models trained without any cross-lingual objective are able to uncover text correspondences in monolingual data [Pires et al., 2019]. This likely happens due to the limited capacity of the models which forces them to economize and find the right alignments between their internal representations. This form of cross-lingual information emerges at the level of context representation and, therefore, is only accessible to machine learning models. It can be leveraged by copying the weights of the pre-trained language model into the neural MT model [Conneau and Lample, 2019] as will be described in Chapter 6.

2.3 Languages of the World

There are estimated to be 7,168 living languages spoken in the world today [Eberhard et al., 2023]. These languages are diverse and vary widely in terms of their structure, grammar,

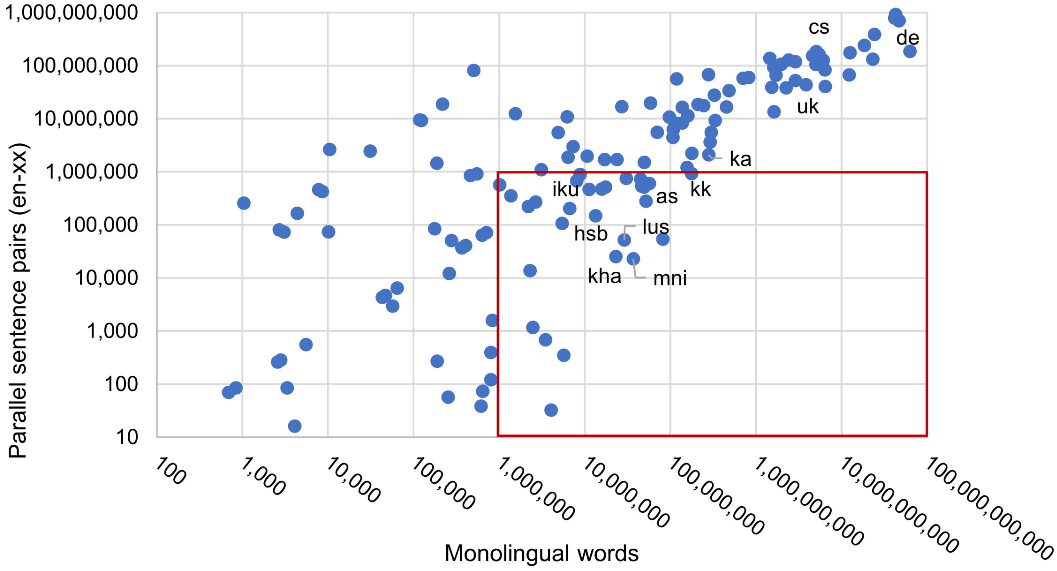


Figure 2.1: World languages plotted in terms of the available textual data – raw monolingual (horizontal axis) and parallel English-aligned (vertical axis). Both axes are in log scale. The rectangle delimits the area of low-resource languages that this thesis focuses on.

vocabulary and usage. Along with the well-developed and universally supported languages with a strong speaker base, there are languages without a proper writing system and with only a handful of speakers left with their unique knowledge. NLP technologies strive to provide support for speakers of low-resource languages as well as work towards the preservation of the language itself. Many of the world's languages are endangered, with some estimates suggesting that up to half of all languages could disappear by the end of the 21st century.[12]

Joshi et al. [2020] distinguish six kinds of languages according to their digital status. They propose a taxonomy which is based on the amount of labeled and unlabeled data available online for each language. According to their findings, 88.38% of the 2,455 considered languages fall into the last category which is completely ignored by digital language technologies. The first category, on the other hand, includes only seven languages (English, Spanish, German, Japanese, French, Chinese, and Arabic) with a dominant online presence and a superiority over other languages in terms of the amount of both labeled and unlabeled data, enabling them to benefit from all NLP breakthroughs. Most of the remaining European languages fall in the second category characterized by dedicated NLP communities and strong economical and political links to the *dominant* languages. In this thesis, we mostly target the *low-resource* languages from the remaining two groups, spoken by almost 2 billion people in total. A sufficient amount of unlabeled (monolingual) data and a lack of labeled (parallel) sentences constitute the ideal scenario for UMT training. The languages we work with and their corresponding categories are listed in Table 2.1.

[12] `https://www.ethnologue.com/insights/how-many-languages-endangered/`

2.4 Low-Resource Languages

In order to determine the scope of this work, we need to assess which languages are considered *low-resource* for the task of machine translation and how many such languages there are. We gauge the quantity of parallel data accessible for each language by calculating the number of English-aligned parallel sentences found on the OPUS website, in conjunction with the supplementary corpora provided for the WMT translation shared tasks.[13] The quantity of parallel sentences aligned with English serves as a rough estimate for the upper limit of parallel sentences aligned with other languages. This is because language pairs not involving English typically have a smaller amount of parallel data. As a proxy for the total amount of monolingual data available, we consider the Oscar corpus sizes. It must be noted that both OPUS and Oscar include uncleaned text data with a lot of noise and possible duplicates. We display the languages in terms of their quantities of labeled and unlabeled data in Figure 2.1. The results are plotted in log scale to better illustrate the distribution of languages.

Out of the 151 languages covered by the Oscar corpus, 79 have less than 1M uncleaned parallel sentence pairs, making them suitable candidates for unsupervised training. For the purposes of this work, we call these languages *low-resource*. The threshold of 1M parallel sentences is motivated by Kocmi [2020] who shows that training MT models with fewer sentences leads to fast over-fitting and hindered translation performance. The rectangle in Figure 2.1 delimits the space where unsupervised pre-training techniques are most needed for the lack of parallel data (<1M sentence pairs) and where they are applicable for the abundance of monolingual data (>1M words for unsupervised training). The languages to the left of the rectangle can be called *very low-resource* and they cannot easily benefit from the techniques we propose due to their limited amounts of monolingual data. Many other languages are not even plotted in the chart as they do not have any data available in the OSCAR corpus.

2.5 The Extent of This Study

In this thesis, we focus on several language pairs, most of which are characterized as low-resource. This section provides an overview of these language pairs, their relevance to the experiments conducted, and essential linguistic details [Eberhard et al., 2023].

- We train domain-specific MT models for translation from English to Ukrainian, Kazakh, and Georgian. Kazakh, belonging to the Turkic language family, and Georgian, belonging to the isolated Kartvelian language family, enable us to validate our approaches across a wide spectrum of linguistic variation.

- We conduct experiments involving translation between English and four low-resource Indic languages (Assamese, Khasi, Manipuri, Mizo). The amount of monolingual data available for these languages is significantly lower than for the languages in the first group, which allows us to test the limits of our approaches in truly low-resource scenarios. These languages are among the 22 official languages of the Indian Republic and

[13] `https://statmt.org/`

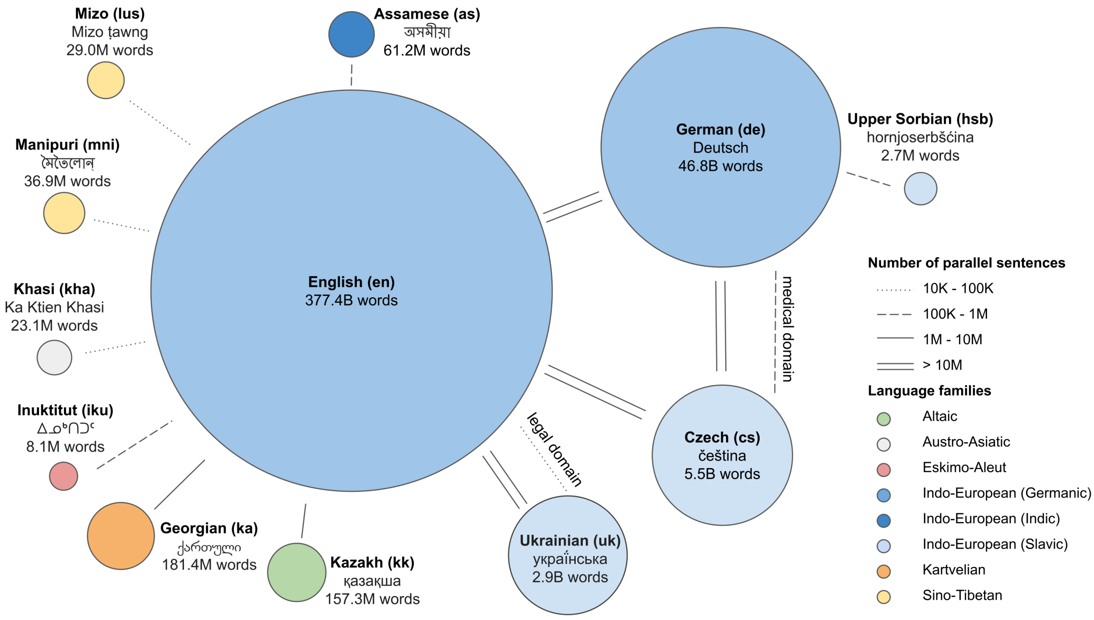


Figure 2.2: Languages used in this thesis in terms of the size of the available monolingual texts. Colors reflect language families and the links between languages represent the amount of parallel data available.

exhibit considerable linguistic diversity. Specifically, Manipuri (also called Meitei) and Mizo belong to the Sino-Tibetan language family, Khasi is a member of the Austro-Asiatic language family, and Assamese is part of the Indo-Aryan branch of the Indo-European language family. Assamese and Manipuri share a common Bengali-Assamese script.

- Inuktitut is an Eskimo-Aleut language and we use it to test our approach to parallel corpus mining on a low-resource language with a unique script.

- Our other experiments encompass more closely related Indo-European languages. While the German-Czech language pair has access to substantial volumes of pre-translated texts, we employed it in our preliminary experiments with unsupervised approaches. On the other hand, German and Upper Sorbian is a language pair which represents an authentic low-resource scenario where translation holds important socio-economic significance, given that Upper Sorbian is spoken in a region of Saxony in Germany.

Figure 2.2 illustrates the language pairs relevant for this thesis, their corpus sizes and their linguistic similarity. Figure 2.3 shows the languages in terms of their speaker base rather than their text data amounts. Comparing the two figures allows us to judge how big a language really is (as represented by the number of native speakers) in contrast with how strong its online presence is. The dominance of English or German is less pronounced when measured by the size of their speaker base. On the other hand, Czech is an example of a language which possesses a comparatively abundant volume of data in relation to its number of speakers which suggests a strong NLP community supporting it. Similarly, Inuktitut has only 38k speakers but a relatively big parallel corpus of 1M languages due to the support of the National Research

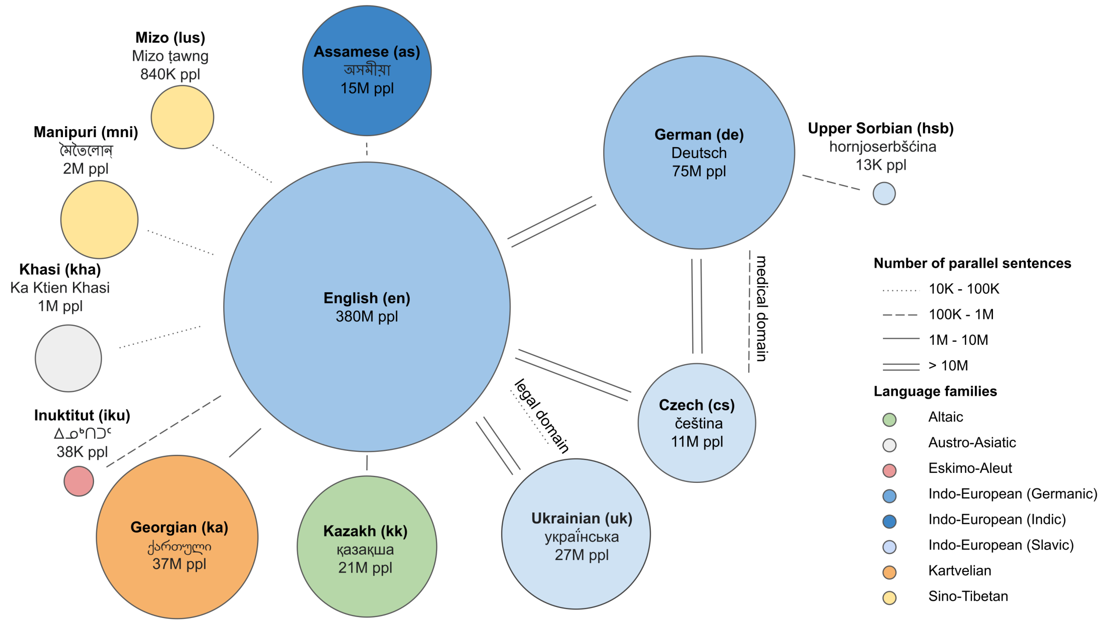


Figure 2.3: Languages used in this thesis in terms of the number of native speakers. Colors reflect language families and the links between languages represent the amount of parallel data available.

Council of Canada which published the proceedings of the Legislative Assembly of Nunavut in the Hansard corpus.[14]

When training a machine translation system, we explore the possibilities of utilizing monolingual data in other languages. However, using parallel data in other languages for translation knowledge transfer is out of scope and the readers are referred to Kocmi [2020] for more details on transfer learning for low-resource languages.

[14]`https://nrc-digital-repository.canada.ca/eng/view/object/?id=c7e34fa7-7629-43c2-bd6d-19b32bf64f60`

3. NLP Fundamentals

In this chapter, we describe the main foundation blocks that we build upon later when describing the methodology of our work. We start by introducing the concept of word embeddings and move on to the state-of-the-art language representation models with the Transformer architecture. We finally introduce the fundamentals of phrase-based machine translation (PBMT) and neural machine translation (NMT).

3.1 Word Embeddings

In order to process words using machine learning models, it is necessary to assign them a numerical representation. The simplest way for the model to differentiate one word from another would be by the so-called one-hot encoding where a vector of length $|V|$ is assigned to each word i of the vocabulary $|+V$ with vector elements $z_j = 0$ if $j \neq i$ and $z_j = 1$ if $j = i$. However, such a vector treats words as mere indices in a vocabulary and does not carry any linguistic information.

Word embeddings, on the other hand, are continuous real-valued vector representations of words trained so that words that are semantically close are also close in the embedding vector space. The concept stems from the *distributional hypothesis* [Harris, 1954] which suggests that words that appear in similar contexts tend to have similar meanings. The first notion of distributed word feature vectors was introduced by Bengio et al. [2003] who proposed them as an ailment for the *curse of dimensionality* inherent to the task of language modeling. An efficient way to obtain these vector was later discovered by Mikolov et al. [2013c].

Word embeddings can also be viewed as a mapping from the high-dimensional space $\{0,1\}^{|V|}$ to a lower-dimensional one $\mathbb{R}^E$ where $|V|$ is the size of the vocabulary and E is the embedding dimension and $E << |V|$. They can be learned by various neural models which will be introduced in the following paragraphs.

3.1.1 Static Word Embeddings

Static word embeddings are fixed-length real-valued vector representations of words that carry semantic information. A major breakthrough was achieved by Mikolov et al. [2013a] and their Word2Vec that learns word embeddings by two types of models – continuous bag-of-words (CBOW) and Skip-gram. The former learns to predict the current word based on its context (surrounding words) while the latter learns to predict the context given the current word. The architecture is illustrated in Figure 3.1. Models trained for other NLP tasks, including MT, also create their own static embeddings which will be discussed in Section 3.2.2 and Section 3.3.1.

Skip-gram Model

Skip-gram model is a feed-forward neural network that takes input as a one-hot vector with dimensions $1 \times |V|$. It has a single hidden layer that projects the input into the E-dimensional

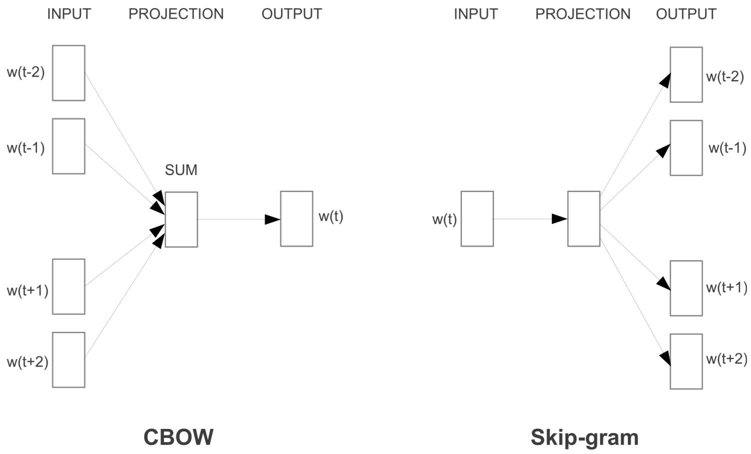


Figure 3.1: Word2Vec model architectures. The CBOW architecture predicts the current word based on the context, the Skip-gram predicts surrounding words given the current word.

Source: Mikolov et al. [2013c]

space and an output layer with a softmax activation function over the vocabulary of size $|V|$, which again outputs a one-hot vector. The dimensions of the hidden and output weights are $|V| \times E$ and $E \times |V|$, respectively.

The training task for the Skip-gram model is to predict the surrounding words of the current word. The model is presented with a pair of words at a time, composed of the current word in the output and one of its context words in the output. The context is defined as the set of words within a window of length c from the current word. Closer context words are sampled more frequently to approximate the looser relationships between more distant words.

Our focus does not lie in solving the task itself. Instead, we seek valuable internal representations that the model must construct in order to address the task effectively. They are stored in the hidden layer of the model and the word embeddings of all words from the vocabulary are obtained by simply extracting the hidden weight matrix ($|V| \times E$).

While embeddings of entire words are useful for semantic processing and tasks such as word similarity search, other tasks, such as machine translation, operate with smaller units (subwords). Kocmi and Bojar [2016] reach a better performance on the Skip-gram test set by a SubGram model which considers the word structure when training the embeddings. Similarly, FastText [Bojanowski et al., 2017] extends the Skip-gram model by enriching it with subword information to reflect the morphological properties of the words. The FastText model represents words by the sum of the vector representations of their character n-grams.

Continuous bag-of-words (CBOW) model

The training task behind the CBOW model is opposite to the Skip-gram. The input to the model is several context words (e.g. 2 or 3 before and after the current word, depending on the size of the window) which are projected to the hidden layer and averaged. The average embedding vector is then projected back to the output layer which should predict the current word. The dimensions of the hidden and the output layer are identical to the Skip-gram model.

According to Mikolov et al. [2013c], CBOW is faster to train than Skip-gram and it is better suited for large corpora, but Skip-gram can better represent less frequent words, especially when the training data is small.

3.1.2 Contextual Word Embeddings

In contrast to static word embeddings, contextual word embeddings are a function of the entire sentence (or any text stream) containing the given word. They arise from the internal representations of language models. As opposed to static embeddings which are type-level, contextual embeddings assign a unique vector to every token being processed based on its context. In order to get rid of the dynamic context dependency of contextual embeddings and obtain an equivalent of static embeddings, one can simply take their average per word type over a text corpus (or its subset). Schuster et al. [2019] show that contextual embeddings cluster around their average anchor and polysemous words are characterized by multi-modal clusters.

Two important examples of pre-trained contextual word embeddings are ELMo (Embeddings from Language Models) and BERT (Bidirectional Encoder Representations). ELMo [Peters et al., 2018] embeddings are computed on top of a bidirectional recurrent language model with character convolutions. The contextual representation of each token is the concatenation of the left-to-right and right-to-left representations. BERT [Devlin et al., 2018] embeddings are retrieved from the encoder outputs of a Transformer language model. More details about the Transformer architecture will be given in Section 3.2.

3.1.3 Cross-lingual Word Embeddings

The notion behind cross-lingual embeddings resembles the theoretical concept of interlingua – a space where meaning is represented regardless of the language it is expressed in.

Static word embeddings were shown to have many favorable properties regarding semantically meaningful geometric arrangements of word representations which could be exploited for turning monolingual embedding vectors into a cross-lingual space. The rationale behind this is that the use of language reflects concepts grounded in the real world. Since real-world concepts do not change upon expression in different languages, the embedding spaces in different languages are expected to be approximately isomorphic [Storer, 1952]. Several authors [Mikolov et al., 2013b, Conneau et al., 2018a, Artetxe et al., 2018c] leverage this property to obtain cross-lingual embeddings by linear mapping as illustrated in Figure 3.2. The idea of language isomorphism is at the core of many UMT approaches.

Formally, if embedding spaces in different languages are perfectly isomorphic, there exists a linear mapping between them [Mikolov et al., 2013b]. In presence of a bilingual seed lexicon L, the problem of finding the mapping matrix $W \in \mathbb{R}^{dim \times dim}$ between monolingual

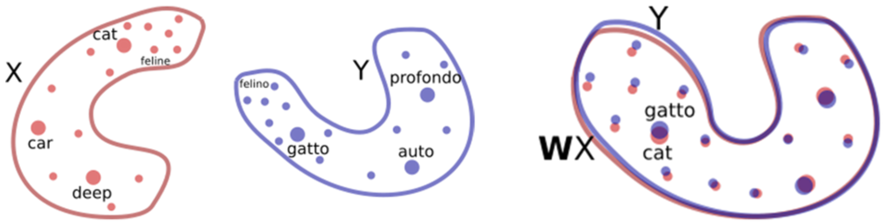


Figure 3.2: Mapping monolingual embeddings to cross-lingual space
Source: [Conneau et al., 2018a]

embeddings of length dim is then defined as

$$W^* = \underset{W \in \mathbb{R}^{dim \times dim}}{\operatorname{argmin}} ||WX_{seed} - Y_{seed}||_F \tag{3.1}$$

where X_{seed} and Y_{seed} are the $|L| \times dim$ matrices of corresponding source embeddings $x_1, ..., x_{|L|}$ and target embeddings $y_1, ..., y_{|L|}$, and $|L|$ is the size of the bilingual seed lexicon. Xing et al. [2015] show that it can be assumed that the mapping is orthogonal which turns the problem of finding the embedding mapping matrix W into the orthogonal Procrustes problem [Hurley and Cattell, 1962] with a closed-form solution [Schönemann, 1966] given by singular value decomposition (SVD)

$$W^* = \underset{W \in \mathbb{R}^{dim \times dim} s.t. W^T W = 1}{\operatorname{argmin}} ||WX_{seed} - Y_{seed}||_F = UV^T \tag{3.2}$$

where $U \Sigma V^T = \text{SVD}(Y_{seed} X_{seed}^T)$.

The mapping matrix W is finally used for post-hoc alignment of all embeddings $x_1, ..., x_{|V_{src}|}$ from the source language vocabulary V_{src} into the target language embedding space. If the embedding spaces are at least approximately isomorphic, the resulting embedding space in $\mathbb{R}^{dim}$ populated by target embeddings $y_j, j = 1, \ldots, |V_{tgt}|$ and aligned source embeddings $Wx_i, i = 1, \ldots, |V_{src}|$ is cross-lingual and can be used for finding word translation pairs based on their vector similarity score, e.g. cosine similarity.

However, several authors [Søgaard et al., 2018, Ormazabal et al., 2019, Patra et al., 2019, Vulić et al., 2020] criticize this theoretically valid approach for not having sufficient ground in real-life situations. They argue that the underlying assumption of the isomorphism of embedding spaces is frequently not met, particularly in scenarios where languages and domains exhibit significant dissimilarities, as is frequently the case in low-resource contexts. According to Søgaard et al. [2018], isomorphism is also influenced by the type and parameters of the word embedding algorithm, and they stress the importance of the same configuration on both sides. They are skeptical about their use for unsupervised translationHowever, when domain-balanced corpora are available, the linear mapping approaches work reasonably well [Mikolov et al., 2013b] even in unsupervised conditions [Conneau et al., 2018a, Artetxe et al., 2018c]. Unsupervised mapping techniques which do not count with a manually created bilingual seed lexicon L for supervision will be described in Chapter 6.

3.2 Transformer Language Models

The Transformer model [Vaswani et al., 2017] was proposed as a new solution to sequence-to-sequence modeling tasks which were previously tackled by recurrent neural networks (RNNs) with gated recurrent units (GRU) [Chung et al., 2014] and long short-term memory (LSTM) [Hochreiter and Schmidhuber, 1997] cells. Recurrent models process text auto-regressively, one token at a time, and the time dependency is modeled by the previous hidden states of the model which serve as an additional input to the recurrent layers. RNNs reached impressive performance both in language modeling and machine translation [Mikolov et al., 2010, Sutskever et al., 2014]. However, RNNs struggle with modeling of long dependencies and remembering earlier contexts. The problem was partially alleviated by using the attention mechanism [Bahdanau et al., 2015] where the model only attends to the part of the input that is relevant for generating the output. The Transformer model goes even further and removes the recurrent part of the model entirely, claiming that the *Attention is All You Need* [Vaswani et al., 2017]. The new architecture processes one sentence as a whole rather than token-by-token, which improves the ability of the model to remember the context and allows parallel computation which significantly reduces the training time.

In the following section, we will introduce the theoretical foundations behind the functioning of the Transformer models, as they will be used in our experiments throughout this thesis. We give a brief overview of the architecture, for more detailed information please refer to Vaswani et al. [2017].

3.2.1 Architecture

The Transformer was introduced as an encoder-decoder model intended for machine translation. For language modeling tasks it can also be used as a solo encoder or a solo decoder.

The encoder-decoder architecture is composed of a stack of encoders and a stack of decoders as illustrated in Figure 3.3. The role of the encoder is to process the source sentence and return a deep bidirectional representation vector for each token of the sentence. The role of the decoder is to process the encoded source sentence and generate a new one. In addition to the encoder representations of the source text, the decoder sees the target words it had already generated.

Encoder

The encoder encodes the input sentence of length len by passing it through a stack of encoder blocks. The depth of the model is governed by the number N of encoder blocks, each of which is composed of a multi-head self-attention layer with M heads and a feed-forward layer, with layer normalization after every layer and residual connections in between. Dropout [Srivastava et al., 2014] is applied before each layer normalization.

The dimensionality dim of the model is one of the model hyperparameters and it is the plength of the per-token vectors which flow between the blocks of the model. One encoder block is also sometimes referred to as one encoder layer which has two sublayers. Formally,

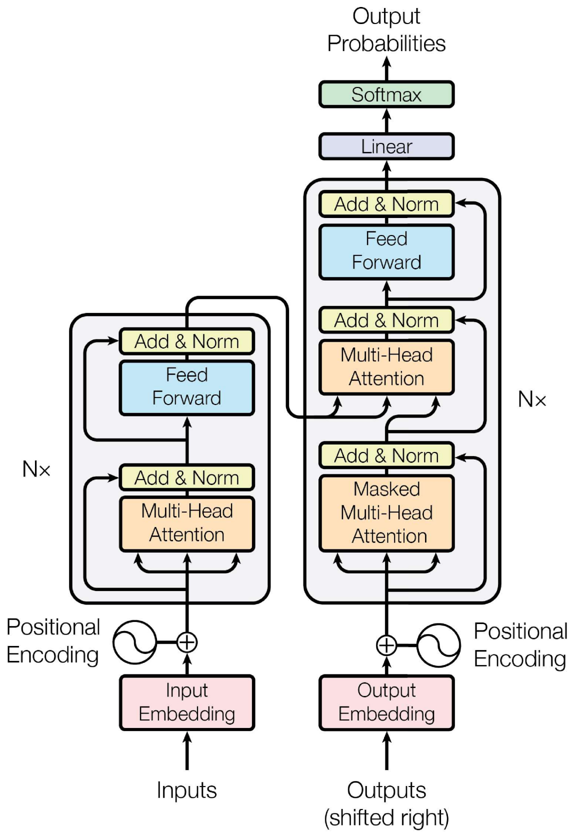


Figure 3.3: Illustration of the full Transformer encoder-decoder architecture.
Source: Vaswani et al. [2017]

the output of one encoder layer $\mathrm{Enc}(X)$ given the previous-layer sentence representation $X \in \mathbb{R}^{len \times dim}$ is calculated as

$$\mathrm{Enc}(X) = \mathrm{LayerNorm}(X' + \mathrm{FFN}(X)) \tag{3.3}$$

$$X' = \mathrm{LayerNorm}(X + \mathrm{MultiHeadAtt}(X)) \tag{3.4}$$

$$\mathrm{FFN}(X) = \Theta(XW_1 + B_1)W_2 + B2 \tag{3.5}$$

where $W_1 \in \mathbb{R}^{dim \times 4dim}$, $W_2 \in \mathbb{R}^{4dim \times dim}$ and their respective biases B_1, B_2 are the parameters of the feed-forward network whose hidden dimension is usually four times the model dimensionality dim. $\Theta(x)$ is a ReLU or GELU [Hendrycks and Gimpel, 2017] activation function. When calculating self-attention for the first encoder block, the previous-layer representation X refers to the input embeddings.

Decoder

The decoder typically has the same number of blocks (layers) N as the encoder. It has an almost identical structure to the encoder, but the decoder blocks include an additional multiheaded cross-attention layer in the middle that attends to the encoder representations of the source sequence.

The input to the decoder is the encoder output and the target sequence of tokens. The first layer is always an embedding layer enriched with positional encoding and optionally an additional sequence type embedding or a language embedding. The final decoder output is passed on to a linear layer with a softmax activation function over the output dictionary. The weight matrix of the linear layer can be thought of as an output embedding matrix and it was shown to be beneficial to tie it to the input embedding matrix and update the two together [Press and Wolf, 2017].

3.2.2 Input Embeddings

The input text stream is fed into the model as a sequence of tokens $(x_1, \ldots, x_{len})$ represented by their vocabulary indices. The first step the model performs is encoding the input tokens using a learned token embedding matrix $W^{TE} \in V \times dim$ where V is the vocabulary size.

Furthermore, as the Transformer model does not rely on any recurrence, the ordering of the sequence tokens must be modeled explicitly by **positional encoding**. It can be done either by learning a position embedding matrix $W^{PE} \in \mathbb{R}^{max\_len \times dim}$ where max_{len} is the maximum sequence length or by using parameterless sinusoidal encoding to calculate the values of W^{POS_EMB} according to

$$W^{PE}(pos; 2i) = \sin(pos/10000^{2i/d}) \tag{3.6}$$

$$W^{PE}(pos; 2i+1) = \cos(pos/10000^{2i/d}) \tag{3.7}$$

where pos is the position being encoded.

In multilingual tasks, it may be beneficial to provide the model with information about the language of the input sequence by embedding its language id using a language embedding matrix $W^{LE} \in \mathbb{R}^{nlangs \times dim}$ where $nlangs$ is the number of languages known to the model.

The final input embeddings $X \in \mathbb{R}^{len \times dim}$ are calculated as a sum of token embeddings, position embeddings, and language embeddings (if applicable).

$$\begin{aligned} X = \mathrm{Emb}((x_1 \ldots x_{len}), W^{TE}) + \mathrm{Emb}((1, \ldots, len), W^{PE}) + \\ + \mathrm{Emb}((lang, \ldots, lang), W^{LE}) \end{aligned} \tag{3.8}$$

For a sequence of non-negative indices seq, $\mathrm{Emb}(seq, W) \in \mathbb{R}^{len \times dim}$ refers to the output of an embedding layer defined by a lookup matrix W.

Finally, dropout is applied to the normalized input embeddings.

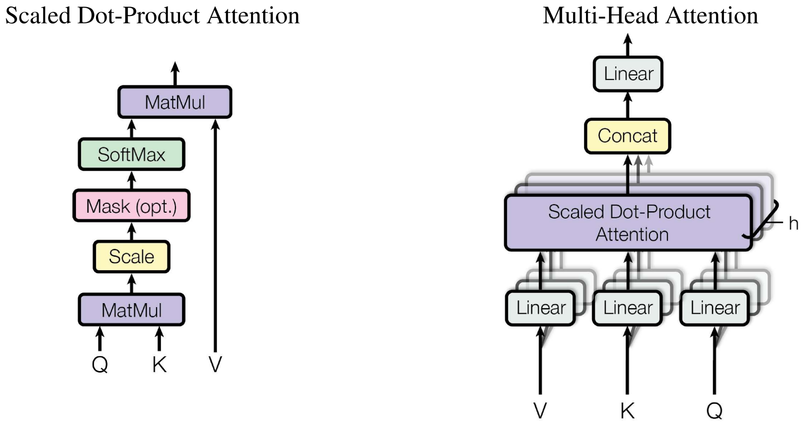


Figure 3.4: Visualization of the inner workings of the self-attention layers.
Source: [Vaswani et al., 2017]

3.2.3 Self-Attention

The key concept behind the Transformer architecture is the self-attention which is illustrated in Figure 3.4. The purpose of self-attention is to determine whether to use the information about token j while encoding or decoding token i. To that end, it needs to score each word of the input sentence against the current word to determine how much focus to place on other parts of the input sentence.

The attention layer is composed of three sets of matrices with dimensions $dim \times d_k$ that need to be trained: query matrix W^Q, key matrix W^K, and value matrix W^V. The embedding dimension of the model dim and the attention dimension d_k are hyperparameters. Multiplying a token representation vector X with these three matrices yields three new sets of vectors: queries (Q), keys (K), and values (V). The attention score is computed as the dot products of the query with all keys, divided by the square root of the length of the key vector d_k. Finally, softmax is calculated to obtain the probability weights on the value vector where a zero weight on a particular position means no information flow between the two tokens.

Formally, the calculation illustrated in Figure 3.4 is the following

$$
\begin{aligned}
&Z = \mathrm{Att}(Q, K, V) = \mathrm{softmax}(QK^T/\sqrt{d_k})V \\
&\qquad\qquad\qquad\qquad \text{where} \quad Q = XW^Q; K = XW^K; V = XW^V
\end{aligned}
\tag{3.9}
$$

where $X \in \mathbb{R}^{len \times dim}$ is the previous-layer representation of the sequence and $Z \in \mathbb{R}^{len \times d_k}$ is the attention representation of the sequence.

Transformer attention is modeled to have multiple *heads*, i.e. multiple sets of queries Q_i, keys K_i, values V_i, and their respective trainable matrices, each of which yields a new sequence representation Z_i in a separate subspace. The outputs are concatenated and projected again as illustrated in Figure 3.4. That way the model can capture multiple types of relationships between words, e.g. on the semantic or the syntactic level. For a number of heads M and a trainable matrix $W^O \in \mathbb{R}^{Md_k \times dim}$, the multi-head attention output is calculated as follows

$$\text{MultiHeadAtt}(X) = \text{Concat}(Z_0, \ldots, Z_M)W^O$$
$$\text{where} \quad Z_i = \text{Att}(XW_i^Q, XW_i^K, XW_i^V) \quad (3.10)$$

Transformer decoder uses *multi-head masked self-attention*. When decoding the word n of a target sentence of length len_{tgt}, the words $(n+1, \ldots len_{tgt})$ are masked to prevent the self-attention layer to consider information about tokens that have not yet been generated.

The information flow between the encoder and the decoder of a full Transformer model is facilitated by *multi-head cross-attention* layers. They work identically to the self-attention layers, only there are two inputs into each cross-attention layer – final encoder representations of the source (X^{enc}) and previous-layer decoder representations of the target (Y^{enc}). Intuitively, for each target word that is being generated, the cross-attention can attend to any source token that it finds relevant. Moreover, it can attend to different tokens in each head. The Equation 3.9 still applies but the calculation of the queries, keys, and values for $i \in (1, M)$ is the following

$$Q_i = Y^{enc}W_i^Q; K_i = X^{enc}W_i^K; V_i = X_i^{enc}W^V \quad (3.11)$$

3.2.4 Unsupervised Pre-Training

The Transformer architecture and the efficiency of its training allow pre-training on large amounts of unlabeled text data to learn the statistical patterns, relationships, and structures present in the language. Soon after the introduction of the Transformer architecture, big NLP players started publishing large-scale NLP models pre-trained on large amounts of non-annotated data (e.g. BERT [Devlin et al., 2018] by Google; GPT [Brown et al., 2020] by OpenAI; RoBERTa [Liu et al., 2019] or BART [Lewis et al., 2020] by Facebook AI). For many tasks across the NLP field, fine-tuning pre-trained models leads to state-of-the-art results with a fraction of resources [Devlin et al., 2018].

Pre-trained models are often publicly available to save the computation power needed for the pre-training stage and allow even smaller research groups to harvest the benefits of large-scale unsupervised pre-training. We present the most common pre-training strategies where training data can be trivially generated from raw monolingual texts. Unsupervised pre-training can be applied to only the encoder (e.g. BERT), only the decoder (e.g. GPT), or the entire encoder-decoder model (e.g. BART) and the training objectives differ accordingly as illustrated in Figure 3.6.

Causal Language Modeling (CLM) training objective can be used for both encoder-only models and decoder-only models. The task consists of modeling the probability of a word given the previous words in a sentence $P(w_t|w_1, \ldots, w_{t-1}, \theta)$ with model parameters θ. This is the traditional objective for language generation. During training, we optimize the maximum likelihood of the next word given the context. The model is able to attend to the left context of the masked word and never sees the right context with future words which have not yet been

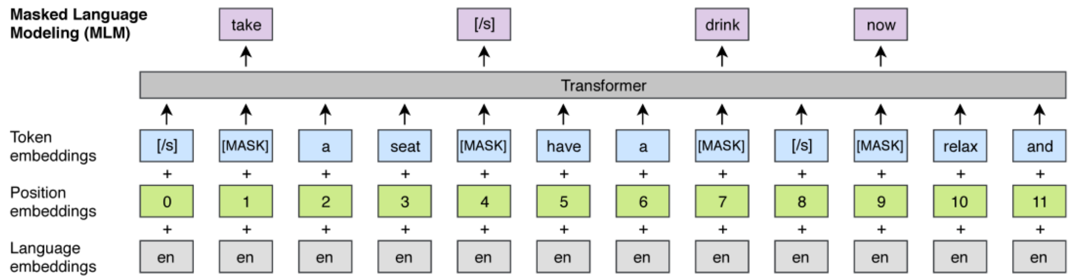


Figure 3.5: Cross-lingual language model design for training with the masked language modeling (MLM) objective.

Source: Conneau and Lample [2019]

generated. The training is usually performed on fixed-length text streams. The GPT family of pre-trained Transformer decoders uses the CLM pre-training objective.

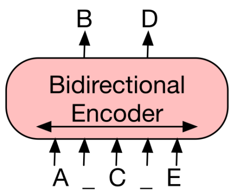
Masked Language Modeling (MLM) training objective is meant for encoder-based Transformer models where the model is trained to predict individual words rather than generate the full sequence. It encourages learning of a bidirectional context of words. It is inspired by the Cloze test on the readability of corrupted text [Taylor, 1953] commonly used in student assessment of learning a foreign language. Random tokens of a word sequence are masked and the task for the model is to fill in the missing tokens given the context. During MLM training used for BERT pre-training, 15% of tokens are randomly sampled to be either replaced by the [MASK] token (80% of time), replaced by a random token (10% of time), or not changed at all (10% of time). An extra head with a softmax linear layer is built on top of the encoder to select the most probable word from the vocabulary for each masked position. The training is usually performed on fixed-length text streams.

In contrast to a causal (left-to-right) language modeling objective, MLM relies on the bidirectional nature of a Transformer encoder. The bidirectionality is achieved by the self-attention layers where the encoder sees both the left-hand-side and the right-hand-side context of the masked word. The BERT family of pre-trained Transformer encoders uses the MLM pre-training objective.

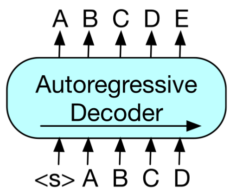
Denoising Autoencoding (DAE) is a training strategy meant for pre-training the entire encoder-decoder model. It was proposed by Vincent et al. [2008] and later customized for NLP by Lample et al. [2018a] and Lewis et al. [2020] who pre-trained and published the popular BART model. Denoising autoencoding entails corrupting the input with a specific noise and training the model to recover the original. The purpose of the input noise is to encourage the model to internally create a high-level representation of the text by simulating a situation where meaning needs to be preserved while the input cannot be trivially copied.

The following strategies can be used in the noise function

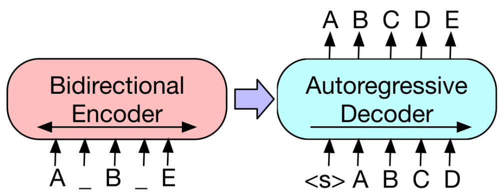
1. token masking, where random tokens are masked with the [MASK] token;



(a) BERT: Random tokens are replaced with masks, and the document is encoded bidirectionally. Missing tokens are predicted independently, so BERT cannot easily be used for generation.



(b) GPT: Tokens are predicted auto-regressively, meaning GPT can be used for generation. However words can only condition on leftward context, so it cannot learn bidirectional interactions.



(c) BART: Inputs to the encoder need not be aligned with decoder outputs, allowing arbitary noise transformations. Here, a document has been corrupted by replacing spans of text with a mask symbols. The corrupted document (left) is encoded with a bidirectional model, and then the likelihood of the original document (right) is calculated with an autoregressive decoder. For fine-tuning, an uncorrupted document is input to both the encoder and decoder, and we use representations from the final hidden state of the decoder.

Figure 3.6: Schematic comparison between BERT, GPT and BART models.
Source: Lewis et al. [2020]

2. token deletion, where random tokens are deleted;
3. text infilling, where random sequences of different lengths (sampled from the Poisson distribution with $\lambda = 3$) are sampled and replaced by [MASK] token (for 0-length sequences, [MASK] token is inserted);
4. token shuffling, where a random permutation within a specified window length is applied to the input sentence
5. sentence permutation, where a random permutation is applied to sentences within one training sample
6. document rotation, where the initial token is selected randomly from the training sample and put at the start, moving the preceding tokens at the end of the model

Lample et al. [2018a] use token deletion and token shuffling; Lewis et al. [2020] use all strategies except for token shuffling and report a crucial role of token masking and token deletion, and poor performance of sentence permutation and document rotation.

3.2.5 Multilingual Pre-Training

The unsupervised training described in the previous paragraphs can also be performed multilingually. The multilingual BERT (mBERT) and XLM [Conneau and Lample, 2019] were trained as the multilingual versions of BERT on the entire Wikipedia dump. XLM-R [Conneau et al., 2019] was trained as the multilingual version of RoBERTa on the large CommonCrawl corpus. Multilingual pre-trained models are immensely popular for their multilingual text representations as well as their capabilities to transfer downstream task knowledge to new languages.

The language id information can be passed to the model by an initial extra language id token (e.g. mBART) or via the language embedding layer (e.g. XLM) but some models treat all text the same, regardless of the language. We will give more details on multilingual pre-training in Chapter 6.

3.2.6 Internal Representations

Internal representations from large language representation models are a valuable source of information on the inner functioning of the Transformer models. Furthermore, they can be extracted and used as contextual embeddings for various purposes.

A sentence is processed by a Transformer encoder as a sequence of tokens and the encoder representations of each token can be understood as its contextual embeddings. The contextual character of the embedding is reached by the self-attention layer which enriches each token vector with the information about the surrounding words. Such enrichment occurs in every encoder block. The enriched embeddings are normalized and processed through a feed-forward network before they are passed to the next block.

Contextual embeddings can be retrieved from any layer of any pre-trained Transformer model. Jawahar et al. [2019] show that different encoder layers represent different linguistic phenomena. They conclude that surface and syntactic features lie on the bottom and middle layers, while semantic features of words lie on the top layers.

3.3 Machine Translation

In the early days of natural language processing, machine translation was approached using a great number of hand-crafted rules designed to cover the extremely complex nature of translation known to human translators. Later in 1990s, it was replaced by data-driven approaches which use machine learning techniques to teach the model directly from a large corpus of pre-translated texts.

Before 2014, the standard approach to MT was statistical PBMT, where n-grams in the source and target languages were modeled and aligned based on their number of common occurrences. The advent of neural networks lead to a dramatic change in the MT field and a complete change of paradigm from statistical phrase-based systems to neural encoder-decoder models [Bahdanau et al., 2015, Sutskever et al., 2014]. In 2017, the state-of-the-art in MT was reached by the Transformer architecture which replaced recurrent neural models. In Section 3.2.4, we introduced the unsupervised training strategies for Transformer models. Here we will describe how they can be trained for supervised machine translation.

3.3.1 Neural Machine Translation

NMT models are sequence-to-sequence models which utilize neural networks to learn the mapping between the source and the target language. They model the task of MT in an end-to-end fashion relying only on sentence-aligned parallel texts with no hand-crafted features or specialized modules. Different neural model architectures are possible but in this thesis, we work exclusively with the Transformer models as presented in Section 3.2.1.

From the research point of view, NMT includes three main questions: how to design the network architecture, how to train it, and how to use it for inference. In this thesis, we rely on the state-of-the-art design and inference techniques for supervised MT and we contribute novel approaches on how to train the model parameters from monolingual data only. Supervised NMT training is introduced in this section to provide the foundations of our work, while the specifics of unsupervised MT training will be described later in Chapter 6.

Tokenization and Vocabulary

NMT is an open vocabulary problem that needs to be solved with a fixed-size vocabulary defined prior to the training. The right balance between the flexibility offered by a large vocabulary and the constraint posed by the model capacity can be struck using one of the existing subword approaches. In contrast to using complete word tokens, employing subword units reduces the size of the vocabulary and eliminates the occurrence of unknown words in the translated output.

Subword-based tokenization first segments the input texts into a group of characters that do not necessarily correspond to full words. A fixed vocabulary of subword units and individual characters ensures that rare words can be represented by the model rather than being tagged unknown, although they might be treated merely as a list of characters. Although the subword units are created algorithmically without any hand-crafted rules, sometimes they reflect the morphological structure of a word.

The BPE algorithm [Sennrich et al., 2016] is a data compression algorithm originally described by Gage [1994]. When applied to text data, it iteratively replaces the most common pair of consecutive characters with a new symbol that does not occur in that data. This procedure is repeated for a given number of iterations or until a pre-defined vocabulary size is reached. Eventually, the most frequent words are represented as a single token while rare words are split into several more common subword units. The algorithm can be applied to the concatenation of the source and the target corpora to obtain a shared vocabulary of subwords.

Embeddings

It was explained in Section 3.1 that machine learning models work with numbers rather than words. The same applies to NMT models which need to first assign a numerical vector (static embedding) to each token of the vocabulary to be able to process the tokenized text. NMT models create their own fixed embeddings in the initial layer, known as the embedding layer. This layer assigns a learnable dense vector to each word in the vocabulary and these vectors are updated throughout the training process. In Transformer systems, the input and output embeddings are usually shared which requires a shared vocabulary for the source and target languages.

Architecture

The state-of-the-art MT architecture is the encoder-decoder Transformer which was described in Section 3.2.1. The most commonly used architectures are *base* (6 layers in both the encoder and decoder, 8 self-attention heads with dimension $d_k = 64$, embedding size 512, and hidden size 2048) and *big* (6 layers in both the encoder and decoder, 16 self-attention heads with dimension $d_k = 64$, embedding size 1024, and hidden size 4096).

Training

Supervised machine translation is trained on pairs of parallel sentences with a cross-entropy training objective, where the model is penalized every time it predicts a different word than the reference translation. The loss over the parallel corpus D is defined as follows

$$L(\theta_{\text{enc}}, \theta_{\text{dec}}) = -\sum_{(x,y)\sim D} \sum_{i=0}^{|y|} log(\hat{p}(y_i)) \tag{3.12}$$

where $(\theta_{\text{enc}}, \theta_{\text{dec}})$ are the trained model parameters, (x, y) is a sentence pair sampled from the parallel data set D, and $\hat{p}(y_i)$ is the predicted probability of token y_i. The model is trained to minimize the negative log-likelihood over the training corpus

$$\theta^*_{\text{enc}}, \theta^*_{\text{dec}} = \underset{\theta_{\text{enc}},\theta_{\text{dec}}}{\operatorname{argmin}} L(\theta_{\text{enc}}, \theta_{\text{dec}}) \tag{3.13}$$

using stochastic gradient descent (SGD) with adaptive learning rate (Adam) [Kingma and Ba, 2015].

Back-translation

Back-translation is a data augmentation method for MT that allows using monolingual texts to synthesize a parallel corpus and expand the translation training data [Sennrich et al., 2016]. It uses the trained MT model to translate monolingual texts, thereby creating an additional parallel corpus to be used for further training of the model. The customary practice is to utilize the synthetic side of the corpus as the source input to the model. It was shown that several iterations of back-translation can significantly improve the results. The unsupervised MT greatly relies on the concepts of back-translation. More details on the specifics of the unsupervised training will be given in Chapter 6.

Decoding

When using a trained model for decoding, we generate tokens autoregressively based on the output probability distribution given the input sentence x. The optimal way would be to find a translation with the highest probability. However, the search space for finding the candidate translation is large and expands with new hypotheses after generating each new candidate token. Therefore, local search algorithms are used to reduce the search space. The *greedy search* algorithm always selects the next token with the highest probability and does not revise its choices. The *beam search*, on the other hand, keeps track of the most promising candidates and prunes less likely ones as the decoding progresses. It remembers b previous hypotheses and expands them with b most likely states until the expanded sentence ends or the maximum length is reached. The final translation is the one with the highest probability. In this work, we use beam search with beam size $b = 4$.

3.3.2 Phrase-Based Machine Translation

In contrast to the end-to-end nature of NMT, statistical phrase-based systems rely on several modules to take care of the translation modelling task. Each module is estimated based on phrase occurrences and alignments from the parallel corpus. Although PBMT systems were replaced by NMT models for standard MT applications, they can still prove useful in low-resource conditions and for translation from monolingual data only. It has been shown [Artetxe et al., 2019a] that it is possible to infer a phrase table in a completely unsupervised way and build a PBMT system around it. Therefore, we briefly introduce the phrase-based systems here as well.

A PBMT model [Koehn et al., 2003] is a log-linear probability model that captures the probability of the target sentence being the translation of the source sentence. To estimate this model, input texts are aligned at the token level using a specific tool, e.g. GIZA++ [Och and Ney, 2003], divided into phrases (n-grams), and assembled into a phrase table along with their frequencies estimated from the parallel training corpus. The log-linear model incorporates the following components:

- phrase translation probability (estimated based on the number of times a phrase pair was observed in the aligned parallel corpus);
- language model (estimated based on the frequencies of individual n-grams observed in the source and target corpora and their backoff probabilities [Katz, 1987]);
- distortion model (penalizing candidate translations with excessive word reordering);
- word/phrase count penalty (balancing overall sentence length and the number of phrases it is composed of).

Each of the features above is complemented by a default weight before entering the model. The weights are tuned using the Minimum Error Rate Training (MERT) [Och, 2003] to maximize the BLEU score of translation quality on a small set of parallel sentences (development set).

Formally, the probability of a sentence tgt being the translation of a sentence src is the following

$$p(tgt|src) = \frac{\exp \sum_i \lambda_i f_i(tgt, src)}{\sum_{tgt'} \exp \sum_i \lambda_i f_i(tgt', src)} \tag{3.14}$$

f_is are the features listed above, λ_is are the feature weights, and tgt' iterates over all possible translation candidates.

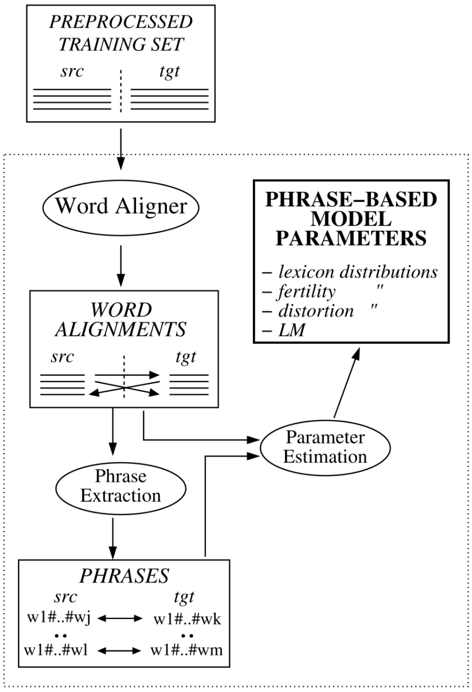


Figure 3.7: Training of an PBMT model: estimation of bidirectional word-alignment, phrase extraction, estimation of phrase-based features

Source: Cettolo et al. [2005]

When training the model, the training data is first tokenized, truecased and aligned. Individual features of the model are then statistically estimated from the training data set. Finally, the feature weights are tuned to maximize the translation quality on a development data set. In the decoding phase, beam search is employed to produce the most likely sentence by combining translation candidates for individual phrases, considering their log-probability scores.

The Moses [Koehn et al., 2007] toolkit with external language modelling tools is used for PBMT model training and decoding.

3.3.3 Machine Translation Evaluation

Machine translation is evaluated using a combination of automated metrics and human evaluations. In this thesis we use the following automatic metrics for evaluation, namely BLEU, COMET and chrF++. Manual evaluation is used for qualitative analysis of the translations.

BLEU Score

Automated evaluation of machine translation output quality can be accomplished using the BLEU metric [Papineni et al., 2002] which assesses the candidate translation by comparing it to the reference translation and assigning a score based on the number of overlapping word n-grams of order 1 up to N. While BLEU has its limitations mostly due to the fact that there is never a single correct translation, it has shown a sufficient correlation with human judgment and it is widely utilized for MT evaluation.

BLEU is calculated as

$$BLEU = BP \cdot e^{\sum_{n=1}^{N} \lambda_i \log p_i} \tag{3.15}$$

where $N = 4$ is the order of the longest considered n-gram, $\lambda_i = 1/N$, p_i is the modified n-gram precision and BP is the brevity penalty defined as

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{1-r/c} & \text{if } c \leq r \end{cases} \tag{3.16}$$

where r and c are the number of tokens in the reference and candidate translation, respectively.

chrF++

The character n-gram F-score (chrF++) by Popović [2017] is another automated metric used for evaluating the quality of translation. It measures the similarity between a machine-generated translation and one or more reference translations based on the combination of a character-level n-gram overlap and a word-level n-gram overlap.

The chrF metric [Popović, 2015] was originally designed to address some of the limitations of other automated metrics like BLEU which exclusively focus on word-level n-gram overlap. Since chrF operates at the character level, it can more adequately assess languages with complex morphology, languages with agglutinative or inflected forms, and languages with significant word order variations. Popović [2017] introduced an improved chrF++ metric by integrating a word-level overlap score.

For both character-level n-grams and word-level n-grams, the calculation of F-score in Equation 3.17 is based on the percentage of n-grams from the reference covered by the hypothesis (n-gram recall *ngrR*) and the percentage of n-grams from the hypothesis covered by the reference (n-gram precision *ngrP*). Character n-grams may exceed word boundaries as spaces are ignored.

$$ngrF = (1 + \beta^2) \frac{ngrP \cdot ngrR}{\beta^2 ngrP + ngrR} \tag{3.17}$$

The values of *ngrR* and *ngrP* are averaged over all n-grams from $n = 1$ to N where the default setting is $N = 2$ for words and $N = 6$ for characters. The parameter β gives higher importance to recall over precision and is commonly set to $\beta = 2$. The word-level F-score and the character-level F-score are averaged to produce the final chrF++ score.

COMET

COMET (Cross-lingual Optimized Metric for Evaluation of Translation) by Rei et al. [2020] is a framework for training MT evaluation models that can function as metrics. It comprises neural models designed to predict human evaluation of MT quality and thus overcome the problem of automated metrics (e.g. BLEU or chrF++) that do not adequately correlate with human judgment. COMET provides scores ranging from 0 to 1 with a value of 1 signifying a perfect translation. We use the model trained on the Direct Assessment (DA) [Graham et al., 2015] scores as collected in WMT22 (`wmt22-comet-da`).

Bootstrapping

Bootstrapping is a statistical resampling technique that can be used to evaluate machine translation systems with statistical confidence [Koehn, 2004]. The bootstrapping process entails the following steps.

1. Randomly selecting N translations from both the MT output and the reference translations, with replacement (resampling) where N is the size of the original test set.
2. Calculation of the evaluation metric (e.g., BLEU) for the resampled set of translations.
3. Repeating the resampling and metric calculation process multiple times (we repeat 1,000 times) to generate a distribution of metric scores.

We can calculate confidence intervals from the distribution of metric scores obtained through bootstrapping, These intervals provide an estimate of the range within which the true metric score is likely to lie which helps in assessing the reliability of the evaluation.

Bootstrapping helps account for variability in evaluation metrics due to the randomness in the selection of sentences and translations. It provides a more robust understanding of the machine translation system's performance and can be particularly useful when the evaluation dataset is limited or when traditional statistical assumptions might not be met.

In this thesis, we use bootstrapping for the BLEU and the chrF++ calculation. We set the number of bootstrap resamples to 1,000.

4. Related Work

To organize the related work in the area of UMT, we devise a taxonomy that maps the approaches. We categorize the methods into model-centric and data-centric, following the conventional approach in domain adaptation models. Model-centric approaches focus on the particularities of the system design and architecture, initialization of the model parameters, training objectives, and decoding strategies. Data-centric approaches focus on the data that are used for training the system, e.g. multilingual data, mined pseudo-parallel data, or back-translated synthetic data. Figure 4.1 illustrates our taxonomy of unsupervised approaches.

4.1 Model-Centric Approaches to UMT

Unsupervised machine translation was first approached by Artetxe et al. [2018d] and Lample et al. [2018c]. They proposed unsupervised training techniques for both PBMT and NMT to extract all necessary translation information from monolingual data. What followed was an overflow of new ideas and improvements upon the initial work which will be listed in the following sections.

4.1.1 Model Architecture

Phrase-based Models

A bilingual lexicon can be induced from a bilingual embedding space created without parallel data (Section 6.1). The simplest form of unsupervised translation is a word-by-word translation using such a bilingual lexicon. Kim et al. [2018] propose improving unsupervised word-by-word translation by integrating surrounding context with a language model.

Lample et al. [2018c], Artetxe et al. [2018b] propose unsupervised methods for creating a full PBMT system. In the absence of parallel training data, the initial phrase table is induced from a cross-lingual n-gram embedding space obtained by unsupervised post-hoc alignment of

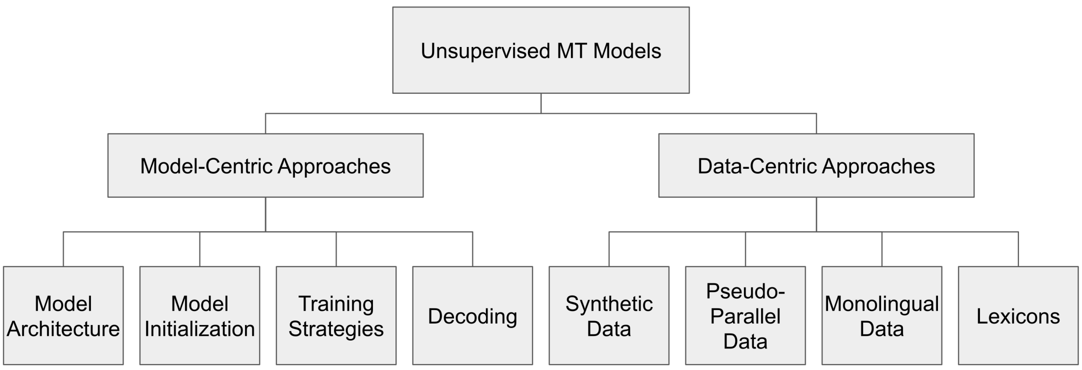


Figure 4.1: Taxonomy of UMT models.

monolingual embedding spaces [Conneau et al., 2018a, Artetxe et al., 2018c]. The translation probabilities are approximated from the cosine distances of candidate n-grams in the cross-lingual embedding spaces. The authors create MT systems in both directions to allow further improvements by back-translating the monolingual training corpora. Artetxe et al. [2018b] also use back-translated data to tune the hyperparameters of the PBMT model while Lample et al. [2018b] use their default values. Artetxe et al. [2019a] improve the training by adding subword information to training the cross-lingual embeddings. Furthermore, they propose an improved strategy for tuning the hyperparameters.

Artetxe et al. [2019b] use an existing PBMT system to extract a bilingual lexicon from back-translated data and conclude that the word translation accuracy is higher than simply searching for word pairs in the original cross-lingual embedding space.

Neural Models

The unsupervised NMT models have an encoder-decoder architecture. In order to produce language-neutral representations, they are designed to share parameters for both language directions. Artetxe et al. [2018d] employ a single encoder and language-dependent decoders while Lample et al. [2018a] share both the encoder and the decoder, with the only language-dependent component of the network being the embedding matrices. Conneau and Lample [2019] use a joint vocabulary for the source and the target language and share even the embeddings, following the multilingual MT design of Johnson et al. [2017]

Other authors go beyond the vanilla encoder-decoder structure. Li et al. [2020b] use a pre-trained cross-lingual model (XLM) as an additional encoder. They let their NMT model interact with the XLM encoder representations using the attention mechanism in all layers of both the encoder and the decoder. Üstün et al. [2021] propose to use denoising adapters – adapter layers with a denoising objective which are placed on top of a pre-trained multilingual denoising autoencoder and trained separately on monolingual data. The cross-attention of the model is also trained separately on an auxiliary parallel corpus. The approach is modular and allows to incrementally incorporate new languages. It requires auxiliary parallel data but it is unique in that it completely relieves the model from the computation burden of back-translation.

4.1.2 Model Initialization

Both unsupervised NMT and PBMT models require to initialize model parameters to some meaningful values as opposed to random initialization, in order to start the training.

Pre-Trained Cross-Lingual Embeddings

Lample et al. [2018a], Artetxe et al. [2018d] initialize their neural system with pre-trained embeddings trained on monolingual corpora and aligned in an unsupervised way. Lample et al. [2018b] pre-train the embeddings on a concatenation of the monolingual corpora without an explicit bilingual alignment and report the benefits of this pre-training strategy, especially for languages that share a significant number of BPE units.

Unsupervised PBMT models can be initialized with a phrase table induced from pre-trained cross-lingual embeddings [Artetxe et al., 2018b, Lample et al., 2018b] or with a phrase table extracted from a pseudo-parallel corpus [Ren et al., 2020].

Cross-lingual word embeddings can be obtained by post-hoc alignment of monolingual word embeddings using a linear mapping relying on the assumption of isomorphic embedding spaces, as discussed in Chapter 3. Aside from a range of supervised methods to learn the mapping matrix, some approaches are completely unsupervised and will be discussed in more detail in Chapter 6. Zhang et al. [2017] and Conneau et al. [2018b] align monolingual embedding spaces through adversarial training. Artetxe et al. [2017] propose an alternative method to learn the linear mapping using the assumption that digits are preserved across languages. Artetxe et al. [2018c] exploit the structural similarity of embedding spaces and iteratively improve the mapping through self-learning.

Chen and Cardie [2018], Heyman et al. [2019], Wada et al. [2019], Jawanpuria et al. [2020] extend the bilingual embedding approaches to the multilingual setup, leveraging the interdependencies between language pairs. Chen and Cardie [2018] employ a series of language discriminators to learn the mapping of N languages into a single space in the framework of adversarial training and further enhance the alignment using an iterative refinement approach of Artetxe et al. [2018c]. Jawanpuria et al. [2020] first induce bilingual lexicons from unsupervised word embedding spaces and use them as supervision for learning a mapping into the multilingual word embedding space. Heyman et al. [2019] propose a strategy that makes the training more stable even for distant languages as they train a multilingual model and add new languages incrementally one by one. They argue that existing multilingual approaches use one hub language without exploiting interdependencies between all languages which leads to suboptimal results especially when working with a language that is distant from the hub language.

Søgaard et al. [2018], Ormazabal et al. [2019], Patra et al. [2019], Vulić et al. [2020] question the use of the mapping approaches in situations when languages and/or domains are dissimilar and their embedding spaces are not isomorphic. Vulić et al. [2019] question the necessity of completely unsupervised approaches.

Wada et al. [2019] loosen the assumption of approximately isomorphic embedding spaces and obtain multilingual word embeddings from a multilingual bidirectional LSTM language model trained separately for each language but with parameter sharing. Mohiuddin et al. [2020] propose a semi-supervised method for non-linear mapping of two independently trained autoencoders in the latent space which also allows them to depart from the assumption of language isomorphism. Nishikawa et al. [2021] argue that learning monolingual embeddings from back-translated corpora generated by a UMT system creates embedding spaces which are approximately isomorphic and report improvement in the task of bilingual lexicon induction as well as other downstream tasks. Cao et al. [2023] integrate features from the source embeddings into the target embeddings to increase the geometric similarity of the two embedding spaces.

Pre-Trained Encoders

Conneau and Lample [2019] take the pre-training of model parameters one step further and pre-train a full encoder with the MLM or CLM objective and copy the weights into the encoder as well as the decoder of the NMT model. They conclude that the MLM strategy brings greater improvement in translation quality. Ren et al. [2019a] propose an MLM pre-training method with an explicit cross-lingual signal. They construct code-switching sentences by randomly choosing source n-grams in the input text stream and replacing them with their translation counterparts from an unsupervised phrase table. They train an encoder to predict the translated segments. Chronopoulou et al. [2021] use cross-lingual subword embeddings to enhance the bilingual MLM pre-training with lexical-level information and report a significant improvement over the baseline trained without the enhancement. Using an entirely different approach, Li et al. [2021] rely on Chomsky's universal grammars to find syntactic similarities between two languages and obtain a weak source of additional signal to the unsupervised training. They pre-train the encoder on the MLM task enhanced with constituent syntax information.

Pre-Trained Encoder-Decoder Models

Song et al. [2019] argue that pre-training only the encoder is not optimal for sequence-to-sequence models and propose a full encoder-decoder framework pre-trained to reconstruct a sentence from its corrupted version where a sentence fragment is masked. The MASS model is presented with the masked sequence and it is taught to generate the full original sentence. Similarly, Liu et al. [2020] pre-train the entire model on the task of denoising autoencoding where the model is taught to reconstruct the original text stream from its noised input, where the noising function includes masking of sentence fragments and sentence permutation. Li et al. [2020a] pre-train the model on the task of explicit sentence compression (ESC) where extra tokens are sampled from the corpus to create additive noise that makes the sentence longer. The tokens of the extended input sentences are shuffled and the model is trained to recover the original, compressed version of the noised sentence. Li et al. [2020b] conclude that the ESC pre-training is on par with MLM pre-training and superior to CLM pre-training.

Baziotis et al. [2021] find that unlike supervised MT systems, UMT systems are very sensitive to noising strategies used during pre-training. Masking strategies lead to a significantly higher performance than shuffling strategies.

Multilingual Pre-Training

Liu et al. [2020] pre-train a large multilingual model on texts in 25 (mBART) or 50 (mBART-50) languages which can be fine-tuned for a specific language pair with state-of-the-art results.

Transfer Learning from Parallel Data

Successful transfer of MT abilities from high-resource language pairs to low-resource language was demonstrated by Kocmi [2020], Zoph et al. [2016], Kim et al. [2019], suggesting that translation has some universal nature that goes beyond generating text in a particular language.

Li et al. [2020b] and Garcia et al. [2020] adapt the approach to the unsupervised setting and use transfer learning to pre-train an NMT system on an auxiliary language pair and fine-tune it in an unsupervised way using back-translation.

4.1.3 Training Strategies

Most unsupervised training strategies rely on a combination of different training objectives and most require some form of back-translation for training. One exception is found in the work of Üstün et al. [2021], who, however, rely on auxiliary parallel data. In the following paragraphs, we list the training strategies used by different authors.

Iterative Training

The iterative training strategy is employed in approaches where the training data is generated by the model being trained, either by online back-translation, or online sentence selection. The quality of the training thus increases as the training progresses.

Lample et al. [2018a] and Artetxe et al. [2018d] propose online back-translation, where a mini-batch of sentences is translated by the emergent NMT model and it is immediately used for training the model in the opposite translation direction, all in one training step.

Other authors select training samples by online parallel sentence mining. Ruiter et al. [2019] use the encoder of the NMT model for incrementally finding cross-lingually similar sentences in the monolingual training corpora and train the NMT model on the retrieved sentences as soon as one training batch is complete. Tran et al. [2020] iteratively train the multilingual mBART model on translation and sentence selection to enhance representation alignment in the course of MT training.

Adversarial Training

Lample et al. [2018a] use the adversarial loss during unsupervised NMT training to induce shared encoder representations but they drop it in Lample et al. [2018b] and train only using iterative back-translation and denoising autoencoding. Yang et al. [2018] also enforce the shared encoder latent space by adversarial training.

Rather than relying on back-translated synthetic sentences, Wu et al. [2019] extract translation candidates from the target monolingual corpus and employ a simple editing mechanism to bring the extracted target sentence representation closer to the source sentence. They do not use the extracted translation candidates as ground truth for MT training directly but rather view them as anchor points that the translated sentence should be close to. They train the translation model together with an evaluation network that assesses the similarity of the extracted sentence pairs to the source sentence using an adversarial approach. The goal of the translation model is to generate a translation with a higher similarity score than the extracted-and-edited candidates and the model plays a minimax game with the evaluator network to reach that goal.

Conneau et al. [2018a] use adversarial training for mapping monolingual embeddings into the cross-lingual space. Hartmann et al. [2019] survey existing unsupervised cross-lingual word embedding techniques and suggest that despite their inherent instability, generative adversarial networks possess the greatest potential for generating valuable seed dictionaries.

Reference Agreement Translation

Garcia et al. [2020] propose a novel cross-translation loss term that enforces cross-language pair consistency utilizing not only monolingual data but also an auxiliary parallel corpus for a related language pair. They show that adding one more language to the training framework can lead to improvements in BLEU scores over state-of-the-art unsupervised models. Wang et al. [2021] propose indirect supervised training using auxiliary parallel data as well as synthetic data forward-translated and back-translated via a third language. Li et al. [2020c] propose a reference language-based framework where they leverage a parallel corpus that the source language has with a third language. They train two models (source to target and reference to target) to translate the parallel source and reference sentences into the target language and combine them to generate an *agreed-upon translation* which is used as the ground truth for the next iterations of translation training. The same translation pairs can also be used to train opposite models in a back-translation framework. The authors report a significant improvement over the systems which do not use the reference language pair as well as over a system pre-trained on the reference language pair and fine-tuned on back-translation.

Reinforcement Learning

Wang et al. [2021] train a UNMT model under the reinforcement learning framework with a reward function that praises the model for producing translations for a high number of n-gram matches and semantic adequacy.

Meta-Learning

Park et al. [2021] explore domain adaptation within UMT by using meta-learning. The objective of meta-learning in MT is to find the optimal parameter initialization that would allow the model to quickly adapt to a new domain even with only a small amount of in-domain monolingual data. They enhance the vanilla meta-learning model by using a cross-domain loss to encourage the model to be able to generalize well to another domain. They report a significant margin of the meta-learning algorithms over domain adaptation via transfer learning.

4.1.4 Decoding Strategies

The specifics of low-resource MT can also be tackled at test time. If auxiliary parallel texts are available and there exists a pivot language that has parallel data both with the source and the target, source-to-target translation can be performed in two steps using two standard supervised MT models: source-to-pivot and pivot-to-target. It is important to note that using pivot translation introduces an additional step in the translation pipeline, which may lead to compounding errors and potentially reduce translation accuracy. The choice of a suitable pivot language is also crucial as it can greatly impact the overall translation quality. Leng et al. [2019] hypothesize that translating between distant languages is easier to learn via a pivot than directly. They train multiple unsupervised NMT systems and conclude that a majority of the distant language pairs indeed require a pivot or even multiple pivots to achieve a higher translation quality. They further propose a strategy for finding the optimal pivoting route from the source to the target language.

Pourdamghani et al. [2019] introduce another two-step translation approach where the mid-step is a synthetic language called Translationese – rough word-by-word translation of source texts obtained using unsupervised source-to-target dictionaries. An MT system is trained on auxiliary parallel data to translate from Translationese into a fluent target language and it can be applied to any source language at test time, provided that an unsupervised dictionary is available.

4.2 Data-Centric Approaches to UMT

Unsupervised training of an MT system is always at least partially data-centric – the training data is synthesized from the monolingual texts which are available or they are mined from the monolingual corpora. Alternatively, multilingual or auxiliary parallel data in other languages are used. In this section, we list the works which introduce a novel method for obtaining the training data.

4.2.1 Pseudo-Parallel Data

Ren et al. [2020] build a pseudo-parallel corpus by retrieving semantically comparable sentences from monolingual corpora and rewriting the target side to get rid of unaligned words and minimize the semantic gap. The state-of-the-art approaches to parallel corpus mining are based on a similarity retrieval of sentence embedding vectors using a margin-based scoring of translation candidates [Artetxe and Schwenk, 2019b].

Most models rely on heavy supervision by parallel corpora for the embedding. Kvapilíková et al. [2020], Keung et al. [2020] show that it is possible to mine sentence pairs without having any parallel texts to start with by using unsupervised multilingual sentence embeddings from a pre-trained Transformer language model. Hangya and Fraser [2019] use word similarity scores for parallel sentence mining, while controlling the length of aligned continuous parallel segments detected in sentence pair candidates to adjust for the fact that sentences with similar words may carry different meanings. Ruiter et al. [2019] mine parallel sentences on-the-fly during translation training using the internal encoder states of the unsupervised model as sentence embeddings. Hangya and Fraser [2019], Ruiter et al. [2021], Kvapilíková and Bojar [2022] integrate mined sentences into UMT training and report improvements over unsupervised baselines.

Earlier work in the area of monolingual sentence representation [Arora et al., 2017, Wieting et al., 2016] shows that averaging static word embeddings is a simple but strong baseline for creating sentence-vectors. Kiros et al. [2015] adapt the Skip-gram [Mikolov et al., 2013a] word embedding model for sentences (SkipThought) and train an LSTM model to reconstruct surrounding sentences of an encoded passage. Cer et al. [2018] train a universal Transformer encoder on a variety of downstream tasks including SkipThought and text classification. Conneau et al. [2017] obtain sentence embeddings from the supervised task of natural language inference (NLI) and argue its superiority over unsupervised methods. Pagliardini et al. [2018] propose a Sent2Vec model composing embedding vectors of individual words and n-grams contained in the sentence.

Schwenk and Douze [2017], Schwenk [2018], España Bonet et al. [2017] derive sentence embeddings from internal representations of a neural machine translation system with a shared encoder. The top performance in parallel data mining is currently achieved by LASER [Artetxe and Schwenk, 2019a], a multilingual BiLSTM model sharing a single encoder for 93 languages trained on parallel corpora to produce language-agnostic sentence representations. LASER has been successfully used to mine billions of sentence pairs from the web [Schwenk et al., 2019]. Reimers and Gurevych [2020] show how to change monolingual sentence embeddings into multilingual using knowledge distillation. Heffernan et al. [2022] use the proposed approach to extend LASER to unseen languages.

The universal sentence encoder (USE) [Cer et al., 2018, Yang et al., 2020] family covers sentence embedding models with a multi-task dual-encoder training framework including the tasks of question-answer prediction or natural language inference. Guo et al. [2018] directly optimize the cosine similarity between the source and target sentences using a bidirectional dual-encoder. Yang et al. [2019] enhance the model with an *additive margin softmax* loss to separate translations from nearby non-translations.

An entirely different (and possibly unsupervised) approach is to construct sentence representations by aggregating cross-lingual word embeddings either by simple averaging [Arora et al., 2017] or using an IDF-weighted average [Litschko et al., 2019]. However, since the mapping is applied to static (non-contextualized) embeddings, this strategy gives up on the contextual information which could be exploited in the sentence representation construction.

4.2.2 Synthetic Data

Synthetic Data from PBMT

Training an NMT model entirely on data from a PBMT system is not a good idea because the quality of the PBMT translations greatly influences the final translation quality. However, the initial cross-lingual signal into the unsupervised NMT model may come from an unsupervised phrased-based model. Unlike the previous initialization approaches based on weights initialization, the signal is passed to the model in the form of the initial synthetic parallel corpus intended for the first stage of the training. Kvapilíková et al. [2019], Stojanovski et al. [2019] use a phrase-based model to translate monolingual sentences and train a neural model on the synthetic samples. Artetxe et al. [2019a] first train their neural models exclusively on the synthetic parallel corpora generated by a phrase-based system and as the training progresses, they adaptively mix in the translations produced by the emergent neural models. Ren et al. [2020]

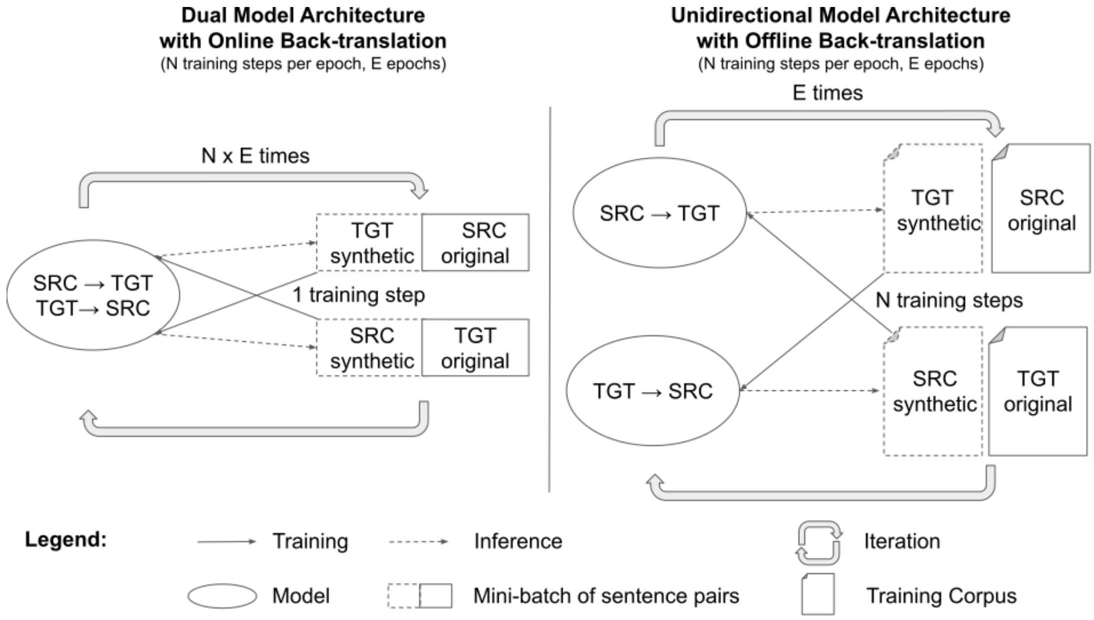


Figure 4.2: Illustration of the dual MT. The bidirectional model (left) is trained jointly in both translation directions using an online back-translation training objective. The two unidirectional models (right) are trained separately for each language pair using the standard supervised MT objective on the back-translated parallel corpus.

improve the initial phrase-based systems by training them on enhanced pseudo-parallel data and argue that less noisy initial translations presented to the NMT model lead to an increase in final translation quality.

Synthetic Data from NMT

Unsupervised systems exploit the dual nature of machine translation where a model trained in one language direction can create training data for a model trained in the reverse direction. Lample et al. [2018a], Conneau and Lample [2019] train a single model for both language directions following the multilingual MT design of Johnson et al. [2017] which allows them to employ back-translation in an online manner where synthetic training data is generated by the very same model that is being trained, one mini-batch at a time. On the other hand, Artetxe et al. [2019a] train two distinct models, one for each translation direction, and they use them to back-translate a large set of 1M sentences. They perform one pass over the synthetic corpus before the next round of back-translation. The two approaches are illustrated in Figure 4.2.

Ren et al. [2019b] use a phrase-based model to filter the noise present in back-translated data from the NMT model by joint incremental training of both the phrase-based and the NMT models in an expectation-maximization framework. Khatri and Bhattacharyya [2020] filter back-translated sentences to give more weight to samples of higher quality, measured by sentence-wise round-trip BLEU score. They report an improvement in translation quality with filtering the synthetic data in the range of 0.5-0.7 BLEU points compared to the baseline trained without filtering. Lu and Zhang [2021] use curriculum learning to reflect different quality of back-translated data. Similarly, Chauhan et al. [2022] weigh back-translated sentences using a round-trip semantic similarity score.

Sun et al. [2021] use synthetic sentences both on the source side and on the target side and confirm that even noisy self-training can improve the MT quality. He et al. [2022] note that the nature of synthetic data creates a style gap between training and inference. The model is trained to translate synthetic sentences biased towards the target domain while it is tested on translating authentic sentences. They try to bridge the gap by mimicking the inference scenario already during training.

4.2.3 Multilingual Data

Garcia et al. [2020] explore the multilingual view on UMT and provide a probabilistic framework that encompasses both supervised and unsupervised training under the framework of expectation-maximization. Sen et al. [2019], Sun et al. [2020] train a multilingual unsupervised NMT model using multilingual denoising and back-translation. Sen et al. [2019] use language-specific decoders, while Sun et al. [2020] report better results when using a shared decoder as well as the encoder. Sun et al. [2020] further improve their results with knowledge distillation.

Garcia et al. [2021] claim that multilinguality is critical for the practical usability of UMT in low-resource conditions. They train a multilingual system with a shared encoder and decoder. They use auxiliary parallel data in three training stages. They pre-train the entire model by masked denoising of monolingual sentences (MASS) [Song et al., 2019], and train for translation with auxiliary parallel data as well as back-translated data. They fine-tune the model using a back-translation term as well as a cross-translation [Garcia et al., 2020] term. They corroborate the robustness of their system in truly low-resource settings.

Wang et al. [2021] confirm the benefits of cross-lingual supervision from a high-resource language pair.

5. Parallel Corpus Mining

Unsupervised machine translation comprises techniques to learn language structure from monolingual data and translate without seeing authentic translation pairs. However, the translation quality is often inadequate for practical purposes and we hypothesize that unsupervised models are not able to exploit all the cross-lingual information hidden in monolingual texts. Therefore, we help them by harvesting some cross-lingual signal ourselves.

Real data collection from human translators leads to creation of data sets of the highest quality, but it is also the slowest and the most expensive option. Arguably, if we want to improve the translation quality of a particular low-resource language or domain, collecting new data from native speakers or domain experts is the best thing that we can do. However, when collecting new natural pieces of text is not an option, we can resort to finding parallel sentences in existing comparable corpora. In this chapter, we explore the possibilities of parallel sentence search and we present a strategy to mine parallel sentences from monolingual corpora. We consider the mined sentence pairs to be *pseudo-parallel* as they should ideally be identical in meaning but in practice only share a certain degree of similarity.

Our approach to parallel corpus mining is the following:

1. embed sentences in a multilingual space;
2. score all possible candidate sentence pairs;
3. set a threshold score for two sentences to be considered parallel;
4. select sentence pairs which score above the threshold.

5.1 Related Work

The state-of-the-art approaches to parallel corpus mining are based on similarity retrieval of sentence embedding vectors using a margin based scoring of translation candidates [Artetxe and Schwenk, 2019b]. Most models rely on heavy supervision by parallel corpora for the embeddings.

Schwenk and Douze [2017], Schwenk [2018], España Bonet et al. [2017] derive sentence embeddings from internal representations of a neural machine translation system with a shared encoder. The top performance in parallel corpus mining is currently achieved by LASER [Artetxe and Schwenk, 2019a], a multilingual BiLSTM model sharing a single encoder for 93 languages trained on parallel corpora to produce language agnostic sentence representations. LASER has been successfully used to mine billions of sentence pairs from the web [Schwenk et al., 2019].

The universal sentence encoder (*USE*) [Cer et al., 2018, Chidambaram et al., 2019, Yang et al., 2020] family covers sentence embedding models with a multi-task dual-encoder training framework including the tasks of question-answer prediction or natural language inference. Guo et al. [2018] directly optimize the cosine similarity between the source and target sentences using a bidirectional dual-encoder. Yang et al. [2020] enhance the model with an *additive margin softmax* loss to separate translations from nearby non-translations.

Since we focus on extracting translation knowledge exclusively from monolingual data, we base our approach in unsupervised multilingual language models such as *M-BERT* [Devlin et al., 2018], XLM [Conneau and Lample, 2019], or *XLM-R* [Conneau et al., 2019]. They were pre-trained with an MLM objective to learn a joint structure of the presented languages without relying on parallel data resources. While several authors [Pires et al., 2019, Wu and Dredze, 2019, Karthikeyan et al., 2019] bring evidence of cross-lingual transfer within such models, their internal representations are not entirely language agnostic [Libovický et al., 2019]. To extend multi-lingual language modelling to low-resource languages, ImaniGooghari et al. [2023] fine-tune *XLM-R* for 500 languages with limited resources (*Glot500*).

An entirely different (and possibly unsupervised) approach is to construct sentence representations by aggregating cross-lingual word embedings either by simple averaging [Arora et al., 2017] or using an IDF weighted average [Litschko et al., 2019]. However, since the mapping is applied to static (non-contextualized) embeddings, this strategy gives up on the contextual information which could be exploited in the sentence representation construction. We use averaged cross-lingual word embeddings obtained in an unsupervised way [Artetxe et al., 2018c] as a baseline for our method.

5.2 Methodology

We propose a method to further align representations from such models into the cross-lingual space and use them to derive sentence embeddings. Our approach is completely unsupervised and is applicable also for distant language pairs. The proposed method outperforms previous

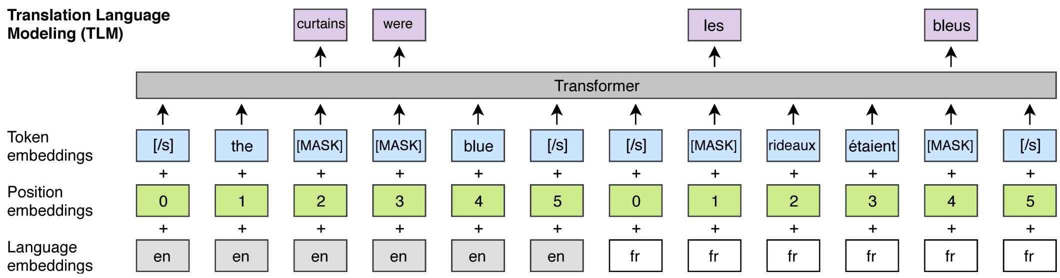


Figure 5.1: Transformer model trained with a translation language modelling (TLM) objective.

Source: Conneau and Lample [2019]

unsupervised approaches on the BUCC 2018[1] shared task, and is even competitive with several supervised baselines. The research work described in this chapter was published [Kvapilíková et al., 2020] and the rest of this chapter includes portions of text and tables verbatim from our research paper.

In the following paragraphs, we describe the multilingual MLM models (Section 5.2.1), the fine-tuning objective for enhanced alignment of their internal representations (Section 5.2.2), and the extraction of sentence embeddings (Section 5.2.4). The experiments in this section were published in Kvapilíková et al. [2020].

5.2.1 Pre-trained Multilingual Masked Language Models

In Section 3.2.4, we introduced the masked language modelling (MLM) training objective used for training Transformer encoder-based language models. Now we show their usability for our purposes.

When training a multilingual MLM, text streams are fed into the model together with a language identification in the form of a language embedding vector which is added to every token embedding. In each training step, the model is presented with one batch of masked text streams for every language. The text streams have usually a fixed size of N tokens and contain several sentences. In our experiments, $N = 256$. The vocabulary of subword units is shared among all languages.

5.2.2 Fine-tuning MLMs with a Translation Objective

When parallel data is available, it can be leveraged in the training of the multilingual language model using a translation language model objective (TLM) [Conneau and Lample, 2019] which is a supervised version of the MLM trained on parallel data. Pairs of sentences are concatenated, random tokens are masked from both sentences and the model is trained to fill in the blanks by attending to any of the words of the two sentences. The training design is illustrated in Figure 5.1. The Transformer self-attention layers thus have the capacity to enrich word

[1] 11th Workshop on Building and Using Comparable Corpora

representations with information about their monolingual context as well as their translation counterparts. This explicit cross-lingual training objective further enhances the alignment of the embeddings in the cross-lingual space.

We use this objective to fine-tune the pre-trained model on a small synthetic parallel data set obtained via unsupervised MT for one language pair, aiming to improve the overall cross-lingual alignment of the internal representations of the model. In our experiments, we also compare the performance to fine-tuning on a small authentic parallel corpus.

Our UMT model follows the approach of Conneau and Lample [2019]. It is a Transformer model with the encoder-decoder architecture. Both the encoder and the decoder are shared across languages and they are initialized with a pre-trained bilingual MLM to bootstrap the training. Both the encoder and the decoder have 6 layers, 8 attention heads, and a hidden unit size of 768. The system is trained using the unsupervised neural MT training pipeline of denoising and back-translation [Lample et al., 2018a] which will be described in detail in Chapter 6.

5.2.3 Fine-tuning MLMs for Unsupported Languages

We work with large-scale pre-trained models which cover a fixed number of languages that appeared in the training data. If we wish to use the model for a language that was not seen during pre-training, we have to fine-tune the model ex-post. If the script of our target language is included in the vocabulary of the pre-trained model, we can proceed directly with fine-tuning for the MLM task. However, it is important to note that the subword segmentation may not be ideal and could potentially result in character-level splitting for less common scripts. If the characters are unknown to the model or the performance is unsatisfactory, the vocabulary can be extended [Wang et al., 2019].

5.2.4 Sentence Embeddings

It was explained in Chapter 3 that Transformer language models produce contextual representations capturing the semantic and syntactic properties of word (subword) tokens in their variable context. Contextualized embeddings can be derived from any of the internal layer outputs of the model. We experiment with representations from different layers and evaluate them on the task of parallel sentence matching to select the one that best suits our objective.

Parallel sentence search requires the use of sentence embeddings rather than subword token embeddings. Aggregating token embeddings to fixed-length sentence representations necessarily leads to an information loss. We compose sentence embeddings from subword representations by simple element-wise averaging. Even though mean-pooling is a naive approach to subword aggregation, it is often used for its simplicity [Reimers and Gurevych, 2019, Ruiter et al., 2019, Ma et al., 2019] and in our scenario it yields better results than max-pooling.

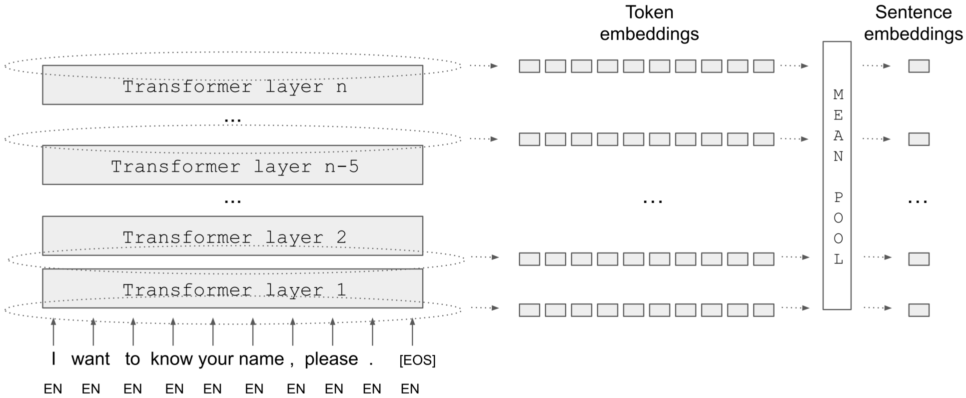


Figure 5.2: Encoding a masked sentence by a Transformer model. Contextualized word embeddings are aggregated by mean-pooling.

5.2.5 Searching in Multilingual Embedding Space

In our approach to parallel sentence mining, the first step is to embed all sentences in a shared multilingual space where they can be scored and matched to find pairs which are equivalent or at least similar in meaning.

In order to score all possible candidate sentence pairs, we use the margin-based approach of Artetxe and Schwenk [2019b] which was proved to eliminate the hubness problem of embedding spaces and yield superior results [Artetxe and Schwenk, 2019a]. The score relies on cosine similarity to measure the distance between sentences but it is defined in relative terms to the average cosine similarity between the two sentences and their nearest neighbors.

$$\mathrm{xsim}(x, y) = \mathrm{margin}(\cos(x, y), \sum_{z \in \mathrm{NN}_k(x)} \frac{\cos(x, z)}{2k} + \sum_{z \in \mathrm{NN}_k(y)} \frac{\cos(y, z)}{2k}) \tag{5.1}$$

where $\mathrm{margin}(a, b) = \frac{a}{b}$, $\mathrm{NN}_k(x)$ is the set of k nearest neighbors of x. The method for scoring involves cosine similarity which is comparatively evaluated against the average cosine similarity of a given sentence with its nearest neighbors to eliminate the "hubs". When the score surpasses a designated threshold T, two sentences are deemed to be parallel:

$$\mathrm{xsim}(x, y) > T \tag{5.2}$$

The optimal threshold for filtering the translation pairs is learned by tuning on the train set F1 scores.

5.3 Experiments

We empirically evaluate the quality of our cross-lingual sentence embeddings and compare it with state-of-the-art supervised methods and unsupervised baselines. We evaluate the proposed method on the task of parallel corpus mining and parallel sentence matching. We fine-tune two different models using English-German (EN-DE) and Czech-German (CS-DE) synthetic parallel data. For comparison, we fine-tune two alternative models using authentic parallel data in the following two low-resource language pairs: English-Nepali (EN-NE) and English-Kazakh (EN-KK).

5.3.1 Model

In this work, we use the publicly available pre-trained model *XLM-100*[2] [Conneau and Lample, 2019] with 16 transformer layers, 16 attention heads, and the hidden unit size of 1280. The model was trained on monolingual corpora in 100 languages mainly from Wikipedia with the BPE vocabulary of 200k subwords. We also experimented with the *bert-base-multilingual-cased* model with similar or slightly worse results. While *XLM-R* [Conneau et al., 2019] was reported to deliver better results on several tasks, we do not observe a significant difference for parallel sentence mining and we use the more lightweight *XLM-100* which has a higher dimension of internal representations than the *large* configuration of *XLM-R* but a lower overall number of parameters. For the sake of brevity, we will refer to the *XLM-100* model as *XLM* throughout the remainder of this chapter.

5.3.2 Data

The *XLM* model was pre-trained on the Wikipedia corpus of 100 languages [Conneau and Lample, 2019]. The monolingual data for fine-tuning was sampled from NewsCrawl 2018 (10k CS sentences, 10k DE sentences, 10k EN sentences).

Monolingual training data for the English-German UMT models was obtained from NewsCrawl 2007-2008 (5M sentences per language). The text was cleaned and tokenized using standard Moses [Koehn et al., 2007] tools and segmented into BPE units based on 60k BPE splits.

5.3.3 Training

To generate synthetic data for fine-tuning the sentence encoder, we train two UMT models (EN-DE, CS-DE) using the same method and parameters as in Conneau and Lample [2019] on 8 GPUs for 24 hours. We use these models to translate 10k sentences in each language. The translations are coupled with the originals into two parallel corpora of 20k synthetic sentence pairs.

The small synthetic parallel corpora obtained in the first step are used to fine-tune the pre-trained *XLM* model using the TLM objective. We measure the quality of induced cross-lingual embeddings from different layers on the task of parallel sentence matching described in Section 5.4.2 to choose the layer and to determine the optimal training time. We conclude that the best cross-lingual performance is achieved at the 12th (5th-to-last) layer and we observe the best results after fine-tuning for one epoch with a batch size of 8 sentences and all other pre-training parameters intact. The development accuracy decreases with fine-tuning on a larger data set. The evaluation across layers is summarized in Figure 5.3.

[2] `https://github.com/facebookresearch/XLM`

5.3.4 Benchmarks

We assess our method against two unsupervised baselines to separately measure the fine-tuning effect on the *XLM* model and to compare our results to another possible unsupervised approach based on post-hoc alignment of word embeddings.

Vanilla XLM: Contextualized token representations are extracted from the 12th layer of the original *XLM*[3] model and mean-pooled into sentence embeddings.

Word Mapping: We use Word2Vec embeddings with 300 dimensions pre-trained on NewsCrawl and map them into the cross-lingual space using the unsupervised version of VecMap [Artetxe et al., 2018c]. As above, word embeddings are aggregated by mean-pooling to represent sentences.[4]

5.4 Results

We explore the multilinguality of a large pre-trained language model *XLM*[5] by assessing its representations on a task of corpus deshuffling. Since the model is trained in a completely unsupervised way, any evidence of cross-lingual transfer is surprising. We dissect the model to assess how much cross-lingual information is hidden in its internal representations on different layers and select which layer outputs the most multilingual representations. We use the findings from this experiment when setting hyperparameters in further experiments.

5.4.1 Evaluation I: Parallel Corpus Mining

We measure the performance of our method on the BUCC shared task of parallel corpus mining where candidate systems are expected to search two comparable non-aligned corpora and identify pairs of parallel sentences. We evaluate on two data sets – the original BUCC 2018 corpus created by inserting parallel sentences into monolingual texts extracted from Wikipedia [Zweigenbaum et al., 2017] and a new BUCC-like data set (News train and test) which we created by shuffling 10k parallel sentence from News Commentary into 400k monolingual sentences from News Crawl. The BUCC and News data sets are comparable in size and contain parallel sentences from the same source, but differ in overall domain.

Tables 5.1 and 5.2 show the results of our proposed model on the BUCC and News test sets. When comparing our method to related work, it must be noted that the underlying *XLM* model was pre-trained on Wikipedia and therefore has seen the monolingual BUCC sentences during training. This could result in an advantage over other systems, as the model could exploit the fact that it has seen the non-parallel part of the comparable corpus during training. However, since both the proposed method and the *vanilla XLM* baseline suffer from this, their results remain comparable. We also report results on the News test set which is free from such potential bias (Table 5.2).

[3] Using *M-BERT* model yielded similar results to XLM.

[4] Weighting word embeddings by their sentence frequency (IDF) did not lead to a significant improvement over a simple average.

[5] `https://github.com/facebookresearch/XLM`.

	EN-DE	EN-FR	EN-RU	EN-ZH	Supervision
Leong et al. [2018]	-	-	-	56.00	bitext (0.5M sent.)
Bouamor and Sajjad [2018]	-	76.00	-	-	bitext (2M sent.)
Schwenk [2018]	76.90	75.80	73.80	71.60	multi (2M sent.)
Azpeitia et al. [2018]	85.52	81.47	81.30	77.45	bitext (2-9M sent.)
Artetxe&Schwenk [2019]	**96.19**	**93.91**	**93.30**	**92.27**	multi (223M sent.)
Word Mapping	32.04	32.94	17.68	20.65	none
Vanilla XLM*	62.10	64.77	61.65	44.79	none
Our method* (EN↔DE)	**80.06**	**78.77**	**77.16**	**67.04**	none (20k sent.**)

Table 5.1: F1 score on the parallel sentence mining task (BUCC test set). The supervised (upper part) and unsupervised (lower part) winners are highlighted in bold. * The model was pre-trained on Wikipedia. ** Synthetic translations produced by unsupervised MT.

Source: Kvapilíková et al. [2020]

	EN-DE	EN-FR	EN-RU	EN-ZH	EN-KK	CS-ZH	DE-RU
Artetxe&Schwenk [2019]	**90.30**	**87.38**	**94.34**	**83.92**	12.07	**73.41**	**88.39**
Word Mapping	28.45	30.79	17.81	16.04	2.28	10.86	19.55
Vanilla XLM	72.58	71.92	72.90	59.26	24.00	43.00	58.29
OUR METHOD (EN↔DE)	**79.32**	**77.05**	**80.98**	**65.49**	**35.41**	**48.79**	**65.91**

Table 5.2: F1 score on the parallel sentence mining task (News test set). The supervised and unsupervised winners are highlighted in bold. Artetxe and Schwenk [2019a] values were obtained using the public implementation of the LASER toolkit.

Source: Kvapilíková et al. [2020]

The results reveal that TLM fine-tuning on the synthetic parallel sentences brings a substantial improvement over the initial pre-trained model trained only using the MLM objective (*vanilla XLM*). In terms of the F1 score, the gain across four BUCC language pairs major and ranges between 14.0-22.3 points. Even though the fine-tuning focused on a single language pair (English-German), the improvement is notable for all evaluated language pairs. The largest margin of 21.6 points is observed for the English-Chinese mining task. We observe that using a small parallel data set of authentic translation pairs instead of synthetic ones does not have a significant effect.

The weak results of the *word mapping* baseline can be partially attributed to the superiority of contextualized embeddings for representation of sentences over static ones. Furthermore, word mapping relies on the questionable assumption of isomorphic embedding spaces which weakens its performance especially for distant languages. In our proposed model, it is possible that joint training of contextualized representations induces an embedding space with more convenient geometric properties which makes it more robust to language diversity.

Although the performance of our model generally lags far behind the supervised *LASER* benchmark, it is valuable because of its fully unsupervised nature and it works even for distant languages such as Chinese-Czech or English-Kazakh.

	DE-EN	CS-EN	CS-DE	CS-FR	CS-RU	FR-ES	FR-RU	ES-RU
Artetxe&Schwenk [2019]	98.78	99.08	99.23	99.37	98.77	99.42	98.60	98.77
Word Mapping	60.60	55.03	75.35	43.33	79.87	71.07	41.25	53.87
Vanilla XLM	87.15	79.83	82.87	80.55	85.15	91.07	85.28	85.73
Our method (EN↔DE)	93.97	**90.47**	90.48	**90.07**	92.23	**94.68**	**91.80**	**91.92**
Our method (CS↔DE)	**94.43**	90.15	**90.50**	89.48	**92.33**	94.65	91.72	91.25

Table 5.3: Accuracy on the deshuffling task (*newstest2012*) averaged over both matching directions. Artetxe and Schwenk [2019a] values were obtained using the public implementation of the LASER toolkit.

Source: Kvapilíková et al. [2020]

5.4.2 Evaluation II: Corpus Deshuffling

To assess the effect of the proposed fine-tuning on other language pairs not covered by BUCC, we evaluate our embeddings on the task of corpus deshuffling. The task entails searching a pool of shuffled parallel sentences to recover correct translation pairs. Cosine similarity is used for the nearest neighbor search.

We first evaluate the pairwise matching accuracy on the *newstest* multi-way parallel data set of 3k sentences in 6 languages.[6] We use *newstest2012* for development and *newstest2013* for testing. The results in Table 5.3 show that the fine-tuned model is able to match correct translations in 90-95% of cases, depending on the language pair, which is $\sim$7% more than *vanilla XLM*. It is notable that the model which was only fine-tuned on English-German synthetic parallel data has a positive effect on completely unrelated language pairs as well (e.g. Russian-Spanish, Czech-French).

Since the greatest appeal of parallel corpus mining is to enhance the resources for low-resource languages, we also measure the deshuffling accuracy on the Tatoeba [Artetxe and Schwenk, 2019a] data set of 0.5–1k sentences in over 100 languages aligned with English. Aside from the two completely unsupervised models, we fine-tune two more models on small authentic parallel data in English-Nepali (5k sentence pairs from the Flores development sets) and English-Kazakh (10k sentence pairs from News Commentary). Table 5.4 confirms that the improvement over *vanilla XLM* is present for every language we evaluated, regardless of the language pair used for fine-tuning. We initially hypothesized that the performance of the English-German model on English-aligned language pairs would exceed the German-Czech model, but their results are equal on average. Fine-tuning on small authentic corpora in low-resource languages exceeds both by a slight margin.

The results are clearly sensitive to the amount of monolingual sentences in the Wikipedia corpus used for XLM pre-training and the matching accuracy of very low-resource languages is significantly lower than we observed for high-resource languages. However, the benefits of fine-tuning are substantial (around 20 percentage points) and for some languages, the results even reach the supervised baseline (e.g. Kazakh, Georgian, Nepali).

It seems that explicitly aligning one language pair during fine-tuning propagates through the shared parameters and improves the overall representation alignment, making the contextu-

[6] Czech, English, French, German, Russian, Spanish

	AF	AR	AZ	BE	BG	CA	CS	DE	EL	EO
Sup. baseline	89.5	92.0	66.0	66.2	95.0	95.9	96.5	99.0	95.0	97.2
Vanilla XLM	38.1	19.9	25.1	33.7	36.2	51.0	31.5	65.0	27.0	45.8
EN↔DE (synth)	57.3	41.1	46.3	58.4	56.0	66.9	53.5	83.1	51.3	68.0
CS↔DE (synth)	54.2	41.2	44.2	61.8	60.7	68.9	59.9	87.3	53.1	67.4
EN↔KK (auth)	58.4	45.6	51.4	60.2	59.2	72.6	53.9	87.0	54.6	72.1
EN↔NE (auth)	59.9	46.6	54.2	63.1	62.9	71.0	57.6	85.0	51.0	71.2

	ET	FI	FY	HI	HR	IA	IS	ID	JA	KA
Sup. baseline	96.7	96.3	51.7	94.7	97.2	95.2	95.6	94.5	91.8	35.9
Vanilla XLM	19.8	31.4	37.0	26.2	47.2	57.3	25.0	46.4	29.5	22.1
EN↔DE (synth)	39.0	47.5	48.6	53.4	68.2	71.4	43.1	64.9	54.4	41.4
CS↔DE (synth)	41.4	49.5	44.8	51.7	71.8	70.5	43.7	64.1	53.3	39.8
EN↔KK (auth)	43.4	51.3	51.7	60.3	71.3	79.5	45.0	66.4	59.6	44.0
EN↔NE (auth)	44.6	52.7	48.6	59.3	72.1	75.7	47.1	67.8	59.6	47.8

	KK	KU	LT	MK	ML	MN	MR	MS	NE	NN
Sup. baseline	18.6	17.2	96.2	94.7	96.9	8.2	91.5	96.4	20.6	88.3
Vanilla XLM	17.4	10.6	22.0	25.8	17.4	12.6	15.3	52.0	21.3	49.9
EN↔DE (synth)	33.6	16.8	43.9	48.8	51.6	29.0	37.3	67.0	32.8	66.8
CS↔DE (synth)	34.7	16.2	46.2	51.1	44.3	24.5	34.2	65.4	31.4	67.5
EN↔KK (auth)	46.1	20.0	46.2	54.7	54.0	32.7	41.9	69.8	37.3	69.2
EN↔NE (auth)	38.4	20.9	47.7	53.8	56.0	34.9	43.5	72.1	42.8	69.2

	OC	SL	SR	SV	TA	TE	TL	UK	UR	YI
Sup. baseline	61.2	95.9	95.3	96.6	69.4	79.7	50.5	94.5	81.9	5.7
Vanilla XLM	20.0	34.7	35.9	47.2	11.9	14.1	14.6	38.0	19.3	9.9
EN↔DE (synth)	34.3	54.9	58.6	69.7	40.9	44.7	24.0	66.1	43.7	22.1
CS↔DE (synth)	35.9	59.2	64.8	71.8	31.9	37.8	20.4	70.4	43.8	22.8
EN↔KK (auth)	40.3	58.0	64.3	73.3	42.8	44.0	24.4	71.6	48.2	25.8
EN↔NE (auth)	36.9	58.8	65.0	72.0	41.7	53.2	26.8	71.0	49.9	26.7

Table 5.4: Accuracy on the deshuffling task (*Tatoeba*) averaged over both matching directions (to and from English). The supervised baseline was obtained using the public implementation of the *LASER* model [Artetxe and Schwenk, 2019a]. Our proposed models were fine-tuned on synthetic parallel data (EN↔DE, CS↔DE) and authentic parallel data (EN↔KK, EN↔NE).

Source: Kvapilíková et al. [2020]

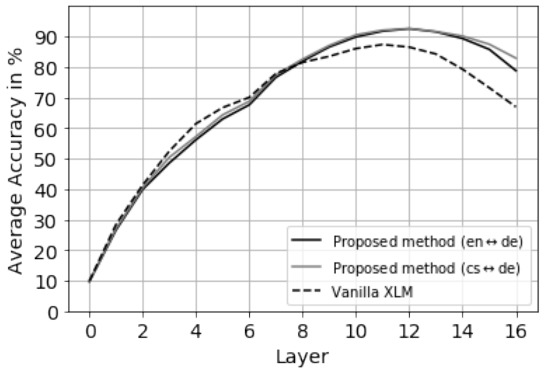


Figure 5.3: Average deshuffling accuracy on *newstest2012* before and after fine-tuning from the input embedding layer (0th) to the deepest layer (16th).

Source: Kvapilíková et al. [2020]

alized embeddings more language agnostic. The propagation effect could also positively influence the ability of cross-lingual transfer within the model in downstream tasks. A verification of this is left to future work.

5.4.3 Analysis: Representations Across Layers

We derive sentence embeddings from each of the layers of the model and show deshuffling results on the development set averaged over all language pairs in Figure 5.3, both before and after fine-tuning. The accuracy differs substantially across the model depth, the best cross-lingual performance is consistently achieved around the 12th (5th-to-last) layer of the model. The TLM fine-tuning affects especially the deepest layers.

5.4.4 Parallel Corpus Mining for Unsupported Languages

The *XLM* model only supports the 100 languages covered during pre-training. In order to use its representations for other languages, the model first has to be fine-tuned.

English-Inuktitut

In the following experiments, we create sentence representations for text in Inuktitut, a language that was not included in the pre-training of the *XLM*, and use them for English-Inuktitut parallel corpus mining.

We create an English-Inuktitut (EN-IKU) encoder by fine-tuning our proposed model (EN↔DE) with the MLM objective on 1M monolingual sentences from the Hansard[7] corpus (IKU) and NewsCrawl (EN). Since the two languages are linguistically distant and Inuktitut has a non-Latin script, this is a particularly difficult scenario.

We experiment with fine-tuning the entire model versus weight-freezing and fine-tuning only the lexical embeddings. Furthermore, we experiment with random initialization of lexical

[7] `https://www.inuktitutcomputing.ca/NunavutHansard/info.php?lang=en`

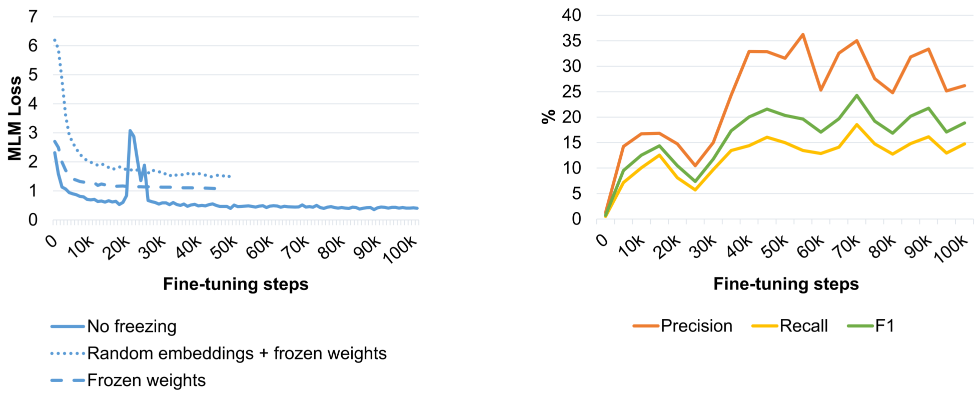


Figure 5.4: Training curves from fine-tuning the proposed model (en↔de) with the MLM objective on English and Inuktitut texts with and without parameter freezing *(left)*. Precision, recall and F1 scores of the model fine-tuned without weight freezing on the task of parallel corpus mining for English and Inuktitut *(right)*.

embeddings prior to the fine-tuning. Otherwise, the training details are identical to the TLM fine-tuning described in Section 5.3.3. The training curves are shown in Figure 5.4. Although updating the entire model experiences a sudden drop in performance at the beginning of the training, it recovers and eventually converges to the highest MLM accuracy out of the three approaches. Therefore, in our future experiments, we do not freeze weights and always update the entire model during fine-tuning.

Decreasing MLM loss does not yet guarantee that the model is creating bilingual representations usable for parallel sentence search. We measure the mining performance of the model by trying to recover 5k parallel sentences[8] mixed into 100k monolingual sentences. The precision, recall, and F1 scores are evaluated as the fine-tuning progresses and plotted in Figure 5.4. We observe an initial performance boost as the model adapts to the new language, followed by fluctuating outcomes, with precision ranging from 25% to 35%. The fact that the model was able to correctly recover up to 18% of the hidden sentences means that it was able to at least partially align its representations of Inuktitut to English.

Indic Languages

For our later MT experiments with Assamese (AS), Khasi (KHA), Manipuri (MNI) and Mizo (MZ) which were also not a part of the original model, we create a new version of the XLM encoder by fine-tuning on monolingual data using the MLM objective without weight freezing. Although AS and MNI use a non-Latin script, the vocabulary of the original model contains all characters from the Bengali-Assamese alphabet so we do not have to extend it. We start from the *XLM-100* model and fine-tune on the MLM task in the four Indic languages and English. We use the batch size of 40 sentences per GPU and train on 2 GPUs. We use Adam optimization with a leaning rate λ=0.00005.

[8]Parallel sentences are taken from the Hansard dev set.

	EN-AS	EN-KHA	EN-MNI	EN-MZ
Glot500 (8th layer)	15.05	-	-	4.02
XLM-R base (8th layer)	2.23	-	-	-
XLM-R large (12th layer)	3.45	-	-	-
XLM-100 (12th layer)	-	-	-	-
$\mapsto$ fine-tuned (Indic)	24.26	10.07	6.63	20.01
$\mapsto$fine-tuned (EN$\leftrightarrow$DE synth)	**47.16**	**25.88**	**12.76**	**36.02**

Table 5.5: F1 scores on the task of parallel corpus mining where the systems try to recover a set of 2k sentences shuffled into monolingual corpora of 200k sentences from the train set. A dash (-) signifies that one of the languages was not covered by the sentence encoder. *Glot500* and *XLM-R* base have 12 layers; *XLM-100* and *XLM-R* large have 16 layers.

In Table 5.5, we report the performance of the fine-tuned model on the task of parallel corpus mining where the model is evaluated on finding parallel sentences in two corpora of 202k sentences built by mixing the development set of 2k parallel sentences into a random set of 200k monolnigual sentences from the training corpus[9]. Since the F1 scores are notably lower than we saw in Section 5.4.1, we attempt to align the representations further. We employ the technique from Section 5.2 where we fine-tune the entire model on a small synthetic English-German corpus. We use the identical corpus now and observe that after the light fine-tuning, the internal representations of the model are more suitable for parallel corpus mining. The positive effect starts diminishing after the model had been exposed to 60k synthetic translations. The results are reported in the last row of Table 5.5.

We compare our fine-tuned sentence encoder to two more recent unsupervised multilingual language models: *XLM-R* (supports AS) and *Glot500* (supports AS and MZ). The models were pre-trained using the identical MLM pre-training objective as the *XLM-100* model but they were exposed to significantly more data. We follow [Jalili Sabet et al., 2020] and take representations from the 8th layer of the base-sized models. For the large-sized models, we follow our earlier experiments and use the 12th layer. The performance of the benchmarks is very low, even for the *Glot500* model which specializes in low-resource languages. We note that the benchmarks have a lower dimensionality in their internal representations (768 for *XLM-R base* and *Glot500*, 1024 for *XLM-R large*, 1280 for *XLM-100*).

5.5 Takeaways

We proposed a completely unsupervised method for training of multilingual sentence embeddings which can be used for building a parallel corpus with no previous translation knowledge.

We showed that by fine-tuning a pre-trained multilingual encoder with the TLM objective of gap-filling in bilingual sentence pairs, we can significantly enhance the cross-lingual alignment of its representations using as little as 20k synthetic translation pairs. Since the synthetic translations were obtained from an unsupervised MT system, the entire procedure requires no authentic parallel sentences for training.

[9]The source of the data is described in Section 7.5

Our sentence embeddings yield significantly better results on the tasks of parallel corpus mining and parallel sentence matching than our unsupervised baselines. Interestingly, targeting only one language pair during the fine-tuning phase suffices to propagate the alignment improvement to unrelated languages. It is therefore not necessary to build a working MT system for every language pair we wish to mine.

The average F1 margin across four language pairs on the BUCC task is $\sim$17 points over the original *XLM* model and $\sim$7 on the News dataset where only one of the evaluated language pairs was seen during fine-tuning. The gain in accuracy in parallel sentence matching across 8 language pairs is 7.2% absolute, lagging only 7.1% absolute behind supervised methods.

It is possible to adapt the proposed approach to new languages outside of the original model coverage by MLM fine-tuning. The performance can be further improved by light fine-tuning of the adapted model using synthetic parallel sentences. The source of this improvement deserves further investigation.

In Chapter 7, we will be using the proposed model to mine parallel sentences and create pseudo-parallel corpora for the training of unsupervised MT systems.

6. Unsupervised Machine Translation Methodology

This chapter outlines the methodology of training UMT systems that we employ in our experiments. We start by describing techniques for extracting cross-lingual signal from monolingual data at the word level, which can serve for initialization of both phrase-based and neural models. Specifically, we detail unsupervised methods to create a cross-lingual embedding space and build a bilingual lexicon. We then explain the functioning of unsupervised phrase-based systems (UPBMT), and finally, we delve into the neural models (UNMT).

Unsupervised models extract translation signal from monolingual texts in several different ways. The core concept remains the same – the semantic structures of text in different languages share similarities in how words interrelate and unsupervised models leverage this commonality. They utilize their constrained internal structures to generate bilingual or even multilingual representations. .

6.1 Unsupervised Cross-Lingual Embeddings

We first discussed the topic of cross-lingual embeddings in Chapter 3 together with the limitations posed by the restrictive assumption of isomorphism of embedding spaces. We formally defined the problem of finding a linear mapping matrix W between the source and the target embedding space in Equation 3.1. We showed that the problem has a closed-form solution (Equation 3.2) provided that a seed bilingual lexicon is available.

6.1.1 Seed Lexicon

A number of approaches has been proposed to create the seed lexicon without the need of parallel texts.

1. If the source and the target languages both use Arabic numerals, they can serve as the initial seed lexicon [Artetxe et al., 2017].

2. If the source and the target languages share identical words (e.g. named entities), they can serve as the initial seed lexicon [Artetxe et al., 2017].

3. The initial seed lexicon can be derived in a fully unsupervised way by exploiting structural similarities between embedding spaces [Artetxe et al., 2018c]. For a source embedding matrix X and a target embedding matrix Y where individual rows correspond to word embeddings x_i and y_i, the similarity matrices $M_X = XX^T$ and $M_Y = YY^T$ should match. In practice, if embedding spaces are at least approximately isomorphic, the initial seed lexicon can be derived by a nearest neighbor search over the rows of the similarity matrices.

4. The initial seed lexicon can be derived from a mapping learned by adversarial training [Conneau et al., 2018a]. An initial proxy for the mapping matrix W between source embeddings x_i and target embeddings y_i is obtained in an adversarial training framework proposed by Ganin et al. [2017]. A discriminator is trained to discriminate between elements randomly sampled from $\{Wx_1, ..., Wx_n\}$ and $\{y_1, ..., y_m\}$ while W is trained to prevent the discriminator from making accurate predictions.

The approaches (1)–(3) are implemented in the VecMap[1] library and the approach (4) is implemented in the MUSE[2] library. In this thesis, we experiment with different approaches and rely on default hyperparameters from the implementations.

6.1.2 Self-Refinement

Initial solutions outlined above can always be improved by a self-learning refinement [Artetxe et al., 2017] where the maping matrix W is iteratively updated using the word pairs from the currently best lexicon as anchor points for the Procrustes problem which has a closed-form solution (Equation 3.2). A new updated lexicon is built in each round by the nearest neighbor retrieval relying on the CSLS similarity metric [Conneau et al., 2018a]

$$\mathrm{CSLS}(x, y) = \cos(x, y) - \sum_{z \in \mathrm{NN}_k(x)} \frac{\cos(x, z)}{2k} - \sum_{z \in \mathrm{NN}_k(y)} \frac{\cos(y, z)}{2k} \tag{6.1}$$

where $\mathrm{NN}_k(x)$ is the set of k nearest neighbors of x that are used to reduce the cosine similarity for embeddings that manifest the hubness problem, characterized by an excessive number of close neighbors.

In summary, the unsupervised learning algorithm for post-hoc alignment of monolingual embeddings into a cross-lingual space is the following:

1. Build the initial bilingual lexicon L using one of the approaches in Section 6.1.1.

2. Given the lexicon L, calculate W as the closed-form solution of the Procrustes problem (Equation 3.2).

3. Obtain an improved lexicon L by a nearest neighbor search among target embeddings y_i and aligned source embeddings Wx_i.

4. Repeat (2) and (3) for a fixed set of iterations or until a convergence criterion is met.

6.1.3 Applications in Unsupervised MT

Pre-trained cross-lingual embedding spaces have been successfully used as the initial source of cross-lingual signal into unsupervised MT systems. We use them in our experiments with both phrase-based and neural models. The methods described in this section can be extended

[1] `https://github.com/artetxem/vecmap`
[2] `https://github.com/facebookresearch/MUSE`

to phrases and used to populate a phrase table of an unsupervised phrase-based system (Section 6.2). Alternatively, when the method is applied on the subword level, the aligned cross-lingual subword embeddings can serve for initialization of the embedding layer of an unsupervised neural model (Section 6.3).

6.2 Unsupervised Phrase-Based Machine Translation

PBMT models were introduced in Section 3.3.2 as log-linear models which operate with phrases (n-grams) and have several components: phrase table, language model, reordering model, and fixed word/phrase penalties. While monolingual texts suffice for the calculation of the language model probabilities and the fixed penalties, the phrase table and the reordering model require parallel data. The reordering model can be omitted in the initial version of the system, but the phrase table is the essential component of the system that facilitates translation. Populating the phrase table with translation candidate phrases and their probabilities in an unsupervised way is the crucial part of UPBMT.

The underlying assumption behind UPBMT is the existence of shared cross-lingual embedding space where words and phrases are represented in a language-neutral way. If we create such an embedding space, phrase translation candidates can be found by a nearest neighbor search and their translation probabilities can be derived from the cosine distance of their vector representations.

UPBMT systems are created in several steps [Artetxe et al., 2018b]:

- input text tokenization and truecasing;
- training of phrase embeddings (Section 6.2.1);
- mapping of phrase embeddings into the cross-lingual space (Section 6.2.1);
- populating the initial phrase table (Section 6.2.2);
- estimation of an n-gram language model (Section 6.2.3);
- weight tuning of the log-linear model (Section 6.2.4);
- back-translation refinement (Section 6.2.5).

The training algorithm is displayed in Figure 6.1.

6.2.1 Cross-Lingual Phrase Embeddings

Phrase embeddings are learned by a generalization of the Skip-gram model that learns embeddings for longer n-grams in addition to the individual word embeddings as implemented in the `phrase2vec`[3] library. We train phrase embeddings for the source and the target language individually. In order to transform the two monolingual embedding spaces in one cross-lingual embedding space, we use the alignment technique described in Section 6.1 which relies on shared Arabic numerals for the initial solution and five iterations of self-refinement.

[3] `https://github.com/artetxem/phrase2vec`

6.2.2 Initial Phrase Table Induction

The next step is to populate the phrase table with translation candidate pairs. For each source phrase, we search the embedding space to extract N nearest neighboring phrases in the target language and vice versa. The translation probability of each candidate pair is calculated as follows

$$p(tgt|src) = \frac{e^{\cos(src,tgt)/\tau}}{\sum_{tgt'} e^{\cos(src,tgt')/\tau}} \tag{6.2}$$

where *src* is the original source phrase, *tgt* is the selected translation and tgt' iterates over the N possible translations. τ is a constant temperature parameter controlling the confidence of the predictions. In our experiments, we follow Lample et al. [2018b] and set $N = 100$ and $\tau = 30$.

6.2.3 Language Model

The role of a language model in PBMT is to assign higher probability values to more likely word sequences (n-grams). Since frequency counts are derived from monolingual corpora, the estimation of n-gram probabilities is not influenced by the absence of parallel data. Back-off and smoothing techniques [Manning and Schütze, 1999] are applied to adjust the probability estimates for unseen n-grams or n-grams with very low counts. In particular, we use modified Kneser-Ney smoothing [Heafield et al., 2013] implemented in the KenLM toolkit.

6.2.4 Unsupervised Tuning

In supervised PBMT, the MERT algorithm is used to tune the weights of individual components of the log-linear model on a small parallel data set. Since it is not available in the unsupervised setting, we first use the src$\rightarrow$tgt PBMT model with its default weights to translate a small portion of the monolingual corpus and use the synthetic parallel data set for MERT tuning of the opposite tgt$\rightarrow$src model. The procedure is iteratively repeated in both translation directions until convergence, as indicated in steps 7–11 of the training algorithm (Figure 6.1).

6.2.5 Back-Translation

Finally, we run several rounds of back-translation whereby we translate the monolingual corpus by the src$\rightarrow$tgt model and use the synthetic corpus for PBMT training of the opposite tgt$\rightarrow$src model in a standard supervised way. Full supervised training consists of estimating the phrase table and the reordering model from the synthetic training corpus and MERT tuning on the synthetic development set for finding the optimal weights. We repeat the process in the opposite translation direction and refine the solution in several iterations of back-translation, as indicated in steps 12–16 of the training algorithm (Figure 6.1). If the monolingual training corpora are large, the back-translation procedure can be run on a smaller subset for higher efficiency. The original paper [Artetxe et al., 2018b] suggests using 2M sentences.

Input: Monolingual training corpora: $train_{src}$ and $train_{tgt}$
Monolingual development corpora: dev_{src} and dev_{tgt}

Output: Trained models: $model_{src \to tgt}$ and $model_{tgt \to src}$
Synthetic parallel training corpora:
$(train_{src\_synth}, train_{tgt})$ and $(train_{tgt\_synth}, train_{src})$
Synthetic parallel development corpora:
$(dev_{src\_synth}, dev_{tgt})$ and $(dev_{tgt\_synth}, dev_{src})$

1. $pt_{src \to tgt} \leftarrow \text{INDUCE\_PHRASE\_TABLE}(train_{src}, train_{tgt})$
2. $pt_{tgt \to src} \leftarrow \text{INDUCE\_PHRASE\_TABLE}(train_{tgt}, train_{src})$
3. $lm_{src} \leftarrow \text{TRAIN\_LM}(train_{src})$
4. $lm_{tgt} \leftarrow \text{TRAIN\_LM}(train_{tgt})$
5. $model_{src \to tgt} \leftarrow \text{BUILD\_MODEL}(lm_{tgt}, pt_{src \to tgt})$
6. $model_{tgt \to src} \leftarrow \text{BUILD\_MODEL}(lm_{src}, pt_{tgt \to src})$
7. Repeat until convergence:
8. $\quad dev_{src\_synth} \leftarrow \text{TRANSLATE}(model_{tgt \to src}, dev_{tgt})$
9. $\quad model_{src \to tgt} \leftarrow \text{TUNE\_WEIGHTS}(model_{src \to tgt}, dev_{src\_synth}, dev_{tgt})$
10. $\quad dev_{tgt\_synth} \leftarrow \text{TRANSLATE}(model_{src \to tgt}, dev_{src})$
11. $\quad model_{tgt \to src} \leftarrow \text{TUNE\_WEIGHTS}(model_{tgt \to src}, dev_{tgt\_synth}, dev_{src})$
12. Repeat until convergence:
13. $\quad train_{src\_synth} \leftarrow \text{TRANSLATE}(model_{tgt \to src}, train_{tgt})$
14. $\quad model_{src \to tgt} \leftarrow \text{MOSES\_TRAIN}(train_{src\_synth}, train_{tgt})$
15. $\quad train_{tgt\_synth} \leftarrow \text{TRANSLATE}(model_{src \to tgt}, train_{src})$,
16. $\quad model_{tgt \to src} \leftarrow \text{MOSES\_TRAIN}(train_{tgt\_synth}, train_{src})$

Figure 6.1: Unsupervised PBMT training algorithm. INDUCE_PHRASE_TABLE creates an initial phrase table from monolingual embeddings as described in Section 6.2.2. TRAIN_LM trains an n-gram language model. BUILD_MODEL uses default weights and pre-computed penalties to build a translation model from the initial phrase table and the target language model. TUNE_WEIGHTS applies the MERT algorithm over the synthetic development set to find optimal weights of the log-linear model. MOSES_TRAIN applies the full supervised PBMT training algorithm (as described in Section 3.3.2) on a synthetic parallel corpus.

6.3 Unsupervised Neural Machine Translation

In this section, we describe the methodology of unsupervised neural MT (UNMT) adopted in our experiments. As we move from the phrase-based translation to neural models, we observe that the principles of UMT underlying the two types of models are similar.

- The initial solution is obtained by pre-trained cross-lingual representations (mapped static embeddings or deeper representations learned during multilingual pre-training).
- Translation is learned together with a monolingual language modelling objective (n-gram LM in UPBMT, denoising autoencoding in UNMT).
- The initial solution is refined using back-translation.

6.3.1 Vocabulary

When training UNMT models, we work with monolingual corpora D_{src} and D_{tgt}. Optionally, we might use additional monolingual corpora $D_{aux1}, \ldots, D_{auxN}$ in auxiliary languages.

In all our experiments, the tokenized input is processed by a single BPE model learned on the concatenation of the monolingual corpora, resulting in a joint vocabulary that enables all languages to use shared embeddings. Using a single BPE model for both the source and the target language is a common practice in NMT in general but in UNMT it is an essential step to allow the model to align its internal representations of the source and the target languages. In experiments which entail multilingual pre-training using auxiliary languages, the BPE model is learned on the concatenation of all available corpora.

In case of disbalanced monolingual corpora in terms of their size, simply concatenating all sentences can create a bias against low-resource languages [Conneau and Lample, 2019]. Therefore, we down-sample the larger corpus before learning the BPE model.

6.3.2 Architecture

The design of an NMT system needs to meet several requirements to be functional for unsupervised translation. Firstly, a significant number of parameters needs to be shared among the languages in order to allow the model to generate a shared latent space where meaning is represented regardless of the language it is expressed in [Lample et al., 2018c]. Secondly, the initialization of the model weights is vital to produce an initial solution and kick-start the training process [Conneau and Lample, 2019].

Our UNMT systems consist of a Transformer encoder and decoder, both of which are shared between the two languages. The shared encoder is essential for creating the shared space of cross-lingual latent representations, the shared decoder serves for regularization. The encoder and the decoder have the same 6-layer Transformer architecture with 8 attention heads and the hidden size of 1024, language embeddings, GELU [Hendrycks and Gimpel, 2017] activations, and a dropout rate of 0.1.

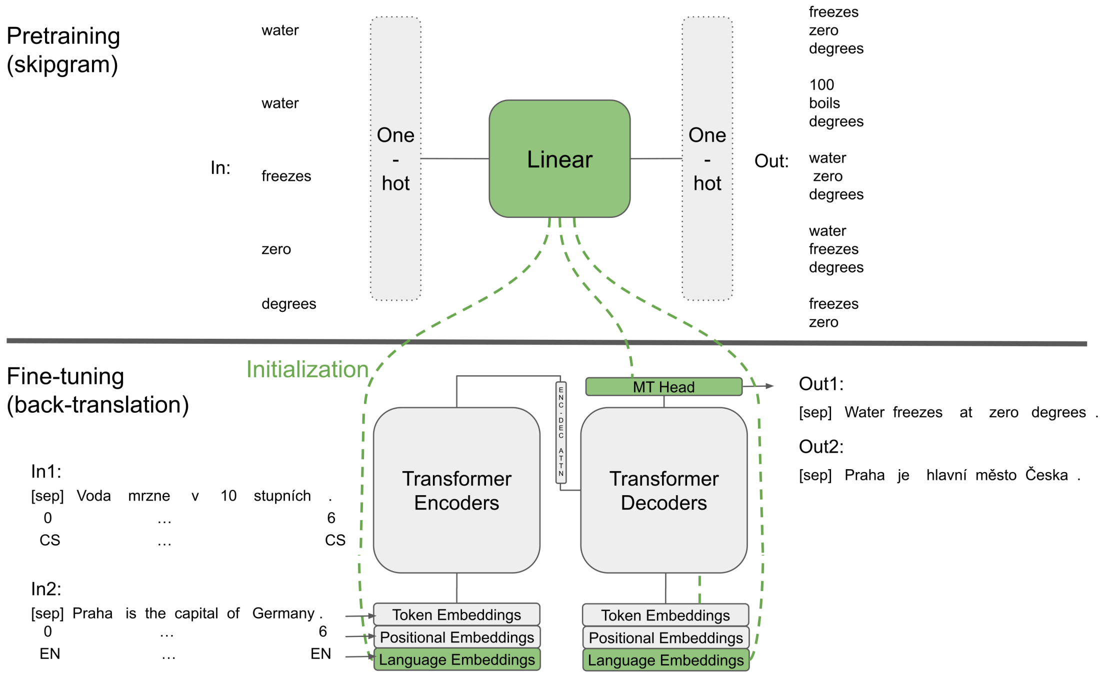


Figure 6.2: Design of an UNMT model with pre-trained embeddings. In the pre-training phase (top), a Skip-gram embedding model is trained on the concatenation of the monolingual corpora. Alternatively, the embeddings can be created by post-hoc alignment of monolingual embeddings (Section 6.1). The embedding layer weights and the tied output layer weight of the NMT model are initialized with the pre-trained embeddings. In the fine-tuning phase (bottom), the model is trained for translation using synthetic (back-translated) sentence pairs.

6.3.3 Pre-Training

There are several options to initialize the UNMT model:

- The encoder-decoder model is initialized randomly, only the token embedding weights are copied from a pre-trained word embedding model.
- The encoder-decoder model is initialized with weights from a masked language model pre-trained on the monolingual corpora and copied into both the encoder and the decoder as in Conneau and Lample [2019].
- The encoder-decoder model is initialized with weights of a bilingual or multilingual denoising autoencoder [Liu et al., 2020] pre-trained on the monolingual data in source and target languages, possibly in additional auxiliary languages.

The different pre-training strategies are illustrated in Figures 6.2 to 6.4.

Pre-trained Embeddings

Lampel et al. [2018a] showed that pre-training cross-lingual embeddings to initialize the embedding layers of an UNMT system provides enough translation signal to start the training.

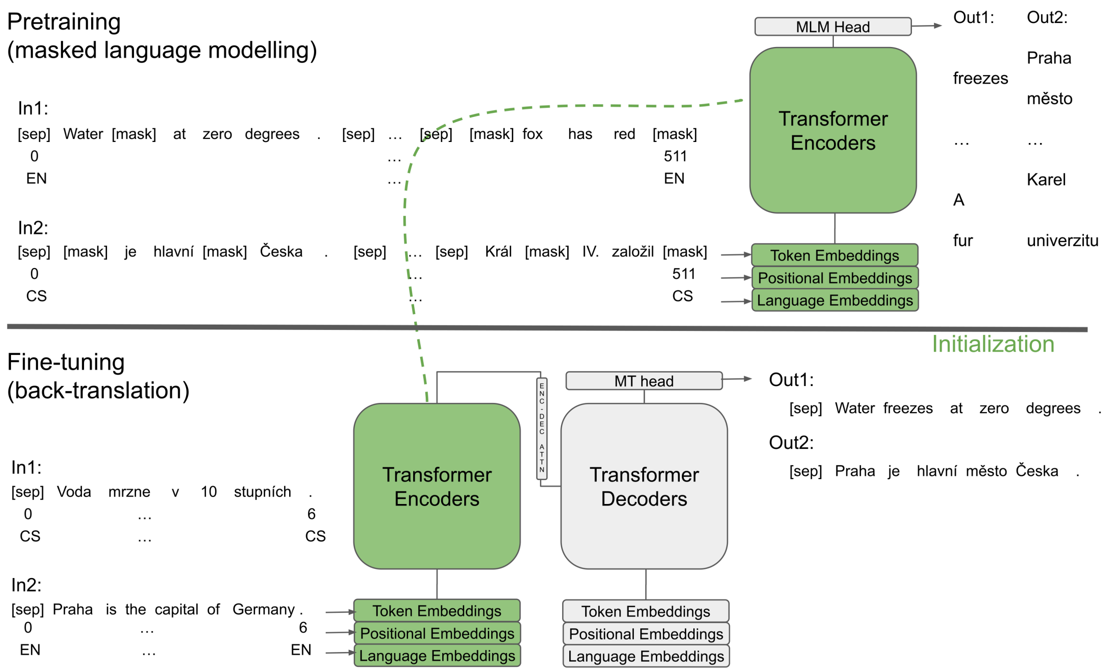


Figure 6.3: Design of an UNMT model with pre-trained encoder. In the pre-training phase (top), a masked language model is trained on the concatenation of the monolingual corpora. The encoder of the MT system is initialized with the pre-trained weights. Alternatively, the pre-trained encoder weights can be also copied to the decoder. In the fine-tuning phase (bottom), the model is trained for translation using synthetic (back-translated) sentence pairs.

While other pre-training strategies focused on the entire encoder or the full MT system later proved more efficient, our initial experiments used pre-trained embeddings.

If the source and the target language share the same alphabet, the simplest approach is to train embeddings jointly on the concatenation of the source and target monolingual corpora segmented into subword units Lample et al. [2018c]. If the alphabets are different or the simple approach does not provide enough cross-lingual signal for successful initialization, the cross-lingual embeddings are obtained by post-hoc alignment of monolingual embeddings as described in Section 6.1.

Pre-trained Encoder

The goal of unsupervised pre-training is to use unlabeled data to learn a general structure of text. Specifically, as shown in Chapter 3, MLM pre-training learns deep bidirectional representations which carry information on each word token and its context and can be used to initialize the encoder (and/or decoder) weights of a Transformer NMT system.

During multilingual MLM training, the model is presented with one text stream per language in every training step. Random tokens of a word sequence are masked and the model is trained to fill in the missing tokens given the context. In particular, 15% of tokens are randomly sampled to be either replaced by the [MASK] token ($p_{MASK} = 0.8$), replaced by a random token ($p_{RAND} = 0.1$) or not changed at all ($p_{KEEP} = 0.1$).

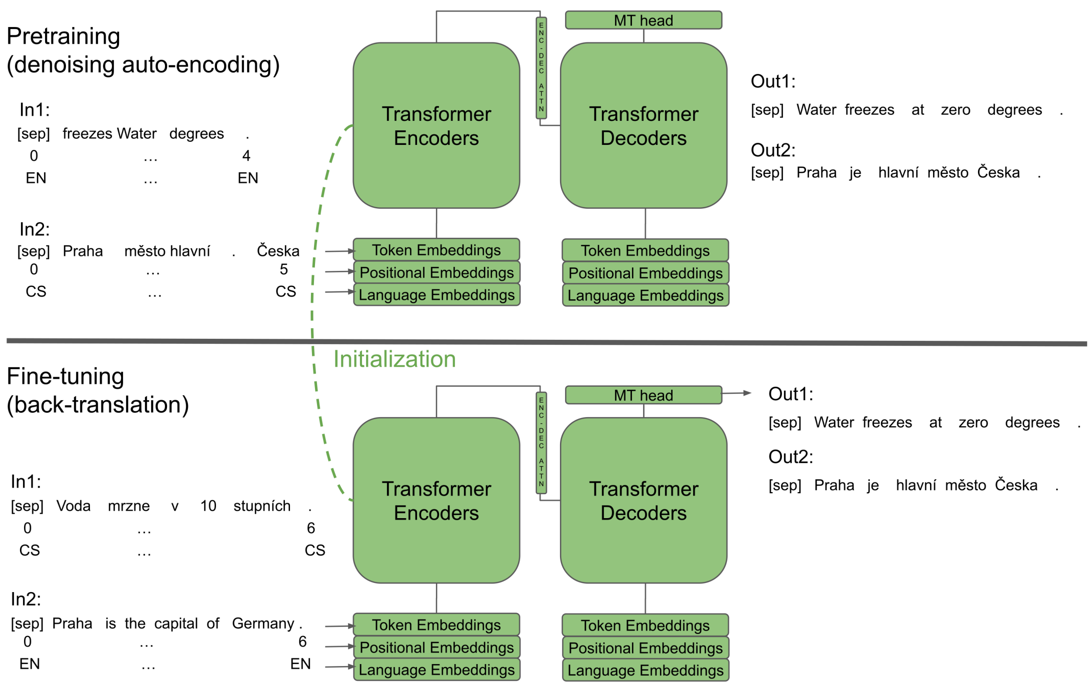


Figure 6.4: Design of an UNMT model pre-trained as denoising autoencoder. In the pre-training phase (top), the entire encoder-decoder model is pre-trained on the denoising task in multiple languages. In the fine-tuning phase (bottom), the model is trained for translation using synthetic (back-translated) sentence pairs.

We pre-train Transformer [Vaswani et al., 2017] encoders on monolingual corpora in multiple languages to learn a joint multilingual structure. The encoder can be pre-trained either only on texts in the source and the target language or on texts in other related languages as well.

In our experiments, we copy the pre-trained encoder weights not only to the encoder but also to the decoder.

Pre-trained Encoder-Decoder System

Denoising autoencoding (DAE) was initially used during the UNMT fine-tuning stage to stabilize the training [Artetxe et al., 2018d, Lample et al., 2018a]. However, we propose to use it already in the pre-training stage either as a replacement for MLM pre-training or as a subsequent step.

It is a monolingual training objective designed to teach the unsupervised model to recover proper sentences from corrupted input. The loss for each language l is the following

$$L_{AE}(\theta_{enc}, \theta_{dec}) = E_{x \sim D_l, \hat{x} \sim \mathrm{dec}(\mathrm{enc}(C(x)))}(\Delta(\hat{x}, x)) \tag{6.3}$$

where x is a sentence sampled from the monolingual data set D_l and $\hat{x}$ is the reconstructed sentence decoded from the noised version of x. The noise process $C(x)$ introduces random noise to a sentence x by dropping words with a probability p_{drop}, masking words with a probability $p_{mask} = 0.1$ and shuffling words within a tunable window size.

Conneau and Lample [2019] initialize their system with pre-trained MLM weights and later use DAE in the fine-tuning stage together with online back-translation. We propose a different method where we initialize the system with MLM weights, further train with the DAE objective and only then start fine-tuning for translation without DAE. The results of our approach are given in Section 7.3.

6.3.4 Fine-Tuning for Translation

Our UNMT systems are trained on synthetic data using online back-translation (sometimes also called on-the-fly back-translation) and on pseudo-parallel data with a standard translation objective.

Online Back-Translation

Online Back-Translation (OBT) is a bilingual objective for training an unsupervised model on synthetic translation samples generated by the model itself in previous iterations. This procedure is crucial for UNMT where we do not have access to any authentic parallel data resources. Back-translation is happening *on-the-fly* during training where the model first generates a batch of synthetic parallel data and immediately trains itself on it.

In the back-translation step, the model is first set to the inference mode and used to translate a batch of sentences. The synthetic translations serve as source sentences for a training step where the target side is the original sentence.

$$L_{BT}(\theta_{\text{enc}}, \theta_{\text{dec}}, l) = E_{x \sim D_l, \hat{x} \sim \text{dec}(\text{enc}(T(x)))}(\Delta(\hat{x}, x)) \tag{6.4}$$

where $T(x)$ is the translation model itself which generates a synthetic translation of sentence x.

Translation Supervised by Pseudo-Parallel Data (PseudoPar)

To fine-tune the model on pseudo-parallel data, the standard supervised MT objective is used. In every step of the training, a mini-batch of pseudo-parallel sentences is passed into the model which is trained to minimize the loss function

$$L_{PPST}(\theta_{\text{enc}}, \theta_{\text{dec}}) = E_{(x,y) \sim \textit{PseudoPar}, \hat{y} \sim \text{dec}(\text{enc}(x))} \Delta(\hat{y}, y) \tag{6.5}$$

where $(\theta_{\text{enc}}, \theta_{\text{dec}})$ is the trained model, (x, y) is a sentence pair sampled from the pseudo-parallel data set $PseudoPar$, and Δ is the cross-entropy loss.

Different methods to obtain pseudo-parallel data will be discussed in Chapter 5.

Translation Supervised by Phrase-based Translations (SynthPar)

In the first stage of UNMT fine-tuning, it can be beneficial to train on translations back-translated by a UPBMT system. Artetxe et al. [2018a] propose a robust system of training

UNMT models on a combination of synthetic translations by UPBMT and UNMT models, where the ratio of UPBMT translations decreases as the training progresses. In this thesis, we post-process the UPBMT translations to be more suitable for MT training. The loss function is identical to Equation 6.5, only the training data changes. Our experiments with SynthPar training will be described in Section 7.2.

6.3.5 Baselines

The baseline for our unsupervised MT experiments is the system of Conneau and Lample [2019] who pre-train both the encoder and the decoder on the bilingual MLM task and fine-tune using DAE and OBT.

7. Experiments & Results

We carried out several sets of experiments with different unsupervised MT approaches and different language pairs. In each section, we focus on a specific unsupervised technique: UPBMT (Section 7.1), combining UPBMT and UNMT (Section 7.2), unsupervised pre-training and initialization strategies (Section 7.3), and training on pseudo-parallel data (Section 7.4). Finally, we point out the limitations of unsupervised techniques (Section 7.5), and we train semi-supervised models in conditions where unsupervised MT fails (Section 7.6).

We have the following hypotheses regarding the outcomes of our experiments.

- We hypothesize that UNMT can benefit from different cross-lingual information brought into the training by synthetic corpora produced by phrase-based models (Section 7.2).
- In contrast to Artetxe et al. [2020] who claim that online back-translation tends to converge to the same translation quality regardless of the initialization strategy, we hypothesize that pre-training plays a key role in UNMT and the quality of the initial solution has a strong link to the final translation quality (Section 7.3).
- We hypothesize that existing UNMT models are not able to fully leverage the cross-lingual signal present in monolingual data and we propose a method to explicitly match similar sentences beforehand to present the model with the matched pseudo-parallel sentence pairs in addition to the unaligned monolingual texts (Section 7.4).

7.1 Phrase-Based Unsupervised MT

Our first experiments with unsupervised MT cover German (DE) to Czech (CS) translation. Although DE-CS is a high-resource language pair with access to several million parallel sentences, we artificially impose restrictions prohibiting the use of any parallel data to limit ourselves exclusively to monolingual data. This scenario was proposed in a WMT19 shared task on unsupervised MT from DE to CS and Sections 7.1 and 7.2 include passages from our system description paper [Kvapilíková et al., 2019].

In our initial experiments, we create UPBMT systems for translation in both directions. Following the strategy of Artetxe et al. [2018b] described in Section 6.2, we first train monolingual phrase embeddings, map them to the cross-lingual space, and use them to initialize the phrase table. We tune the hyperparameters of the model and run several iterations of back-translation, following the algorithm described in Figure 6.1. We then use the trained CS→DE model to translate the Czech monolingual corpus and create a synthetic parallel corpus which can be used later for training an NMT model.

7.1.1 Data

We trained our models on the NewsCrawl[1] corpus of newspaper articles collected over the period of 2007 to 2018. We tokenized and truecased the text using standard Moses scripts.

[1] `http://data.statmt.org/news-crawl/`

Sentences with less than 3 or more than 80 tokens were removed. The resulting monolingual corpora used for training of the unsupervised PBMT system consisted of 70M Czech sentences and 267M German sentences.

We performed further filtering of the Czech corpus before the NMT training stage. Since there are a lot of Slovak sentences in the Czech NewsCrawl corpus, we used the language tagger `langid` [Lui and Baldwin, 2012] to tag all sentences and remove the ones which were not tagged as Czech. After cleaning the corpus, the resulting Czech training set comprises 62M sentences.

Since small parallel data was allowed to tune the unsupervised system, we used *newstest2013* for development of the UPBMT system. Finally, we used *newstest2012* for model selection.

7.1.2 Model & Training

Phrase Embeddings

We first train phrase embeddings (up to trigrams) independently in the two languages. We use an extension of the word2vec Skip-gram model with negative sampling [Mikolov et al., 2013c] to train phrase embeddings. We use a window size of 5, embedding size of 300, 10 negative samples, 5 iterations and no subsampling. We restricted the vocabulary of each of the languages to the most frequent 200,000 unigrams, 400,000 bigrams and 400,000 trigrams.

Having trained the monolingual phrase embeddings, we use *VecMap* [Artetxe et al., 2018c] to learn a linear transformation to map the embeddings to a shared cross-lingual space. We use a list of Arabic numerals as the initial lexicon required to learn the mapping, as described in Section 6.1.

Unsupervised Phrase Table

The mapped embeddings are used to generate an unsupervised phrase table which is populated with source and target n-grams. For the sake of a reasonable phrase table size, only the 100 nearest neighbors are kept as translation candidates for each source phrase. The phrase translation probabilities are calculated as described in Section 6.2.2.

Initial UPBMT Model

We followed the Monoses[2] pipeline of Artetxe et al. [2018b] for our unsupervised phrase-based MT training. The phrase-based models are estimated using Moses [Koehn et al., 2007], with KenLM [Heafield, 2011] for 5-gram language modelling and `fast_align` [Dyer et al., 2013] for alignments. The feature weights of the log-linear model are tuned using minimum error rate training (MERT) using both an authentic parallel dev set and a synthetic back-translated dev set. The log-linear model of the initial system includes only the language model, translation probabilities and lexical weightings. Reordering model is introduced in further iterations.

[2] https://github.com/artetxem/monoses

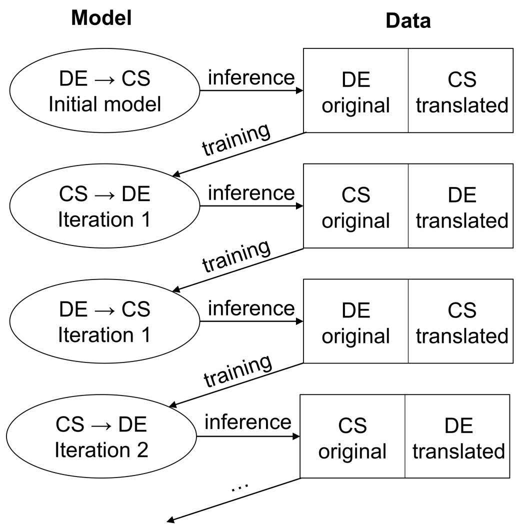


Figure 7.1: Step-by-step illustration of the iterative back-translation procedure.

	Authentic Dev Set		**Synthetic Dev Set**	
	DE→CS	**CS→DE**	**DE→CS**	**CS→DE**
Initial model	9.44	11.46	9.06	11.06
Iteration 1	11.11	**12.06***	4.61	12.92
Iteration 2	7.26	6.78	11.70	**14.22***
Iteration 3	1.06	2.32	**12.06**	14.07
Iteration 4	-	-	5.65	13.67
Iteration 5	-	-	11.69	14.18
Iteration 6	-	-	11.56	13.96

Table 7.1: Results of the PBMT models on newstest2012. The systems in the left two columns were tuned on the parallel newstest2013 (3K sentence pairs) and iteratively refined on 2M synthetic sentence pairs. The ones in the right two columns were tuned on a synthetic set (10K back-translated sentence pairs which remain fixed throughout the experiment) and iteratively refined on 4M synthetic sentence pairs. * indicates the best-performing CS→DE models selected for creating the synthetic parallel corpora.

Back-translation

The back-translation process is illustrated in Figure 7.1. Both DE→CS and CS→DE systems are needed at this step. The DE→CS system is used to translate a portion of the DE monolingual corpus to CS and create a synthetic parallel data set, which is then used to train the CS→DE system and the procedure cyclically continues. Note that we do not make use of the initial model for CS→DE. Once the synthetic parallel data set is created, the problem turns into a supervised one and we can use standard PBMT features, including the standard phrase table extraction procedure and the reordering model estimated on the aligned data sets.

Since back-translation is computationally demanding, we experiment with using a synthetic corpus of 2 and 4 million sentences for back-translation rather than translating the entire monolingual corpus.

7.1.3 Results & Discussion

We evaluate various UPBMT models to select the best candidate and observe an increasing translation quality with the first rounds of back-translation (Table 7.1).We note that even the initial model induced from the mapped embedding space produces meaningful translations with a BLEU score of 9.4 (DE→CS) and 11.5 (CS→DE). The quality increases with back-translation up to 12.1 and 14.2 BLEU, respectively.

We experiment with tuning the models both on an authentic parallel development set (3K sentence pairs) and a synthetic back-translated development set (10K sentence pairs). In the first scenario, possibly as a result of a smaller development set, the model started diverging after the first round of back-translation. In the second scenario, despite the synthetic nature of the development data, the models converge to a higher BLEU score. The best result is achieved after two and three rounds of back-translation for the CS→DE and DE→CS model, respectively (see the results in Table 7.1). As we were suspicious about the superior results of the systems tuned on synthetic rather than authentic data, we manually evaluated a random sample of 100 translations by the best-performing CS→DE system from each of the scenarios. After reviewing the translations and despite the BLEU results, we conclude that the best model refined with an authentic dev set produces superior translations especially in terms of word order.

Synthetic Corpora

We translated a random subset of 30M sentences of the target monolingual corpus from Czech to German using the two best performing CS→DE PBMT models (15M sentences each). The resulting synthetic corpus exhibits various errors, which we attempted to address as described in the following paragraphs. The final cleaned corpus size is 26M parallel sentences.

We detected three error patterns that are not easily detectable by BLEU but have a significant impact on human evaluation:

- German translations contaminated with words in other languages, especially Slovak;
- wrong word order (e.g. in contrast to the Czech word order, verbs in subordinate clauses and verbs following a modal verb should be placed at the end of a sentence in German);
- non-translated Czech words in German sentences (e.g. a German synthetic phrase *auf písčitém Küste* where the Czech word *písčitém (sandy)* remains non-translated);
- randomly mistranslated named entities (NEs) (e.g. *king Ludvik* translated as *king Harold* or *Brno* translated as *Kraluv Dvur*).

Heuristics to Improve Synthetic Corpora

In order to reduce the detrimental effects of the above errors on subsequent NMT training, we devised several post-processing strategies. Here we summarize the final versions of the corpora:

- *SynthPar-Initial*: The best-performing PBMT model was used for creating the synthetic training corpus for the initial training of the NMT model. We used a language tagger `langid.py` [Lui and Baldwin, 2012] to tag all synthetic sentences and remove the ones which were not tagged as German.

- *SynthPar-noCzech*: We cleaned the German side of the synthetic corpus by removing the Czech words which the PBMT model failed to translate and only copied. We identified words with Czech diacritics and replaced them on the German side with the `<unk>` token.

- *SynthPar-noCzech-reordered*: The corpus was further treated to eliminate the problem of wrong word order on the German side of the synthetic parallel corpus. We shuffled words in the synthetic German sentences within a 5-word window and mixed the re-ordered sentences into the original ones. We essentially doubled the size of the training corpus by first reordering odd-indexed sentences while keeping even-indexed sentences intact and then vice versa.

  The motivation for the augmentation was to prevent the NMT system from copying German source words directly into the target and support the NMT system in learning to handle word reordering. Ideally, the model should learn that German word order need not be strictly followed when translating to Czech. This feature is easy to observe in authentic parallel texts but the synthetic corpora are too monotone. We are aware of the fact that a 5-word window is not sufficient to illustrate the reordering necessary for German verbs but we did not want to introduce components which would be too language-specific to our technique.

- *SynthPar-noCzech-reordered-NEs*: The corpus was further treated to alleviate the problem of mistranslated NEs present in the data. NEs were identified in the monolingual Czech corpus by a NE recognition tagger NameTag[3] [Straková et al., 2014] trained on the Czech Named Entity Corpus 2.0[4] and aligned with the synthetic German size by *fast_align* [Dyer et al., 2013]. If the German counterpart was close enough (Levenshtein distance of at most 3) to the Czech original, we trusted the translation. If not, they were either removed from the corpus (geographic names) or copied from the source Czech size (numbers, personal names, institutions, media names, artifact names and time expressions as recognized by NameTag). More details about the procedure are given in Kvapilíková et al. [2019].

7.1.4 Takeaways

We created UPBMT models for translation between German and Czech. The models reach a BLEU score of over 10 points in both translation directions which can be considered a good result given that they were trained without any translation resources. However, the translations

[3] `http://ufal.mff.cuni.cz/nametag`

[4] `http://ufal.mff.cuni.cz/cnec/cnec2.0`

suffer from several repeating errors patterns: named entities are often mistranslated, the word order is wrong, and the translations include non-translated words from the source.

There is a potential for reaching a higher translation quality by training an NMT model on synthetic translations generated by the phrase-based model, especially if the translations are post-processed to prevent the known error patterns from contaminating the NMT training. The experiments with NMT models trained on post-processed UPBMT-generated corpora will be described in the following Section 7.2. For comparison of our UPBMT systems to a supervised benchmark, please also refer to the next section.

7.2 Hybrid Unsupervised MT

In this section, our goal is to improve the solution of the unsupervised DE→CS translation task from our previous experiments. The systems covered here are termed "hybrid" due to their neural model architecture which incorporates PBMT-generated synthetic data during training. We compare the results to a supervised benchmark to evaluate the gap between unsupervised and supervised models. Furthermore, we compare to a pivoting benchmark where we translate from German to Czech via English.

7.2.1 Data

Our models are trained on 26M sentence pairs where the source German size was generated by an unsupervised PBMT system described in Section 7.1.3 and the target Czech data of the same size is authentic from NewsCrawl. We train the model on several variations of the synthetic corpus described in Section 7.1.3 as we attempt to fix the errors present in the PBMT translations. We used *newstest3013* for validation and *newstest2019* for testing.

For training the supervised benchmark model, we used the following Czech-German parallel corpora available at the OPUS[5] website: OpenSubtitles (18M), MultiParaCrawl, Europarl, EUBookshop, DGT (5M), EMEA and JRC. The combined dataset has 26M sentence pairs.

For the training of the pivoting Czech-English-German model, we extracted 26M sentence pairs from the CzEng 1.6 corpus of Czech-English parallel data and 26M sentence pairs from the Europarl (2M), EUBookshop (10M) and OpenSubtitle (14M) corpora.

7.2.2 Model & Training

Model Architecture

We use the Transformer architecture described in Chapter 6 to train the DE→CS hybrid models.

Training on Synthetic Data

We experiment with different methods of MT training on synthetic parallel sentences. With regard to the terminology introduced in Chapter 6, we use online back-translation (OBT) where

[5] http://opus.nlpl.eu/

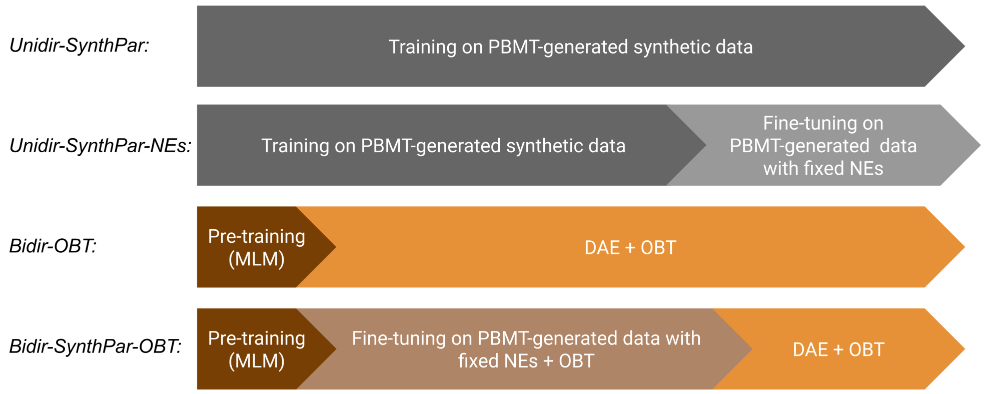


Figure 7.2: Schematic illustration of the training pipeline of our models. The size of the blocks is not proportional to training time.

synthetic sentence pairs are generated on-the-fly by the UNMT system, and compare to training on a full synthetic parallel corpus (*SynthPar*) generated by a UPBMT system prior to the training.

Our systems trained exclusively on the *SynthPar* corpus are unidirectional (DE→CS) whereas systems trained with OBT must be bidirectional (DE↔CS). While the unidirectional models are trained from scratch, the bidirectional models are pre-trained on the MLM task as described in Section 6.3.

Due to smaller and noisier training data, we set the dropout between Transformer layers to 0.3, which is higher than the typical dropout rate used in supervised systems. We train all models on 8 GPUs with a batch size of 2,400 tokens per GPU. We train our unidirectional models in the `Marian` toolkit [Junczys-Dowmunt et al., 2018] and the bidirectional models in the `XLM` toolkit [Conneau and Lample, 2019] with the same hyperparameters. The training pipeline of different systems is illustrated in Figure 7.2. The rest of the hyperparameters are given in the Appendix A.2.

- The *Unidir-SynthPar* system was trained on the initial synthetic data set *SynthPar-Initial* until convergence (249k steps) and then fine-tuned on the *SynthPar-noCzech* corpus for 12k steps, and for another 12k steps on *SynthPar-noCzech-reordered*.

- The *Unidir-SynthPar-NEs* system is a result of additional 12k fine-tuning steps on the *SynthPar-noCzech-reordered-NEs* corpus. Although the effect of this fine-tuning on the final translation might not be significant in terms of BLEU points, the problem of mistranslated named entities is perceived strongly by human evaluators and warrants an improvement.

- The *Bidir-OBT* is a UNMT model trained without any UPBMT component. It is a bidirectional model pre-trained on MLM and fine-tuned using online back-translation (OBT) and denoising autoencoding (DAE).

- The *Bidir-SynthPar-OBT* is a bidirectional model pre-trained on MLM and fine-tuned

	DE→CS		
	BLEU	**chrF++**	**COMET**
UPBMT	11.6	38.0	0.59
UNMT (Bidir-OBT)	14.6	39.2	0.72
*Unidir-SynthPar**	15.0	40.8	0.74
*Unidir-SynthPar-NEs**	14.3	40.5	0.74
Bidir-SynthPar-OBT	**16.7**	**42.6**	**0.79**
Benchmark-Supervised	*18.8*	*44.7*	*0.83*
Benchmark-Pivot	*15.1*	*40.1*	*0.75*

Table 7.2: Our unsupervised hybrid systems and their performance on newstest2019. For more details on the UPBMT models please refer to Section 7.1. * indicates models submitted for the WMT19 shared task. The WMT19 winning system [Marie et al., 2019] scored 3.4 BLEU points more than our best system but it was fine-tuned on 16.6k parallel sentences provided by the organizers for validation so it cannot be directly compared to our fully unsupervised systems.

for translation using a combination of the *SynthPar-noCzech* (70%) and *SynthPar-noCzech-reordered-NEs* (30%) corpora together with online back-translation. After 10k training steps, the synthetic corpus is dropped and the model is trained with online back-translation until convergence. We assume that keeping the less-fluent UPBMT-generated training corpus for too long might have a detrimental effect on the final quality.

Benchmarks

For comparison, we created an NMT system with the same architecture as our unsupervised models but trained it in a supervised way on the DE-CS parallel corpus of 8.8M sentence pairs (*Benchmark-Supervised*).

We also compare our results to the pivoting approach (*Benchmark-Pivot*) which is composed of two supervised models, DE→EN and EN→CS, trained on available parallel corpora. We eventually translate from German to Czech using the combination of the two models.

7.2.3 Results & Discussion

The scores of the systems on out test set are reported in Table 7.2. They demonstrate that we can significantly elevate translation quality by training an UNMT system on the UPBMT-generated synthetic data. COMET and chrF++ metrics are in line with the BLEU score.

Training on Synthetic Data

We were interested in evaluating the effect of employing synthetic data from various origins. The *Bidir-OBT* model was trained exclusively on UNMT-generated data, *Unidir-SynthPar* was trained exclusively on UPBMT-generated data, and *Bidir-SynthPar-OBT* was trained on both.

Due to the differences in their uni/bidirectional design and pre-training, the *Bidir-OBT* and *Unidir-SynthPar* models cannot be assessed only based on the nature of the data used for training. While *Bidir-OBT* is trained for translation in both directions indicated by language embeddings, the *Unidir-SynthPar* model specializes in DE→CS translation which puts it at an advantage. On the other hand, the *Bidir-OBT* model was pre-trained on the MLM task where it had the opportunity to internally align cross-lingual representations and use them for unsupervised translation.

Bidir-OBT is outperformed by *Unidir-SynthPar* but the difference is not statistically significant. However, we clearly observe the benefit of combining the two approaches to synthetic data generation. Upon comparison of the bidirectional *Bidir-OBT* and *Bidir-SynthPar-OBT* models which differ only in one training stage (see Figure 7.2), we conclude that incorporating UPBMT-generated data into the first stages of UNMT training brings a significant improvement of $\sim$2 BLEU points over the *Bidir-OBT* system trained using online back-translation only. The UPBMT-generated synthetic corpus is a valuable source of cross-lingual signal to the UNMT model.

Online Back-Translation

It must be noted that while the UPBMT-generated translations were produced by a finished model, the UNMT-generated synthetic sentence pairs are produced on-the-fly by OBT and are of progressively increasing quality, starting at translations full of repeating punctuation marks and copied (non-translated) words. We had a closer look at the quality of the back-translated sentences and made the following observations.

- Already after 1k training steps the structure of OBT translations starts corresponding to the source sentence.

- It lasts several more iterations to get rid of most mistranslations and copied German source words. For example, at 1k training steps, the German sentence *"Krähen stehen unter Naturschutz."* ("*Crows are protected by nature conservation laws.*") is translated as *"Krämerovy houby stojí mimo Naturschutz"*, where *"Naturschutz"* is copied and *"Krämerovy houby"* (*"Krämer's mushrooms"*) is a complete mistranslation motivated by a subword overlap of the first word.

- Although the translation is subword-based, it happens only rarely that a part of a word would remain non-translated, e.g. *"Erfolgverprechende"* (*"promising"* translated as a non-existent word *"Erfolgtivní"*. Even in long German compound words which mostly get copied as a whole (e.g. *"Witterungsbedingungen"*). This is likely the result of MLM pre-training and possibly also the fairly big BPE vocabulary of 60k units.

Named Entity Translation

We showed in Section 7.1 that UPBMT systems suffer from frequent mistranslations of named entities. After our experiments with UNMT and hybrid systems, we confirm that name translation is also a challenge for UNMT and hybrid systems.

	Sentences with NEs	Sentences with no NEs
Unidir-SynthPar	28%	26%
Unidir-SynthPar-NEs	52%	28%
No winner	20%	46%

	Sentences with NEs	Sentences with no NEs
Bidir-OBT	22%	18%
Bidir-SynthPar-OBT	38%	40%
No winner	40%	42%

Table 7.3: Results of manual evaluation of three systems on a stratified subset of the validation data set created by randomly selecting 100 sentences with NEs and 100 sentences without NEs.

In Section 7.1.3, we attempted to mitigate the problem by post-processing the UPBMT-generated corpus. This corpus was used in the training of the *Unidir-SynthPar-NEs* and *Bidir-SynthPar-OBT* models. Table 7.3 summarizes the improvement we gained by introducing such named entity treatment. We manually evaluated the following systems on a stratified subset of the validation data set created by randomly selecting 100 sentences with NEs and 100 sentences without NEs: *Unidir-SynthPar* against *Unidir-SynthPar-NEs* and *Bidir-OBT* against *Bidir-SynthPar-OBT*. The results show that despite the decrease in BLEU score we see in Table 7.2, fine-tuning of the *Unidir-SynthPar* model on a synthetic corpus with amended NEs proved beneficial in 52% of tested sentences which included NEs and it did not harm in 20% of sentences. When comparing the two systems on sentences with no NEs, their performance is very similar.

The translations by *Bidir-SynthPar-OBT* are superior to the translations by *Bidir-OBT* both in terms of named entities and general quality which is in line with the results from Table 7.2 and confirms that training on the *SynthPar* corpus with NE treatment reduces the problem of mistranslated names.

Translations by bidirectional models with MLM pre-training suffer less from the problem of mistranslated NEs than the unidirectional models which rely on the PBMT synthetic corpora for all cross-lingual signal. Nevertheless, incorrectly translated names continue to be one of the most serious errors generated by unsupervised translation systems. See Table 7.4 for a sample translation.

7.2.4 Takeaways

The UPBMT-generated synthetic corpus serves as a valuable source of cross-lingual signal for UNMT models. Such hybrid models consistently achieve higher quality compared to pure neural models. The synthetic corpus brings the most value at the beginning of the training when the UNMT model is not yet able to generate meaningful translations on its own. Once the UNMT model attains a satisfactory level of quality, it is advisable to phase out the initial synthetic corpus, as it can potentially impede further training. If the UNMT system is initialized well, the training starts successfully, and at 1k training steps we observe that the UNMT starts generating meaningful translations.

Source	**Phrase**
Original	Der Lyriker **Werner Söllner** ist IM **Walter**.
Reference	Básník **Werner Söllner** je tajný agent **Walter**.
PBMT	Český prozaik **~~Miroslav Mišák~~** je agentem StB **~~Josef~~** .
Unidir-SynthPar	Prozaik **~~Filip Bubeníček~~** je agentem StB **~~Josefem~~**.
Unidir-SynthPar-NEs	Prozaik **~~Filip~~ Söllner** je agentem StB **~~Ladislavem Bártou~~**.
Bidir-OBT	Lyrik **~~Jiří~~ Söllner** je IM **Walter~~man~~**.
Bidir-SynthPar-OBT	Prozaik **Werner Söllner** je IM **Walter~~man~~**.

Table 7.4: Sample translations showing that fine-tuning on synthetic corpus with cleaned NEs (*Unidir-SynthPar-NEs* and *Bidir-SynthPar-OBT*) alleviates a part of the NE problem. However, note the imperfect translation of *Lyriker* as *novelist* rather than *poet*. The bidirectional systems seem to be more prone to copying which can help for some NEs but also hurt, e.g. copying the word *IM* rather than recognizing it as a shortcut for *"inoffizieller Mitarbeiter"* and translating it as *secret agent*.

In our view, one of the most significant types of translation errors in unsupervised systems involves a high frequency of randomly mistranslated named entities. This problem is not adequately addressed by the BLEU score but it has a considerable impact on the perceived translation quality. We have concentrated our efforts on mitigating this issue during the fine-tuning of the UNMT system by rectifying NEs in the synthetic training corpus. Some names where deleted, others were replaced by their copy. While our approach may not be flawless, we believe that an omitted named entity or a non-translated named entity is less detrimental than a randomly substituted one. Unfortunately, this approach to amending NEs can only be applied to languages with a name tagger available, which is not the case for many truly low-resource languages.

In the next experiments, we will be working only with bidirectional UNMT systems and focus on their ability to create cross-lingual internal representations in the pre-training stage of the training pipeline.

7.3 Effect of Pre-Training Strategies

A number of pre-training and initialization strategies have been proposed since the inception of UNMT. The first models were initialized with cross-lingual embeddings [Lample et al., 2018a, Artetxe et al., 2018d]. A significant increase of translation quality came after discovering the benefits of cross-lingual MLM pre-training [Conneau and Lample, 2019]. The latest UNMT systems rely on multilingual pre-training of the entire encoder-decoder model on some variation of a denoising task [Liu et al., 2020]. We measure the effect of different pre-training strategies on the final translation quality and propose a combined approach which yields the most favorable results across different language pairs. Furthermore, we hypothesize that pre-training on multiple languages could help the model create a language-neutral internal representation space and lead to a more effective initialization of weights for unsupervised translation.

A more thorough exploration of the multilingual aspects of UMT is not the goal of this

	DE-HSB	CS-DE	EN-KA	EN-KK	EN-UK
train (mono)	29.4M/0.9M	26M/29.4M	17.1M/6.6M	17.1M/7.7M	17.1M/17.3M
train (para)	-	5M	1M	-	-
dev	2k	3k	1k	1k	1k
general test	1.6k	-	1k	1k	1k
legal test	-	-	1k	1k	1k

Table 7.5: Number of sentences used for unsupervised training and evaluation. The *para* data was only used for training the transfer learning benchmarks.

thesis but it was studied in Sun et al. [2020] or Üstün et al. [2021].

We evaluate the effect of pre-training strategies on the following language pairs in the legal domain: German-Upper Sorbian (DE-HSB), English-Georgian (EN-KA), English-Kazakh (EN-KK) and English-Ukrainian (EN-UK).

7.3.1 Data

The DE and HSB monolingual training data as well as the parallel validation and test sets were provided in the WMT22 unsupervised shared task [Weller-Di Marco and Fraser, 2022]. The auxiliary CS monolingual corpus is a random selection of 26M sentences from NewsCrawl. The monolingual training data for EN, KA, KK and UK come from the Oscar[6] corpus and the MT4All shared task [de Gibert Bonet et al., 2022] which provided domain-specific data from the legal domain. The training data summary is given in Table 7.5. The English-centric validation and test sets were taken from the Flores Evaluation Benchmark [Costa-jussà et al., 2022]. In addition, the legal test sets from the MT4All shared task were used for evaluation.

For our side experiments with supervised pre-training on parallel texts in Czech-German (CS-DE) and English-Georgian (EN-KA). For EN-KA, we used the CCAligned corpus available at OPUS. For CS-DE, we trained on a random sample of 5M parallel sentences from the OPUS website: OpenSubtitles, MultiParaCrawl, Europarl, EUBookshop, DGT, EMEA and JRC.

The data was tokenized and split into BPE units using the fastText [Joulin et al., 2016] library. We shared one BPE vocabulary of 55k entries for EN-KA-KK-UK and another vocabulary of 18k entries for CS-DE-HSB.

7.3.2 Model & Training

Model Architecture

We use the Transformer architecture described in Chapter 6 to train all our models.

Pre-training Strategies

We experiment with the following pre-training tasks introduced in Chapter 6 to determine the optimal strategy for further experiments:

1. Skip-gram for static embeddings with post-hoc mapping;

[6] `https://oscar-project.org/`

2. cross-lingual masked language modelling (MLM);

3. denoising autoencoding (DAE);

4. MLM followed by DAE.

The details of MLM and DAE pre-training were given in Chapter 6. All models are fine-tuned using OBT or OBT+DAE, depending whether DAE was a part of the pre-training stage.

Both MLMs and DAEs are either trained in a bilingual fashion on a combination of samples in the source and the target languages, or in a multilingual fashion on samples in several auxiliary languages. The languge of the sentence or the text stream is indicated to the model by language embeddings. We pre-train both bilingual and multilingual versions of the MLMs and DAEs to be able to draw conclusions about the effects of multilingual pre-training.

MLM pre-training was proposed by Conneau and Lample [2019], while DAE was later used by Liu et al. [2020] for pre-training of the *MBART* model which brought state-of-the-art results in UMT. We compare the two approaches and propose a modification where we first pre-train an MLM encoder, use it to initialize both the encoder and the decoder of a full Transformer model and continue pre-training as a denoising autoencoder. While MLM pre-training helps the encoder and decoder separately to create cross-lingual representations, DAE prepares the full model for conditional text generation. We believe that combining the two strategies will allow the model to benefit from both.

Furthermore, combining MLM and DAE allows us to drop the denoising task from the fine-tuning stage. The denoising training objective was proposed by Artetxe et al. [2018d] and Lample et al. [2018a] to stabilize the training of UNMT. We found that dropping it does not cause any harm, provided that DAE was a part of the pre-training stage. Therefore, this method also eases some computational burden as we pre-train the model only once, enabling subsequent experiments with various fine-tuning strategies. This is especially useful in the case of multilingual pre-training. We will focus on fine-tuning the models for translation in the next round of experiments which will be described in Section 7.4.

Training Details

Monolingual embeddings are trained on the subword-segmented training corpus using the Skip-gram approach described in Section 3.1.1. We keep the default hyperparameters of the word2vec[7] implementation and train the embedding model for 5 epochs using 10 negative samples and a 5-word window. We align the embeddings into a bilingual embedding space using the MUSE[8] library where we train an adversarial model with 5 iteration of refinement.

The MLMs are trained on mini-batches with 64 text streams (fixed-length segments of texts which go beyond sentence boundaries) per batch, 256 tokens per stream. 15% of the tokens are masked. The details of the masking of input sentences were given in Section 6.3.3. All models are trained on 8 GPUs.

[7] `https://github.com/tmikolov/word2vec`
[8] `https://github.com/facebookresearch/MUSE`

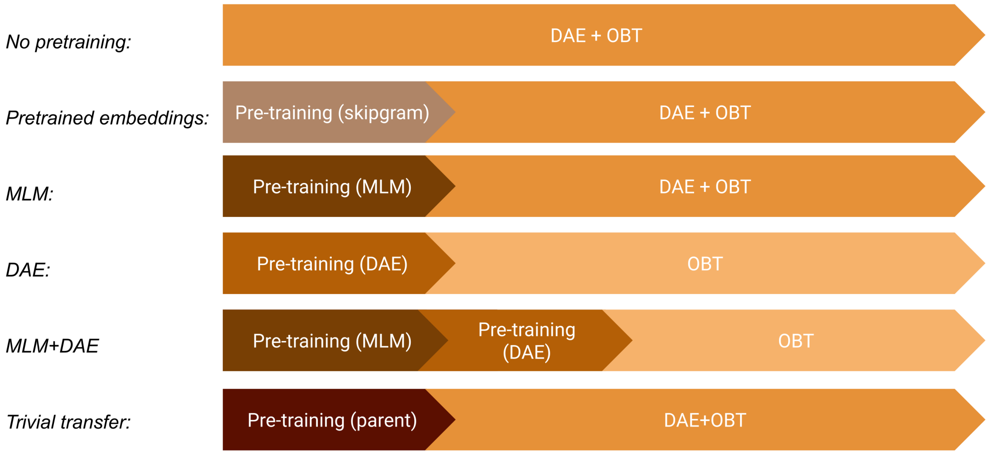


Figure 7.3: Schematic illustration of the training pipeline of our models. The size of the blocks is not proportional to training time.

The DAEs are trained on data noised by shuffling tokens within a 3-token window, dropping words with a probability $p_{drop} = 0.1$ and masking words with a probability $p_{mask} = 0.1$. We train sentence-by-sentence on 8 GPUs with 3,400 tokens per batch.

Fine-tuning for MT using online back-translation is run on 8 GPUs with mini-batches of 3,400 tokens per GPU using Adam optimization with a linear warm-up `(beta1=0.9, beta2=0.98, lr=0.0001)`. Greedy decoding is used during back-translation. For evaluation, we use beam search with beam size set to 6.

7.3.3 Results & Discussion

Unsupervised Pre-training Strategies

In contrast with the conclusions of Artetxe et al. [2020], we argue that the translation quality of UMT systems is highly sensitive to the choice of the pre-training and initialization strategy. The initial solution ignites further training by back-translation and if the pre-training stage fails to deliver this solution, the model never starts learning. It can be observed in the case of EN-KK translation where only the MLM pre-training allows the model to initiate the back-translation process while other pre-training strategies trap the model in a suboptimal solution with no translation capabilities, similar to random initialization.

We select the best performing versions (bilingual or multilingual) of the proposed pre-training strategies (MLM, DAE, and MLM followed by DAE) and evaluate their benefit over random initialization of the model with no pre-training at all and over the weak initialization of the embedding layer only. The results are summarized in Table 7.6.

As expected, a meaningful initialization is one of the key features of the UMT design and without it the models are impossible to train. At minimum, the embeddings need to start with some cross-lingual signal, although this signal might not be strong enough to yield high translation quality, particularly for linguistically distant languages. In line with the conclusions of Conneau and Lample [2019], we observe a major increase in DE-HSB translation quality ($\sim$

	DE-HSB	HSB-DE	EN-KA	KA-EN	EN-KK	KK-EN	EN-UK	UK-EN
No pre-training	2.9	2.7	0.8	1.0	1.2	1.6	0.5	0.8
Pretr. embeddings	8.8	8.9	-	-	-	-	-	-
MLM	21.6	22.2	4.4	5.2	**3.6**	**4.9**	7.4	10.7
DAE	21.2	24.1	1.8	2.2	1.9	2.7	3.7	5.4
MLM + DAE	**27.3**	**30.6**	**5.9**	**6.3**	1.2	1.3	**10.1**	**13.5**
Trivial transfer	***28.9***	***33.6***	-	-	*1.1*	*1.2*	-	-

Table 7.6: The impact of different pre-training strategies on translation quality measured on the validation sets by BLEU score.

13 BLEU) upon the introduction of MLM pre-training.

While we cannot establish a clear winner between MLM and DAE pre-training strategies, we reached a significant improvement with a combination of the two. Further pre-training of the initialized model with a DAE objective can boost performance by additional ∼6 BLEU points in the case of DE-HSB translation, and ∼8 BLEU points in the case of UK-EN translation. However, we observe that the combined strategy fails to deliver the initial solution for the EN↔KK model. In the subsequent experiments, we will see how the situation can be alleviated in the fine-tuning stage (Section 7.4).

Trivial Transfer Approach

We decided to benchmark our unsupervised pre-training strategies against a trivial transfer learning approach based on [Kocmi and Bojar, 2018]. We pre-trained two "parent" supervised models on parallel data: DE-CS translation model on 5M parallel sentences and EN-KA translation model on 1M parallel sentences. We used these to initialize the "children" (DE-HSB model and EN-KK model, respectively) trained in an unsupervised way using OBT. The only requirement for using this method is a shared vocabulary between the "parent" and "child" models which is met in our setup.

The outcomes are documented in the last row of Table 7.6, and they lead to contrasting conclusions for the two language pairs. While for DE-HSB translation, the DE-CS pre-training leads to an improvement of up to 3 BLEU points over the best unsupervised pre-training strategy, EN-KK unsupervised translation learning fails to ignite from the EN-KA initialization and results in downward-sloping training curves.

We conclude that learning by back-translation can be bootstrapped from a "parent" translation model but only if the two language pairs are closely related (such as CS and HSB). This is in contrast with the conclusions that hold for supervised MT where a successful transfer of translation knowledge occurs even for unrelated languages [Kocmi and Bojar, 2018]. In practical use cases of low-resource MT from monolingual data, if a related language pair with a shared source or target language and abundant parallel data is available, it seems reasonable to use it for pre-training rather than relying on one of the fully unsupervised pre-training strategies.

Multilingual vs. Bilingual Pre-training

Finally, we aim to determine whether it is beneficial to include auxiliary languages in the pre-training stage. For DE-HSB translation, we compare the models pre-trained on bilingual (DE-HSB) data only to models pre-trained on multilingual (CS-DE-HSB) data and hypothesize that adding another Slavic language into the pipeline may increase the final translation quality. For the remaining language pairs, we pre-train both bilingually and on the combination of all EN, KA, KK, and UK training corpora. Note that these languages are linguistically very diverse and use different scripts: Latin (EN), Cyrillic (KK, UK), and Mkhedruli (KA).

Table 7.7 shows BLEU scores on the validation sets. Asides from the languages included in the pre-training, differences in translation quality may also stem from the number of steps in each training stage which varies across experiments and might influence the results. Therefore, we report the duration of the training in Table 7.7 together with the results.

It proved to be difficult to draw a universal conclusion in favor of either the bilingual or the multilingual pre-training setup. In contrast to our initial hypotheses, bilingual MLM pre-training is superior to multilingual MLM for DE-HSB translation, and leads to a difference in BLEU score of up to 3.8 BLEU points. It must be noted that the results are likely also influenced by the fact that the bilingual MLM model has seen 6 times more DE and HSB sentences than the multilingual model. Conversely, the situation is the opposite for the English-centric language pairs where the multilingual model performs better, despite the linguistic dissimilarity, and despite the fact that the bilingual models were trained for slightly longer. We take the weights from the best performing pre-trained MLM (the bilingual model for DE-HSB and the multilingual model for the remaining language pairs) and train on a multilingual or bilingual denoising task. Table 7.7 shows that bilingual DAE pre-training of MLM-initialized model is more effective than its multilingual counterpart. Particularly, multilingual denoising of the EN-KA model harms the MLM initialization and leads to a similar result as if no pre-training happened at all. For other language pairs, bilingual pre-training is also superior. The difference is especially pronounced in the case of the DE-HSB translation where it amounts to 6–7 BLEU points.

Given the state-of-the-art MT results of the *MBART* model [Liu et al., 2020] pre-trained via multilingual denoising, our initial hypothesis was that this pre-training strategy would lead to competitive results in our setup as well. However, we were not able to fully exploit the benefits of multilingual DAE pre-training in our conditions. There are several possible reasons for that. First of all, the *MBART* model has substantially more parameters (12-layer Transformer with 16 heads and internal dimension 1024 vs. 6-layer Transformer with 8 heads and internal dimension 512) and it was trained on entire documents in 25, resp. 50 languages. Furthermore, *MBART* relies on a slightly different noise function to corrupt the training data. Pre-training a smaller model on three or four languages did not have the desired effect on final translation quality.

MLM	**DE-HSB**	**HSB-DE**	**EN-KA**	**KA-EN**	**EN-KK**	**KK-EN**	**EN-UK**	**UK-EN**
multilingual	17.8	20.6	**4.4**	**5.2**	**3.6**	**4.9**	**7.4**	**10.7**
	CS,DE,HSB (51k)		EN,KA,KK,UK (33k)					
bilingual	**21.6**	**22.2**	3.5	**4.7**	2.6	4.1	3.7	7.6
	DE,HSB (**304k**)		EN,KA (50k)		EN,KK (40k)		EN, uk (**71k**)	

DAE	**DE-HSB**	**HSB-DE**	**EN-KA**	**KA-EN**	**EN-KK**	**KK-EN**	**EN-UK**	**UK-EN**
multilingual	**21.2**	**24.1**	1.8	2.2	**1.9**	**2.7**	3.7	5.4
	CS,DE,HSB (200k)		EN,KA,KK,UK (71k)					
bilingual	19.2	21.4	-	-	**1.8**	**2.5**	-	-
	DE,HSB (195k)		-		EN,KK (**189k**)		-	

MLM + DAE	**DE-HSB**	**HSB-DE**	**EN-KA**	**KA-EN**	**EN-KK**	**KK-EN**	**EN-UK**	**UK-EN**
multilingual	21.3	23.4	0.7	0.8	**1.2**	**1.3**	**9.3**	**12.8**
	CS,DE,HSB (51k+100k)		EN,KA,KK,UK (33k+82k)					
bilingual	**27.3**	**30.6**	**5.9**	**6.3**	**1.2**	**1.3**	**10.1**	**13.5**
	DE,HSB (**304k**+102k)		EN,KA (33k+67k)		EN,KK (33k+67k)		EN,UK (33k+76k)	

Table 7.7: The impact of bilingual and multilingual pre-training strategies on translation quality measured by BLEU score on the validation sets. The highest BLEU scores per language pair and category are indicated in bold. If more than one figure is bold, the difference is not statistically significant. We also report training duration in terms of the number of training steps and indicate if it is considerably higher in either the bilingual or the monolingual setup.

7.3.4 Takeaways

We experimented with different pre-training tasks and conclude that the translation results are highly sensitive to the choice of the pre-training strategy. For most of our models, the most effective approach is to first initiate the weights based on MLM, follow it with DAE pre-training, and only then start fine-tuning for translation. The combination of these two objectives in the pre-trainning stage is novel, as most authors use either one or the other. Although DAE is customarily used later in the fine-tuning stage of the UNMT pipeline to stabilize the training, we observe a positive impact of isolating it into the pre-training stage. However, especially when auxiliary languages are used, this strategy carries the risk of distorting the initial solution and hindering further learning. In such cases, reverting to the approach of MLM pre-training and OBT+DAE fine-tuning is the optimal choice.

We also explored the benefits of transfer learning for unsupervised MT and we conclude that if a related language pair with parallel data is available, it is advisable to consider initializing the model with a supervised MT model trained for that related pair. However, the translation transfer does not work for unrelated languages.

It must be noted that translation quality for the most linguistically dissimilar language pairs (EN-KK and EN-KA) is low (below 7 BLEU points). We will be working on improving the translation quality for remote languages in the next experiments.

7.4 Boosting Unsupervised MT with Pseudo-Parallel Data

In this section, we measure the effect of incorporating pseudo-parallel sentences into unsupervised MT. We hypothesize that they can serve as a new source of cross-lingual information that the model can benefit from. Although pseudo-parallel sentences are not perfect translation equivalents, we believe that they can improve the translation quality nonetheless, especially when used in the beginning of the training.

We employ the same methodology as in our previous experiments described in Section 7.3, and we incorporate an additional training step where the pseudo-parallel corpus is used to train the NMT system with a standard supervised MT objective. We experiment with different training schedules to determine when to incorporate the pseudo-parallel data and when to remove it from the training.

The experiments from this section were published in Kvapilíková and Bojar [2023] and some portions of text and tables are taken verbatim from there. We evaluate on the same language pairs as in the previous Section 7.3 (DE-HSB, EN-KA, EN-KK, EN-UK).

7.4.1 Data

We use the same data as described in Section 7.3.1 for the experiments in this section.

7.4.2 Model & Training

Model Architecture

We use the Transformer architecture described in Chapter 6 to train all our models.

Pseudo-Parallel Corpus Creation

We first create a pseudo-parallel corpus as described in Chapter 5. We use the *XLM-100* model fine-tuned on English-German synthetic sentence pairs as our sentence encoder for parallel corpus mining. To measure its ability to create representations with a high level of multilingualism for the languages of our interest, we evaluate its performance on an auxiliary task of parallel corpus mining (PCM). For each language pair, we randomly select 200k sentences from the monolingual training data, mix in the parallel validation set, and measure the precision and recall of the model when trying to reconstruct it.

Since *XLM-100* was trained on 100 languages and HSB is not one of them, we fine-tune the model on DE and HSB sentences before using it to mine parallel sentences for this language pair. We stop fine-tuning when the quality of the mined corpus starts deteriorating. We determine the optimal length of fine-tuning on the PSM task and observe that both precision and recall start slowly decreasing after the model had seen 500k sentences.

To retrieve sentence embeddings from the trained model, we mean-pool the encoder outputs from the fifth-to-last layer across sentence tokens (the layer and aggregation choice ex-

	DE-HSB	EN-KA	EN-KK	EN-UK
Precision	87.08	44.8	49.3	67.4
Recall	76.15	44.4	42.4	74.2
F1	81.25	44.6	45.6	70.6
Threshold	1.034	1.023	1.022	1.026

Table 7.8: The evaluation metrics on the PSM task and the respective mining thresholds.

	DE-HSB	CS-DE	EN-KA	EN-KK	EN-UK
train (mono)	29.4M/0.9M	26M/29.4M	17.1M/6.6M	17.1M/7.7M	17.1M/17.3M
train (pseudo-para)	770K	-	230K	169K	496K

Table 7.9: Sizes of monolingual corpora and the number of pseudo-parallel sentences mined from them.

plained in Section 5.2). We search the embedding space as described in Equation 5.1 and Equation 5.2. We select a threshold T that maximizes the F1 score on the PSM task. Table 7.8 lists the precision and recall of all sentence encoders used for mining together with the optimal mining threshold. The amount of mined parallel sentences used for the MT training is given in Table 7.9.

UNMT Pre-training

We follow the results of the experiments in Section 7.3 when selecting the pre-training strategy for our experiments. We pre-train one multilingual language model for EN-KA-KK-UK and one bilingual language model for DE-HSB using the MLM objective. The weights from the pre-trained language models are copied into both the encoder and the decoder of the respective bilingual NMT models. The initialized NMT model for each language pair is then further pre-trained with the denoising autoencoding loss on the two languages until convergence. The details of the denoising task are identical to Lample et al. [2018a].

UNMT Fine-Tuning

We experiment with different fine-tuning strategies for unsupervised machine translation as illustrated in Figure 7.4. For each language pair, all translation models are initialized with the same weights obtained in the pre-training stage described in the previous paragraph.

OBT (baseline) models are fine-tuned solely with the iterative back-translation loss.

PseudoPar models are fine-tuned with the standard supervised MT loss on our pseudo-parallel corpora.

OBT+PseudoPar models are fine-tuned simultaneously with the iterative back-translation loss on the monolingual sentences and with the standard MT loss on the pseudo-parallel sentence pairs.

OBT+PseudoPar$\mapsto$*OBT* models are a continuation from different checkpoints of the *OBT+PseudoPar* models where the supervised MT objective is dropped and the training continues with iterative back-translation only. We experiment with different checkpoints to

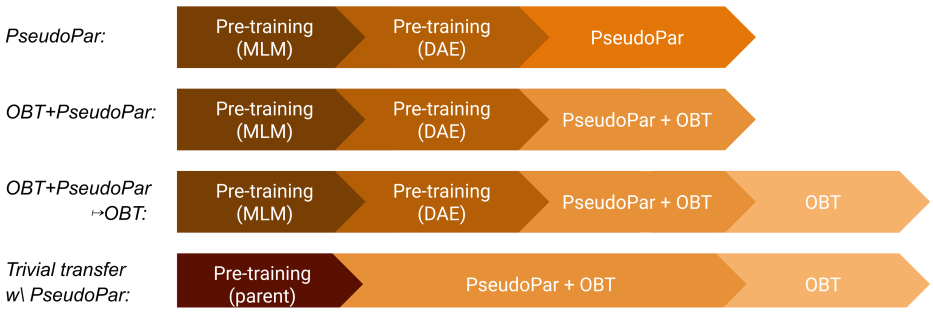


Figure 7.4: Schematic illustration of the training pipeline of our models. The size of the blocks is not proportional to training time.

find the optimal point to switch the training.

Training Details

Training configuration is identical to Section 7.3.

Benchmarks

The baseline for our approach is an improved model of Conneau and Lample [2019] with an extra pre-training step on the DAE task for better performance. We initialize the baseline model with the weights of a cross-lingual MLM model, further pre-train as a denoising autoencoder and fine-tune with iterative back-translation.

We benchmark our results against MT systems of de Gibert Bonet et al. [2022] trained as a baseline for the MT4All shared task according to the methodology of Artetxe et al. [2019a], and against Shapiro et al. [2022] who won the WMT22 DE-HSB unsupervised task with a multilingual system that was pre-trained according to the mBART [Liu et al., 2020] methodology and fine-tuned on synthetic texts generated by a phrase-based system.

To challenge the relevance of unsupervised MT in the world of large language models, we also translate our test sets by the GPT-3.5 Turbo model[9] using the ChatGPT API and compare to our results.

7.4.3 Results & Discussion

We observed a significant improvement in translation quality over the baseline for all translation pairs. Table 7.10 shows that the baseline *OBT* system falls short of our proposed method by between 4.7 BLEU points (EN→KK) and 10.7 BLEU points (UK→EN) on the general test set. The differences on the legal test set are even more pronounced: we observe an increase of up to 14.5 BLEU over the baseline (EN→UK). Our DE→HSB system outperforms the WMT22 winner by 17 BLEU points. When translating from English to Kazakh, our approach reaches a BLEU score of 16.3 while the baseline which solely relies on iterative back-translation does

[9] `https://platform.openai.com/docs/models/gpt-3-5`

	DE-HSB	HSB-DE	EN-KA	KA-EN	EN-KK	KK-EN	EN-UK	UK-EN
WMT22 best	17.9	18.0	-	-	-	-	-	-
ChatGPT	6.6	-	3.9	-	5.2	-	**25.8**	-
OBT (baseline)	29.6	36.3	3.6	5.2	0.8	1.0	8.4	12.9
PseudoPar	11.3	12.0	1.9	4.8	1.0	3.1	4.6	8.6
OBT+PseudoPar	32.9	36.3	6.8	12.7	5.9	11.3	12.2	20.8
$\mapsto$OBT	**35.0**	**39.6**	**7.7**	**14.0**	**7.2**	**12.1**	**15.7**	**23.7**

	DE-HSB	HSB-DE	EN-KA	KA-EN	EN-KK	KK-EN	EN-UK	UK-EN
de Gibert Bonet (2022)	-	-	12.0	-	6.4	-	20.8	-
OBT (baseline)	-	-	9.0	12.7	0.3	0.3	14.9	12.6
PseudoPar	-	-	2.1	6.8	8.0	11.6	14.6	13.1
OBT+PseudoPar	-	-	11.5	22.0	**16.3**	**18.6**	**29.3**	**21.7**
$\mapsto$OBT	-	-	**15.0**	**23.5**	9.3	12.7	27.5	**21.8**

Table 7.10: MT performance of our systems measured by BLEU scores on the general test set (top) and the legal test set (bottom). Compared to the WMT22 winner [Shapiro et al., 2022], ChatGPT, and the system trained by de Gibert Bonet et al. [2022].

not receive enough cross-lingual signal to start learning at all. The hybrid system by de Gibert Bonet et al. [2022] which uses additional translation information from an unsupervised phrase-based system falls behind with a BLEU score of 6.4.

The results of translation by ChatGPT from English or German into truly low-resource languages (HSB, ka, KK) are significantly worse than our results. After manually evaluating several translations with a zero BLEU score, we suspected that the automatic metric puts ChatGPT's fluent but less literal translations at a disadvantage. We calculated the COMET score which is better able to capture the meaning similarity between texts but this hypothesis was not confirmed. The COMET score ranks chatGPT outputs similarly as the BLEU score (Table A.1).

Nonetheless, the EN→UK translation by ChatGPT is better than all unsupervised MT systems according to all used metrics. It must be noted that the systems cannot be directly compared to ChatGPT since its training corpus is larger and might include parallel texts.

The detailed evaluation with additional metrics (COMET and chrF++) is available in Appendix A.1. The results are generally in line with the BLEU score and the combination of training on pseudo-parallel and back-translated data performs the best under all three evaluation metrics.

Training Schedules

Figure 7.5 shows training curves with validation BLEU scores of all our DE↔HSB systems. We see that the *OBT+PseudoPar* system trained simultaneously on back-translated and pseudo-parallel data without any special schedule outperforms the baseline for DE→HSB but not in the opposite direction. For HSB→DE, the baseline performance is surpassed as soon as we remove the pseudo-parallel corpus from the training.

We trained several DE-HSB models starting from *OBT+PseudoPar* after each completed epoch of 770k pseudo-parallel sentences. Upon examination of the training curves in Figure 7.5, we see an immediate increase in validation BLEU score of ∼0.9–4.9 BLEU points

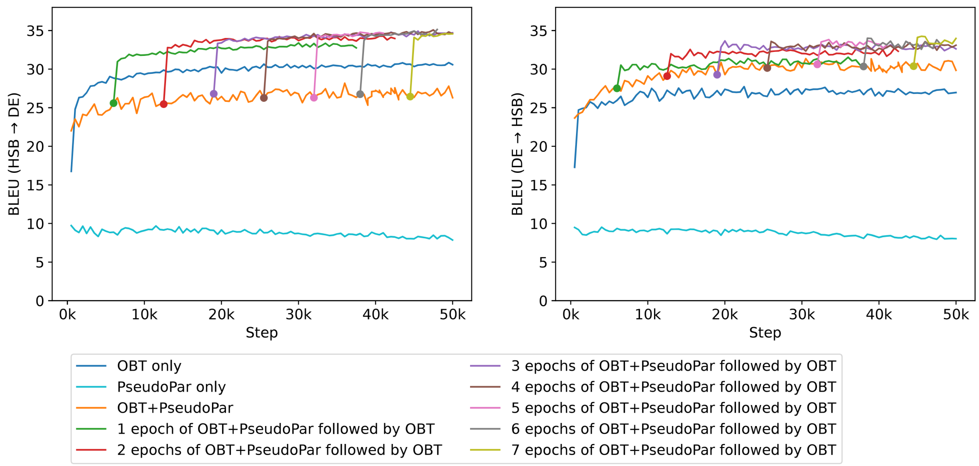


Figure 7.5: The development of validation BLEU scores during the training of HSB→DE (left) and DE→HSB (right) models. Any parallel resources were prohibited.

which occurred within the first 500 training steps after removing the pseudo-parallel corpus from the training. This observation confirms our hypothesis that pseudo-parallel sentence pairs aid the training in the beginning but the quality of the corpus itself poses an upper bound on the performance of the system. However, removing the corpus too early (after one or two epochs) leads to a lower final BLEU score. Therefore, we recommend to keep training the *OBT+PseudoPar* model until convergence and only then switch to iterative back-translation alone *OBT+PseudoPar↦OBT*. We note that the differences between *OBT+PseudoPar* and *OBT+PseudoPar↦OBT* are less pronounced when measured by the COMET score (Table A.1).

The flat *PseudoPar* training curves indicate that the quality of the pseudo-parallel corpus alone is inadequate for training a functional MT system without back-translation.

Domain-specific MT

Interestingly, removing the pseudo-parallel corpus from the training harms the translation quality measured on the legal test sets where the best performance for EN→KK, KK→EN and EN→UK is achieved by *OBT+PseudoPar*. We suspect that this is the result of the repeating terminology in the domain-specific test sets which is better handled by the *OBT+PseudoPar* for some language pairs. This is consistent with the fact that the *PseudoPar* system trained exclusively on pseudo-parallel data performs quite well on the EN-KK and EN-UK legal test set (8.0 on EN→KK, 11.6 on KK→EN and 14.6 on EN→UK) while having poor results on the general test set (1.0 on EN→KK, 3.1 on KK→EN and 4.6 on EN→UK). Based on our findings, we believe that utilizing pseudo-parallel sentences extracted from domain-specific monolingual corpora has the potential to enhance the training of domain-specific MT in general. However, further experiments are out of the scope of this thesis.

#	DE	HSB	Score
1	Thomas de Maizière	Thomas de Maizière	1.286
2	*Knut* ist tot.	*Bayer* ist tot.	1.245
3	Es ist ein harter Kampf, die Konkurrenz ist groß.	To bě napjata hra, a konkurenca bě wulka.	1.185
4	Der Roman hat *1200* Seiten.	*Kniha* ma *300* stronow.	1.178
5	Er passt zu diesem Team wie der Deckel auf den Topf.	Wón so k mustwu hodźi kaž wěko na hornc.	1.161
6	Die größte misst über *fünf Meter, die kleinste wenige Milli*meter.	Najkrótša měri *10 cm*, najdlěša *1 meter*.	1.101
7	Wer Wohlstand will, braucht Wissenschaft.	Štóž chce *něšto změnić*, *trjeba sylnu wolu*.	1.063
8	*Morgen ist doch auch noch ein Tag*!	*Ale to njeje hišće wšo*!	1.053
7	*Auch für Apple ist das iPhone wichtig.*	*Tež aleje su jara važne.*	1.037

Table 7.11: A sample from the DE-HSB mined parallel corpus. Non-matching words in italics.

Data quality

The sentence pairs in the pseudo-parallel corpus are far from equivalent in meaning. As illustrated in Table 7.11, many of the sentences are paired because they share a named entity, a numeral (not necessarily identical), a punctuation mark, or one distinctive word. Others have a similar sentence structure, they contain a similar segment or they contain words that are somehow related, e.g. Apple/alleys (*"aleje"*), although the word Apple is not the fruit in this context. On the other hand, synthetic sentences in the first training iterations are also extremely noisy, and even later they contain artifacts such as non-translated words or mistranslated named entities.

Table 7.12 shows what the back-translated and pseudo-parallel data can look like. We observed how the back-translated version of one sentence changes as the training progresses and witnessed several types of error, e.g. the German word *"laufend"* is not translated at all in the initial iterations; the word "April" remains mistranslated as "March" ("*měrc*") throughout the entire training. On the other hand, the pseudo-parallel sentence matched based on its distance from the source sentence has a similar meaning but is factually inaccurate.

We see that the meaning of many of the pseudo-parallel sentence pairs significantly differs but it is difficult to measure the quality of the entire corpus. We measure it indirectly by the increase in BLEU score associated with introducing the corpus into the UNMT training or by measuring the quality of the sentence encoder used for creating the corpus. To be able to evaluate the precision/recall of the sentence encoder, we have to control the number of parallel sentences hidden in the input corpora. However, in real-life scenarios, the level of comparability of two monolingual corpora is difficult to estimate. If the monolingual corpora provided for unsupervised translation come from a different domain and contain dissimilar sentences, the model has no good candidates to find. This poses a challenge especially when setting the correct mining threshold for the monolingual corpora at hand.

It is not clear what are the attributes of the pseudo-parallel corpus that the UNMT train-

SRC	Ich musste mich laufend weiterbilden, und so legte ich im April 1952 die erste und ein Jahr darauf die zweite Lehramtsprüfung ab.
REF	Dyrbjach so běžnje dale kwalifikować, a tak złožich w aprylu 1952 prěnje a lěto po tym druhe wučerske pruwowanje.
PseudoPar	*Hańža Winarjec-Orsesowa* wotpołoži prěnje wučerske pruwowanje *w lěće 1949 a druhe w lěće* 1952.
OBT @ 500	Dyrbjach so *laufend* dale *kubłać*, a tak *legte w měrcu* 1952 *prěnje* a lěto na to druhe *Lejnjanske* pruwowanje *ab.*
OBT @ 3000	Dyrbjach so běžnje dale *kubłać*, a tak w *měrcu* 1952 prěnju a lěto na to druhu *lektoratu serbšćiny wotpołožichmy.*
OBT @ 10000	Dyrbjach so běžnje dale *kubłać*, a tak wotpołožich w *měrcu* 1952 prěnju a lěto na to druhu *lektoratu.*

Table 7.12: A sample sentence translated by the OBT model after 500, 3,000 and 10,000 training steps compared to the closest neighbor of such sentence from the bilingual sentence space (*PseudoPar*). The mistranslated words are indicated in italics.

ing benefits from the most. We believe that the benefits of training on such noisy data are twofold: 1) the perfect matches are a valuable source of correct supervision, and 2) the abundant less-than-perfect matches still introduce a new translation signal which can help the model leave a suboptimal situation which we often observe during back-translation when the model learns to mistranslate a word and never forgets it. An example of error pattern induced by back-translation can be seen in Table 7.12 where the model in different stages of the training consistently mistranslates the word *"weiterbilden"* as *"kubłać"* ("to pour") when the meaning is "to further educate oneself". On the other hand, the word *"laufend"* was first mistranslated but later fixed and at 3k training steps it was correctly translated as *"běžnje"*.

7.4.4 Takeaways

We have demonstrated the benefits of MT training on pseudo-parallel data in situations when true parallel data is not available. While the pseudo-parallel corpus alone does not reach sufficient quality for standard supervised MT training, it works well in combination with online back-translation. We found it optimal to train the model until convergence on both pseudo-parallel and synthetic sentence pairs, then remove the pseudo-parallel corpus and continue training with iterative back-translation only.

We confirm our hypothesis that UNMT models are not able to fully exploit the cross-lingual knowledge present in monolingual data. If we match similar sentences prior to the training using an external tool and present the model with the matched pairs, translation quality improves.

Incorporating similar sentence pairs into the standard UNMT training increases translation quality across all evaluated language pairs with an improvement of up to 14.5 BLEU over the baseline trained without pseudo-parallel data and 8.5 BLEU over a hybrid unsupervised system (EN→UK). Furthermore, we observed that in some situations (EN↔KK), the online back-translation became trapped in a suboptimal state where no learning occurs. Introducing pseudo-parallel data can rescue the model from this state and restart the learning process.

After evaluating our approach on a test set in the legal domain, we believe that training on pseudo-parallel sentences could be particularly useful for domain-specific unsupervised MT. If we have two in-domain monolingual corpora at hand, parallel corpus mining is an efficient strategy to retrieve translation information.

The pseudo-parallel corpus helps the training despite being noisy. We hypothesize that while exact translations help the model find correct correspondences, also the noise can introduce new information and prevent the model from memorizing some of the artifacts of back-translated sentences. We leave it up to future research to evaluate whether a cleaner but smaller corpus would bring even larger gains.

In the following section, we stress-test this approach in the conditions of truly low-resource languages where monolingual corpora have a limited size and cover different domains.

7.5 Limitations of Unsupervised MT

In the previous sections, we established that if parallel texts are not available, MT models can learn using unsupervised techniques from monolingual data only. We tested on four language pairs exhibiting rich linguistic variety, out of which DE-HSB, EN-KA and EN-KK are considered low-resource according to the definition given in Chapter 2.

While the results are promising, the absolute BLEU scores for the more remote language pairs are still fairly low. It has been argued [Marchisio et al., 2020], that unsupervised techniques fail when

- languages are linguistically dissimilar;
- or there is a domain mismatch between the training corpora;
- or there is not enough monolingual sentences (less than 1M) for training.

In the previous section, we showed that using pseudo-parallel data for training in combination with the right pre-training strategy, we can train functional UMT systems even in the scenarios above. In particular, Georgian and Kazakh are linguistically far from English, and the Upper Sorbian training corpus is only 0.9M sentences.

Here we perform several experiments in even more adverse conditions and train MT models for translation between English and four low-resource Indic languages: Assamese (AS), Khasi (KHA), Mizo (MZ), and Manipuri (MNI). All of these languages are linguistically dissimilar from English, the amount of monolingual data is limited (only 183k sentences in Khasi), and the corpora exhibit a domain mismatch. We employ our approach of training on pseudo-parallel corpora to determine whether it can help in situations where other unsupervised techniques fail. The experiments from this section were carried out as part of the Indic MT shared task[10] of WMT23 and the system description will be published in the workshop proceedings.

[10] http://www2.statmt.org/wmt23/indic-mt-task.html

	AS	KHA	MNI	MZ	EN
train (mono)	2.6M	183k	2.1M	1.9M	33M
train (para)	50k	24k	50k	20k	-
train (pseudo-para)	32k	5k	95k	66k	-
dev	2k	1k	1k	1.5k	-
test	2k	1k	1k	2k	-

Table 7.13: The number of sentences in the training, dev and test sets used in the Indic MT shared task.

7.5.1 Data

We use the data provided for the WMT23 shared task of Indic MT. The Indic training data cover a combination of the news domain and the religious domain. In addition to the provided data, participants were allowed to use any monolingual texts and any pre-trained models trained on monolingual texts. We used 33M English sentences from NewsCrawl2022 and relied on the pre-trained model from Chapter 5 for parallel corpus mining. The summary of the data is in Table 7.13.

We trained a BPE model on the concatenation of all Indic corpora and a downsampled Englih corpus. The BPE vocabulary size is 52k. During pre-processing, we first tokenized the texts using the Moses tokenizer which created a problem with the Bengali-Assamese script as it decomposed several compound Unicode characters which had an impact on the segmentation of texts using this script (AS, MNI). The decomposed accents form a separate BPE unit which lead to a high segmentation of the Assamese and Manipuri texts. During post-processing we managed to compose the segmented text by running a reverse substitution on top of the standard detokenization. The unnecessary step of Moses tokenization likely cost us some final translation performance due to the suboptimal BPE segmentation.

We obtain our pseudo-parallel data using two version of the Indic sentence encoder we described in Section 5.4.4. The *XLM-100 (Indic)* model was fine-tuned on monolingual data in EN, AS, KHA, MNI, MZ. The *XLM-100 (Indic+EN-DE synth)* model was further fine-tuned using EN-DE synthetic parallel data. In Table 7.14, we report the performance of the two encoders on the task of parallel corpus mining where the model is evaluated on finding parallel sentences in two corpora of 202k sentences built by mixing the development set of 2k parallel sentences into a random set of 200k monolingual sentences from the training corpus.

7.5.2 Model & Training

We pre-train all our models using the most successful pre-training strategy from Section 7.3 which is MLM followed by DAE. During translation training, we use the combination of OBT and MT on a mined parallel corpus (*PseudoPar*) as described in Section 7.4.

Training Details

The training configuration is identical to Section 7.3.

	XLM-100 (Indic)				XLM-100 (Indic+EN-DE synth)			
	EN-AS	**EN-KHA**	**EN-MNI**	**EN-MZ**	**EN-AS**	**EN-KHA**	**EN-MNI**	**EN-MZ**
Precision	35.03	9.67	7.92	22.54	58.08	28.64	13.49	38.56
Recall	18.55	10.50	5.70	18.00	39.70	23.60	12.10	33.80
F1 Score	24.26	10.07	6.63	20.01	47.16	25.88	12.76	36.02
Threshold	1.023	1.025	1.022	1.022	1.022	1.027	1.022	1.022

Table 7.14: Precision, recall and F1 score of the *XLM-100 (Indic)* and *XLM-100 (Indic + EN-DE synth)* models on the parallel corpus mining task. The thresholds were optimized for the highest F1 score and used for mining training sentences for our MT models.

System	Sentence Encoder	EN-AS	AS-EN	EN-MNI	MNI-EN
OBT (baseline)	-	0.2	0.3	0.1	0.1
OBT+PseudoPar	XLM-100 (Indic)	1.0	**1.4**	0.2	0.3
OBT+PseudoPar	XLM-100 (Indic+EN-DE synth)	**1.4**	**1.5**	**2.8**	0.7

System	Sentence Encoder	EN-MZ	MZ-EN	EN-KHA	KHA-EN
OBT (baseline)	-	2.0	0.8	7.7	2.3
OBT+PseudoPar	XLM-100 (Indic)	4.1	**2.3**	7.4	2.0
OBT+PseudoPar	XLM-100 (Indic+EN-DE synth)	**4.8**	**2.5**	**12.6**	**4.6**

Table 7.15: BLEU score of Indic unsupervised MT systems on the WMT23 test set. COMET and chrF++ results are reported in the Appendix.

7.5.3 Results & Discussion

The unsupervised results are reported in Table 7.15. We observe that the BLEU scores for EN-AS and EN-MNI are less than 1 BLEU using the baseline unsupervised approach, meaning that the models learn almost zero translation knowledge. The performance can be significantly improved by adding noisy pseudo-parallel sentences, but BLEU still remains below 3 points. Upon closer analysis of the best translation candidates, we see that such low scores correspond to an average of 2 word matches per reference-candidate sentence pair. We review the translations and observe that the models generate fluent sentences within the same topic as the source sentence but their meaning is completely off. This finding points in the direction that unsupervised techniques could be useful for domain adaptation or style transfer even in high resource languages.

There are several possible explanations for such subpar results. Both AS and MNI share a non-Latin script. We experienced problems with the Moses tokenization where words containing compound Unicode characters were often incorrectly split or even segmented at the character level. The amount of monolingual data ($\sim$2M) is lower than we had in our previous experiments. Both languages are linguistically distant from English (which, however, also applies to KA and KK where the unsupervised methods work). And finally, Indic texts contain segments from religious texts whereas English training data is from the news domain.

The results for EN-KHA and EN-MZ are slightly more promising. The effect of training on pseudo-parallel sentences is significant for both language pairs and amounts to $\sim$5 BLEU points. However, we see that the models quickly converge to these values, marking a distinct training trajectory compared to what we witnessed in our experiments from Section 7.4. Moreover, we see very low results in the translation direction from the Indic languages into English which contrasts with our prior experiments where translating into English was less problematic than the reverse direction.

The impact of using the *PseudoPar* corpus for UMT training across evaluated language pairs does not fully correspond to the per-language performance of the sentence encoder reported in Table 7.14. On one hand, the mining precision is significantly higher for the improved encoder *XLM-100 (Indic+EN-DE synth)* and we observe a corresponding increase in translation quality when using pseudo-parallel sentences retrieved by this model. On the other hand, the strongest impact on translation quality is observed for EN-KHA where the encoder precision is only 29%. Moreover, the encoder precision for EN-AS is 58% but despite this high value, the unsupervised MT training fails to start. For comparison, the precision for EN-KA and EN-KK was 45% and 49% (Table 7.8), respectively, and the models were able to extract significant translation knowledge from the retrieved pseudo-parallel data. To have a clearer view of what the data looks like, we carried out a manual evaluation of EN-KK and EN-KA pseudo-parallel corpora (see Figure 8.1 in Discussion) and found that the structure of the two corpora is relatively similar. However, given the smaller AS monolingual corpus, the AS-EN pseudo-parallel corpus has only 33k sentence pairs. Moreover, the AS-EN data suffers a domain mismatch since the AS corpus contains a significant amount of religious texts. These challenges, together with the linguistic dissimilarity and the problematic Assamese script, might be the reasons why the model fails to start learning.

Surprisingly, despite the low amount of KHA training data (183k sentences), the KHA-EN MT system was able to reach a reasonable level of translation quality without seeing any authentic KHA-EN translations. We will see in the next section that the BLEU score is close to the semi-supervised result.

7.5.4 Takeaways

We confirm that in the situation of training data domain mismatch, linguistic dissimilarity, different scripts (AS, MNI) and limited amounts of monolingual data, unsupervised MT models struggle. Without *PseudoPar* data in the training mix, the majority of unsupervised models we experimented with did not even start learning. Upon the introduction of *PseudoPar* texts, the BLEU score increases but remains low.

7.6 Pseudo-Parallel Data in Semi-Supervised MT

In the following section, we will depart from the constraints posed by the unsupervised MT scenario and study low-resource translation between English and the four Indic languages introduced in the previous section with limited amounts of parallel data available. We will be incrementally adding parallel sentences into the unsupervised training and create semi-supervised systems to determine:

- how translation quality increases as we add more parallel sentences into the training;

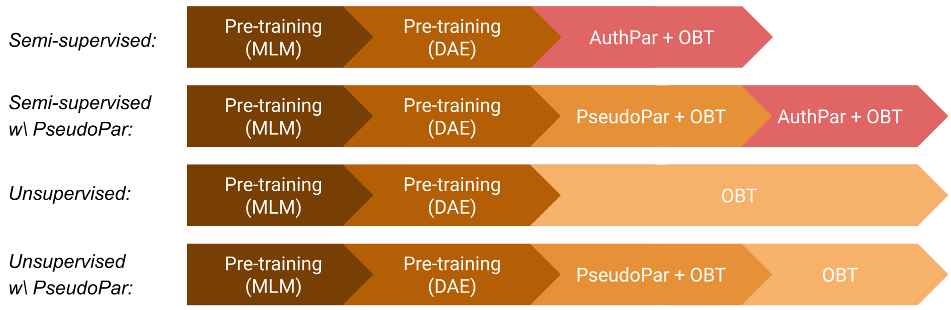


Figure 7.6: Schematic illustration of the training pipeline of our models. The size of the blocks is not proportional to training time.

- whether incorporating pseudo-parallel data into the training helps in semi-supervised scenarios;
- how many authentic parallel sentence pairs are required for the model to not see any further benefit in the noisy pseudo-parallel data.

In Section 7.4, we established that pseudo-parallel data play an important role in unsupervised training. In Section 7.5, we pointed at the limitations of unsupervised MT techniques in authentically low-resource scenarios. In the experiments presented in this section, we examine whether pseudo-parallel data can be useful in situations where small amounts of authentic parallel data are available.

7.6.1 Data

In addition to the data from Section 7.5 listed in Section 7.5, small amounts of parallel training data (*AuthPar*) provided for the WMT23 shared task was used (50k sentence pairs for EN-AS, 24k sentence pairs for en-KHA, 22k sentence pairs for EN-MNI and 50k sentence pairs for EN-MZ). Pseudo-parallel corpora (*PseudoPar*) used in our semi-supervised experiments are identical to those from Section 7.5. The number of retrieved pseudo-parallel sentence pairs is indicated in Table 7.13.

7.6.2 Model & Training

For our WMT23 submission to the shared task on Indic MT, we trained MT models in a semi-supervised manner using available parallel data as well as unsupervised techniques. We experiment with the same language pairs as in Section 7.5: English-Assamese (EN-AS), English-Manipuri (EN-MNI), English-Mizo (EN-MZ) and English-Khasi (EN-KHA).

This shared task was proposed as a realistic scenario where for each Indic language, the participants have access to several thousand parallel sentences paired with English, up to 2.6M additional unaligned sentences in each Indic language, and an unlimited amount of English texts. In addition, using any model pre-trained on monolingual texts was allowed.

	EN-AS	EN-KHA	EN-MNI	EN-MZ
AuthPar+OBT (semi-sup)	**14.1**	**16.6**	**29.5**	**31.2**
PseudoPar+AuthPar+OBT (semi-sup)	13.3	15.9	**29.8**	**30.8**
OBT (unsup)	0.2	7.7	0.1	2.0
OBT+PseudoPar$\mapsto$OBT (unsup)	1.4	12.6	2.8	4.8

Table 7.16: BLEU score of EN-AS, EN-KHA, EN-MNI and EN-MZ semi-supervised MT systems on the WMT23 test set.

Pre-Training on Monolingual Texts

All our systems are pre-trained on the MLM and DAE tasks as described in Section 7.3. A schematic illustration of the training pipeline is in Figure 7.6.

Semi-Supervised MT Training

In the semi-supervised setup, we fine-tune a bidirectional model for each language pair with the standard supervised MT objective (first on the pseudo-parallel corpus *PseudoPar* and then on the authentic parallel corpus *AuthPar*) as well as the OBT objective (on the monolingual corpus). We compare the results of the semi-supervised models to completely unsupervised models trained only with OBT and *PseudoPar* data to measure the effect of limited amounts of parallel texts. We experiment with gradually adding parallel data into the training and evaluate the performance of a model trained on 1k, 2k, 5k, 10k and 25k parallel sentences. Furthermore, we train models with and without the *PseudoPar* pre-training stage and we evaluate the impact of using pseudo-parallel data on the final translation quality as the amount of authentic parallel texts increases.

7.6.3 Results & Discussion

Shared Task Results

Regarding the semi-supervised shared task results, our EN→MNI system ranked second out of 14 participants. Our EN→MZ system ranked fourth out of 11 participants. The remaining systems finished on the 5th-7th places. The winning system for all language directions was a system called TRANSSION-MT which outperformed other systems with almost double the BLEU score of the second best candidate. Since the participants were allowed to use unlimited amounts of monolingual data in any languages, there might be great discrepancies between the amounts of monolingual data and auxiliary languages used by other participants. Furthermore, the participants were allowed to use any available models pre-trained on monolingual data which makes it difficult to guarantee that used models do not suffer from test set contamination. System descriptions of other participants were not available at the time of submitting this thesis.

Pseudo-Parallel Sentences in Semi-supervised Training

Outside of the scope of the shared task, we were interested in the following phenomena which we measured in our experiments:

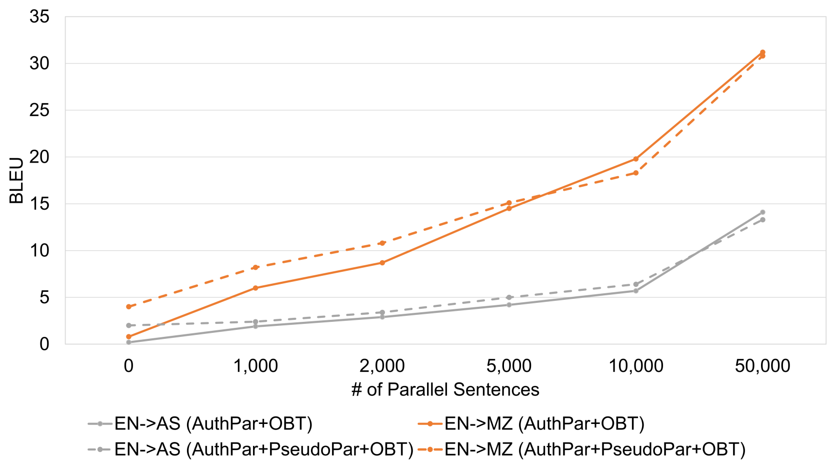


Figure 7.7: Relationship between the translation quality and the number of authentic parallel sentences used for training. Dashed lines represent systems trained on pseudo-parallel (*PseudoPar*) sentence pairs in addition to authentic (*AuthPar*) and back-translated (OBT) sentence pairs.

- the gap between unsupervised and semi-supervised translation systems;
- the impact of training with pseudo-parallel sentence pairs on the final translation quality;
- the development of translation quality in relation to the number of authentic parallel sentences used during training.

We trained unsupervised MT systems as described in Section 7.5. Table 7.16 shows that the unsupervised systems reach less than 5 BLEU which is not a sufficient quality for practical use. The large gap between the unsupervised and supervised systems is most likely the consequence of linguistic dissimilarity and the domain mismatch between English and Indic data. Our conclusions support the claims of other researchers [Marchisio et al., 2020, Vulić et al., 2019] that unsupervised MT models often fail in truly low-resource scenarios where it is not possible to get enough clean and domain-balanced monolingual training data.

Furthermore, Table 7.16 shows that data augmentation with pseudo-parallel sentences has zero or even a negative impact on the performance of our semi-supervised systems. For the unsupervised systems, on the other hand, it increases BLEU score by up to 3.6 BLEU points.

Our previous experiments showed that the pseudo-parallel data in EN-AS and EN-MZ have sufficient quality to aid translation training. Therefore, we trained several other systems, gradually adding authentic parallel sentences to measure the threshold where the positive impact of pseudo-parallel sentences disappears. Figure 7.7 illustrates the relationship between translation quality and the size of the authentic parallel corpus and reveals that when we have between 10k and 25k parallel-sentences, the unsupervised data augmentation technique of adding pseudo-parallel sentence pairs is not beneficial anymore.

7.6.4 Takeaways

We trained semi-supervised and unsupervised systems for translation between English and Indic languages and we conclude that the translation quality rises rapidly by adding small amounts of parallel data into the training. We use back-translated and pseudo-parallel sentences to prevent the model from over-fitting to the small authentic parallel corpus and reached favorable results. We showed that for translation from English into Assamese and Mizo, data augmentation with noisy pseudo-parallel data is beneficial when we have less than 10k authentic sentence pairs.

In situations where unsupervised techniques fail, adding a thousand authentic translations into the training can significantly improve the results. With 50k parallel sentences and online back-translation, the models reach a solid translation quality.

8. Discussion

We performed a number of experiments across several tasks and various language pairs. In this chapter, we summarize the observations we have made, and we list several challenges we have faced.

Observation 1: Sentence representations extracted from multilingual Transformer language models can be used for parallel corpus mining.

Although several authors [Feng et al., 2022, Reimers and Gurevych, 2020] claim that representations from Transformer language models cannot be used for sentence retrieval without fine-tuning with a sentence-level objective, we show that under certain conditions, averaging per-token representations suffices to produce meaningful sentence embeddings. We perform light fine-tuning of the pre-trained *XLM-100* model on a translation masked language modelling (TLM) task. Using this technique, we observe an improvement of up to 22 points in the F1 on score on a parallel corpus mining task. We use retrieved (pseudo-parallel) sentences for training an unsupervised MT system and report a significant boost in translation quality upon the introduction of the pseudo-parallel data into the training.

Utilizing sentence embeddings from newer models like LaBSE [Feng et al., 2022], distilled Sentence-BERT [Reimers and Gurevych, 2020], or distilled LASER [Heffernan et al., 2022], which leverage parallel data to enhance the alignment of cross-lingual representations for equivalent sentences, would yield improved results. However, adopting these models would require departing from the constraint of a fully unsupervised scenario. In this thesis, we explore the highest translation quality attainable by training on monolingual data only and we strive to move towards that theoretical limit. Therefore, using small amounts of parallel data is outside of the scope of this thesis (except our small experiment in Section 7.6). In practical applications involving low-resource languages, it would be advisable to use any parallel data available. It has been shown that several thousand parallel sentences suffice to distill the knowledge of a heavily supervised model (e.g. LASER or MuSE [Yang et al., 2020]) into a new model which specializes in a low-resource language [Costa-jussà et al., 2022].

Observation 2: The benefits of light fine-tuning of the XLM model extend to unrelated language pairs.

In Chapter 5, we showed that fine-tuning the *XLM-100* model with a TLM objective improves its sentence retrieval capability regardless of the languages used during fine-tuning. For instance, fine-tuning on Czech-German synthetic sentence pairs with masked tokens improves the results on all evaluated language pairs (e.g. English-Afrikaans, English-Kazakh, English-Georgian). Similarly, fine-tuning the *XLM-100 (Indic)* model on either Czech-German or English-German sentence pairs further identically improves the results for the Indic language pairs.

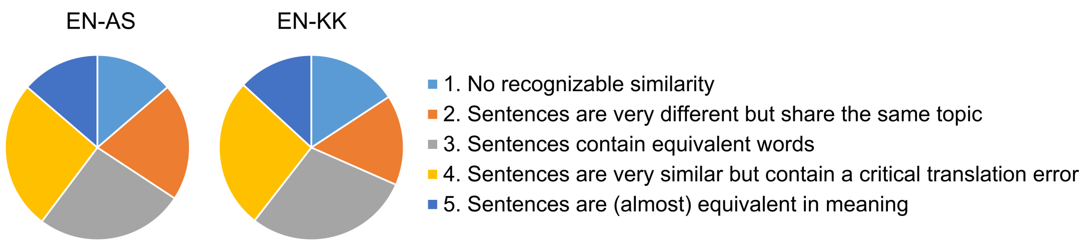


Figure 8.1: Manual evaluation of 100 sentence from English-Kazakh and English-Assamese pseudo-parallel corpora. The evaluation was carried out in English based on the translations from Google Translate.

The reasons for such a cross-lingual, or even cross-task, improvement are not quite clear and definitely deserve future exploration.

Observation 3: Unsupervised MT models benefit from training on noisy pseudo-parallel sentences.

We have shown throughout this thesis that pseudo-parallel sentences aid unsupervised MT training despite being noisy. In order to better assess how noisy the data is, we carried out a manual evaluation on a sample of 100 sentences from the English-Kazakh and English-Assamese parallel corpora. The evaluators were asked to assign a category to each pseudo-parallel sentence pair to assess its similarity.

The results in Figure 8.1 show that only a small fraction of sentence pairs are (almost) perfect translation equivalents. This is also the consequence of the fact that the monolingual corpora of limited size rarely include sentences which are fully equivalent, especially the longer ones. Many sentences are labeled as "very similar with a critical translation error" where the two sentences are virtually identical, but they include a different name or number, which is critical as far as translation quality is concerned. A majority of sentence pairs was matched because they include several equivalent words. A small portion of sentences was matched solely based on their sentence structure (e.g. the same punctuation or sentence length) with no semantic similarity.

Seeing the quality of the data and the low number of equivalent sentence pairs, it might seem unexpected, but all our findings indicate that training on such noisy parallel data is still preferable to using no parallel data at all. However, in severely adverse conditions of linguistic dissimilarity, technical problems with correct script processing, and domain mismatch in the training data (English-Assamese, English-Manipuri), we were not able to make unsupervised MT work even with the help of pseudo-parallel data.

Surprisingly, the structure of the two monolingual corpora in Figure 8.1 is very similar in terms of the quality we evaluated. We saw in Section 7.4 and Section 7.5 that while the English-Kazakh corpus significantly helps the training, English-Assamese UNMT systems quickly converge to a low score. An important factor here is the size of monolingual corpora. Out of 2.6M Assamese sentences and 32M English sentences, we were only able to mine 33k pseudo/parallel sentences. In the case of English-Kazakh translation, we found 169k translation pairs out of 8M Kazakh sentences and 17M English sentences. There's a possibility that employing large-scale mining in an English corpus ten times the current size could yield improved results for English-Assamese translation as well.

Observation 4: Unsupervised MT systems struggle with named entities.

Translating names, especially proper nouns, is always a challenge as they might not have direct equivalents in the target language. They can be culturally specific or unique, making it challenging for the system to find suitable translations without context.

In unsupervised MT, the problem is much more severe. Even the MT systems that reach high BLEU scores very often mistranslate names and numbers, and this deficiency significantly hampers their practical use. This problem was discussed in more detail in Section 7.2. The reason is that the vector representations of names and numbers often lie close to each other in the embedding space, as illustrated in Figure 8.2 Since the initial translation signal for both UNMT and UPBMT systems comes from such shared latent space, the problem is introduced already in the beginning, and subsequent training by back-translation, unfortunately, has no way to block such mistranslations and thus effectively ensures that the problem persists. The introduction of pseudo-parallel data into the training can partially alleviate the problem but also introduces new mistranslations of named entities which were present in the pseudo-parallel corpus. We saw in Section 7.4 that sentences in the pseudo-parallel corpus were often matched because they included a name or a number, but not necessarily an equivalent one, or because they matched in most of the message *except for* a name or a number.

Challenge 1: Data quality

Data cleaning is a challenge in truly low-resource conditions, as we cannot rely on common solutions and tools that we take for granted for high resource languages. We faced this when processing the Mizo monolingual corpus which was infested with a great number of sentences in other languages. In normal conditions, we would have used a language tagger to clean the data but none of the common pre-trained language taggers (fasttext-lid, langdetect, langid, cld2) supports the Mizo language. We realized the extent of the problem only when searching for equivalent sentences in the Mizo and English corpora and finding a great number of English sentences which were hidden in the Mizo corpus. Re-training the model with a cleaned corpus would most likely increase the translation quality, especially when using mined-parallel sentences for training. Many of the mined sentence pairs were identical English sentences, others were different sentences which were matched based on the identical English words they included. Both of these likely harmed the training, teaching the model to copy English words from the source to the target.

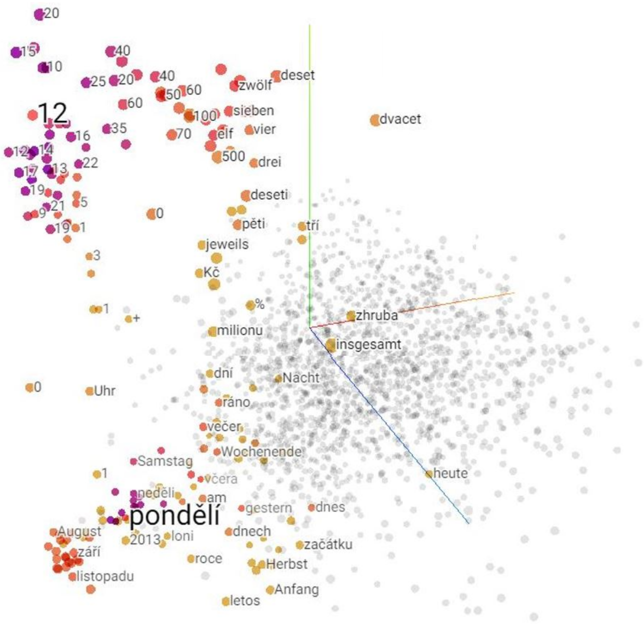


Figure 8.2: PCA visualisation of Czech and German cross-lingual word embedding spaces aligned as described in Chapter 6. We illustrate the nearest neighbors of the words "12" and "pondělí" (*"Monday"*) and see that different numbers and temporal words (e.g. *today, yesterday, autumn, August, Saturday* etc.) cluster together.

Source: Kvapilíková [2020]

Challenge 2: Working with less common scripts

Low-resource languages, especially those that have received limited attention in terms of linguistic resources and technological development, often use different character sets or scripts compared to high-resource languages. Many of the languages we experimented with (Assamese, Manipuri, Georgian, Inuktitut) use a non-Latin and non-Cyrillic alphabets. Handling different character sets can be a challenge. We faced it during text pre-processing of languages using the Bengali-Assamese script (Assamese and Manipuri) where tokenization and subword segmentation lead to a decomposition of several Unicode characters. It resulted in the isolation of accents into separate characters and too granular segmentation. We only noticed this after the end of the training and applied a reverse operation to reconstruct the texts. Since unsupervised training relies on the geometrical properties of embedding spaces, the suboptimal segmentation could have significantly harmed the performance.

Moreover, new alphabets can be challenging for pre-trained models. Fortunately, the *XLM-100* model we worked with had individual characters of these alphabets in its vocabulary but the texts were split at the character-level. This may hinder the cross-lingual transfer within the model that we rely on during fine-tuning. Moreover, many sentences exceeded the maximum number of tokens allowed per model input due to the excessive granularity of segmentation.

Challenge 3: Domain mismatch in training corpora

Unsupervised MT is based on the underlying idea that the concepts described by a language are grounded in the real world, regardless of the language we use. While this assumption might be true in general, it is not applicable in situations when the texts we have available for each language exhibit a domain mismatch. We cannot assume that texts from movie subtitles or sports news describe the same word as the Bible. We faced precisely this issue when creating our unsupervised systems in Indic languages. In low-resource scenarios, the problem is exacerbated by the fact that we cannot use off-the-shelf tools for domain classification and we do not have training data to create such tools on our own. Moreover, for languages in different scripts with little English influence, we cannot even roughly check what kind of data we are dealing with and we cannot use any commercial MT system to gain an understanding prior to our own training (e.g. Khasi and Mizo are not supported by neither Google Translate nor ChatGPT; Manipuri is supported by Google Translate with very poor results). On the other hand, this shows the importance of MT research for these languages which are completely excluded from existing NLP technologies.

Challenge 4: Lack of language experts and annotators

Obtaining access to language experts and annotators for low-resource languages can be challenging. These languages often have smaller speaker populations, limited digital presence, and fewer resources dedicated to linguistic research or technological development. As a result, finding individuals proficient in these languages for tasks like annotation, translation, or linguistic analysis can be more difficult compared to high-resource languages. This scarcity of experts and annotators can significantly impact the progress of language-related projects for these languages. We intended to conduct a manual evaluation of the translation output and pseudo-parallel corpora, but we did not reach enough speakers of Khasi, Mizo, and Manipuri to proceed with the plan.

Conclusion

Our research aim was to determine the most effective way of exploiting cross-lingual signal from monolingual data. We conclude that the most effective approach does not lie in determining the single best strategy but rather using a combination of methods. Unsupervised MT comprises a set of techniques that rely on monolingual texts and we contribute by extending this set with a modified pre-training strategy and a novel fully unsupervised way of training data creation. In Chapter 4, we introduced a taxonomy of unsupervised approaches and now we can place our methods on the map. We focused on both model-centric and data-centric approaches as we investigated the role of pre-training and model initialization (model-centric) and we experimented with different automatic methods of obtaining parallel data and using them for MT training (data-centric).

Unsupervised MT models relying only on model pre-training and back-translation often fail in truly low-resource conditions. We showed that they are not able to fully exploit the translation signal present in monolingual data and they benefit from explicit supervision extracted from the same data using an external model. We proposed a training strategy where we included pseudo-parallel data mined from monolingual corpora in unsupervised MT training and reached a significant improvement across all evaluated language pairs. Although pseudo-parallel texts obtained in a completely unsupervised way are very noisy with a majority of sentence pairs being similar rather than equivalent, they offer the model a source of external translation knowledge that complements the training on synthetic back-translated examples.

An alternative way of introducing a different source of translation signal to unsupervised neural MT models is by training on synthetic parallel sentences generated by phrase-based models. We showed that training on a combination of synthetic sentences produced by different types of MT systems is superior to training only on back-translated sentences generated by the neural model during training.

In this thesis, we created two kinds of unsupervised models: (1) unsupervised MT systems which create their own cross-lingual representations and use them for generating translations, and (2) multilingual sentence encoders which are capable of selecting equivalent or similar sentences from a pool of monolingual sentences. We showed that the two kinds of models can benefit from each other: unsupervised MT systems trained on pseudo-parallel data improve in translation quality, and multilingual encoders fine-tuned on synthetic parallel data improve their translation matching accuracy.

For the practical applications of low-resource MT translation, we see the highest potential in large-scale parallel corpus mining and subsequent MT training on mined parallel corpora. If we relax the strict requirement of no parallel data, it is possible to employ multilingual sentence encoders trained on large parallel corpora in high-resource languages. Using very small amounts of parallel texts coupled with English then suffices for knowledge distillation to new languages. If not already available, collecting such small data could be the most effective way to increase MT quality for a particular low-resource language. Furthermore, unsupervised pre-training (e.g. masked-language modelling, denoising autoencoding) or transfer learning from related language pairs are effective methods to increase translation quality of low-resource MT.

In the beginning of this thesis, we asked what is the theoretical limit of translation based on monolingual texts. While we cannot answer this question beyond the methods we had experimented with, we believe the limit lies in the size and the domain overlap of monolingual data available. In high-resource conditions with large amounts of monolingual data, domain-balanced corpora, and ideally also linguistic similarity, the performance gap between models trained in a supervised and unsupervised way is narrow. We witnessed this when training our Czech-German MT systems. However, in such situations, unsupervised techniques are effectively not necessary because parallel resources typically exist, too.

When experimenting with translation between German and Upper Sorbian, a truly low resource language pair, the gap between semi-supervised approaches relying on limited amounts of parallel data and unsupervised approaches was wider. However, we were able to significantly reduce it by using our modified pre-training strategy and pseudo-parallel data. Similar results were reached when translating between English and Kazakh, Georgian and Ukrainian using monolingual data only.

Several authors pointed out the limitations of unsupervised approaches rooted in the underlying assumptions of unsupervised MT. Namely, if the representation spaces of two languages do not exhibit a sufficient level of isomorphism, unsupervised translation between them is not possible. While our method of training on pseudo-parallel data helped in situations where the baseline unsupervised approach failed, the limitation of our research remains the fact that in adverse conditions which are often present in truly low-resource scenarios, the translation quality is inadequate. We experienced this when training models for translation between English and four Indic languages: Assamese, Khasi, Manipuri and Mizo.

We see two possible directions of future research in continuation to this work. First of all, exploring the representations hidden in pre-trained multilingual models and improving their cross-lingual alignment is a very relevant topic especially in the era of large language models. We showed a simple fine-tuning strategy which makes the representations more language-agnostic but the source of that improvement deserves more investigation. Secondly, we believe that the techniques from unsupervised MT are applicable in high-resource scenarios where they can serve for domain adaptation or style transfer. Exploring how to effectively use them for that purpose constitutes a very interesting research avenue.

Bibliography

Vesa Akerman, David Baines, Damien Daspit, Ulf Hermjakob, Taeho Jang, Colin Leong, Michael Martin, Joel Mathew, Jonathan Robie, and Marcus Schwarting. The ebible corpus: Data and model benchmarks for bible translation for low-resource languages, 2023.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A simple but tough-to-beat baseline for sentence embeddings. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 1 2017. URL `https://openreview.net/pdf?id=SyK00v5xx`.

Mikel Artetxe and Holger Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610, Mar 2019a. ISSN 2307-387X. doi: 10.1162/tacl_a_00288. URL `http://dx.doi.org/10.1162/tacl_a_00288`.

Mikel Artetxe and Holger Schwenk. Margin-based parallel corpus mining with multilingual sentence embeddings. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019b. doi: 10.18653/v1/p19-1309. URL `http://dx.doi.org/10.18653/v1/p19-1309`.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1042. URL `https://aclanthology.org/P17-1042`.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 789–798, Melbourne, Australia, July 2018a. Association for Computational Linguistics. doi: 10.18653/v1/P18-1073. URL `https://aclanthology.org/P18-1073`.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Unsupervised statistical machine translation. In *Proceedings of the 2018 Conference on EMNLP*, Brussels, November 2018b. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/D18-1399.pdf`.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 789–798, Melbourne, 2018c. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P18-1073.pdf`.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. Unsupervised neural machine translation. In *Proceedings of the Sixth International Conference on Learning Representations*, April 2018d. URL https://arxiv.org/pdf/1710.11041.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. An effective approach to unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 194–203, Florence, July 2019a. Association for Computational Linguistics. doi: 10.18653/v1/P19-1019. URL https://www.aclweb.org/anthology/P19-1019.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. Bilingual lexicon induction through unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5002–5007, Florence, Italy, July 2019b. Association for Computational Linguistics. doi: 10.18653/v1/P19-1494. URL https://aclanthology.org/P19-1494.

Mikel Artetxe, Gorka Labaka, Noe Casas, and Eneko Agirre. Do all roads lead to rome? understanding the role of initialization in iterative back-translation. *Knowledge-Based Systems*, 206:106401, 2020. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2020.106401. URL https://www.sciencedirect.com/science/article/pii/S0950705120305335.

Mikel Artetxe, Itziar Aldabe, Rodrigo Agerri, Olatz Perez-de Viñaspre, and Aitor Soroa. Does corpus quality really matter for low-resource languages? In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7383–7390, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.emnlp-main.499.

Andoni Azpeitia, Thierry Etchegoyhen, and Eva Martínez Garcia. Extracting parallel sentences from comparable corpora with stacc variants. In Reinhard Rapp, Pierre Zweigenbaum, and Serge Sharoff, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France, may 2018. European Language Resources Association (ELRA). ISBN 979-10-95546-07-8. URL http://lrec-conf.org/workshops/lrec2018/W8/pdf/6_W8.pdf.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1409.0473.

Christos Baziotis, Ivan Titov, Alexandra Birch, and Barry Haddow. Exploring unsupervised pretraining objectives for machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2956–2971, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.261. URL https://aclanthology.org/2021.findings-acl.261.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, March 2003. ISSN 1532-4435. URL http://dl.acm.org/citation.cfm?id=944919.944966.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146, 2017. doi: 10.1162/tacl_a_00051. URL https://www.aclweb.org/anthology/Q17-1010.

Houda Bouamor and Hassan Sajjad. H2@bucc18: Parallel sentence extraction from comparable corpora using multilingual sentence embeddings. In Reinhard Rapp, Pierre Zweigenbaum, and Serge Sharoff, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France, may 2018. European Language Resources Association (ELRA). ISBN 979-10-95546-07-8. URL http://alt.qcri.org/~hsajjad/publications/papers/2018_BUCC_Bouamor_Comparable_Corpora.pdf.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf.

Hailong Cao, Tiejun Zhao, Weixuan Wang, and Wei Peng. Bilingual word embedding fusion for robust unsupervised bilingual lexicon induction. *Information Fusion*, 97:101818, 2023. ISSN 1566-2535. doi: https://doi.org/10.1016/j.inffus.2023.101818. URL https://www.sciencedirect.com/science/article/pii/S1566253523001343.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-2029. URL https://www.aclweb.org/anthology/D18-2029.

M. Cettolo, M. Federico, N. Bertoldi, R. Cattoni, and B. Chen. A look inside the itc-irst smt system. In *Proceedings of the 10th Machine Translation Summit*, pages 451–457, 2005.

Shweta Chauhan, Shefali Saxena, and Philemon Daniel. Improved unsupervised neural machine translation with semantically weighted back translation for morphologically rich and low resource languages. *Neural Process. Lett.*, 54(3):1707–1726, jun 2022. ISSN 1370-4621. doi: 10.1007/s11063-021-10702-8. URL https://doi.org/10.1007/s11063-021-10702-8.

Xilun Chen and Claire Cardie. Unsupervised multilingual word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 261–270, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1024. URL https://aclanthology.org/D18-1024.

Muthu Chidambaram, Yinfei Yang, Daniel Cer, Steve Yuan, Yunhsuan Sung, Brian Strope, and Ray Kurzweil. Learning cross-lingual sentence representations via a multi-task dual-encoder model. *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, 2019. doi: 10.18653/v1/w19-4330. URL http://dx.doi.org/10.18653/v1/w19-4330.

Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. Improving the lexical ability of pretrained language models for unsupervised neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 173–180, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.16. URL https://aclanthology.org/2021.naacl-main.16.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014. URL http://arxiv.org/abs/1412.3555.

Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 7059–7069. Curran Associates, Inc., 2019. URL http://papers.nips.cc/paper/8928-cross-lingual-language-model-pretraining.pdf.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1070. URL https://www.aclweb.org/anthology/D17-1070.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. In *6th International Conference on Learning Representations, ICLR 2018*, 2018a. URL https://arxiv.org/pdf/1710.04087.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. Word translation without parallel data. In *International Conference on Learning Representations*, 2018b. URL https://openreview.net/forum?id=H196sainb.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale, 2019.

Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. No language left behind: Scaling human-centered machine translation, 2022.

Ona de Gibert Bonet, Iakes Goenaga, Jordi Armengol-Estapé, Olatz Perez-de Viñaspre, Carla Parra Escartín, Marina Sanchez, Mārcis Pinnis, Gorka Labaka, and Maite Melero. Unsupervised machine translation in real-world scenarios. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 3038–3047, Marseille, France, June 2022. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.325.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *arXiv [e-Print archive]*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2013. URL https://www.aclweb.org/anthology/N13-1073.

David M. Eberhard, Gary F. Simons, and Charles D. Fennig, editors. *Ethnologue: Languages of the World*. SIL International, Dallas, Texas, 26th edition, 2023. URL http://www.ethnologue.com.

Cristina España Bonet, Adam Csaba Varga, Alberto Barron-Cedeno, and Josef van Genabith. An empirical analysis of nmt-derived interlingual embeddings and their use in parallel sentence identification. *IEEE Journal of Selected Topics in Signal Processing*, 11 (8):1340–1350, Dec 2017. ISSN 1941-0484. doi: 10.1109/jstsp.2017.2764273. URL http://dx.doi.org/10.1109/JSTSP.2017.2764273.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. Language-agnostic BERT sentence embedding. In Smaranda Muresan, Preslav Nakov, and Aline

Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.62. URL https://aclanthology.org/2022.acl-long.62.

Philip Gage. A new algorithm for data compression. *The C Users Journal archive*, 12:23–38, 1994. URL https://api.semanticscholar.org/CorpusID:59804030.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Advances in Computer Vision and Pattern Recognition*, page 189–209, 2017. ISSN 2191-6594. doi: 10.1007/978-3-319-58347-1_10. URL http://dx.doi.org/10.1007/978-3-319-58347-1_10.

Xavier Garcia, Pierre Foret, Thibault Sellam, and Ankur P. Parikh. A multilingual view of unsupervised machine translation, 2020.

Xavier Garcia, Aditya Siddhant, Orhan Firat, and Ankur Parikh. Harnessing multilinguality in unsupervised machine translation for rare languages. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1126–1137, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.89. URL https://aclanthology.org/2021.naacl-main.89.

Yvette Graham, Timothy Baldwin, and Nitika Mathur. Accurate evaluation of segment-level machine translation metrics. In Rada Mihalcea, Joyce Chai, and Anoop Sarkar, editors, *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1183–1191, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1124. URL https://aclanthology.org/N15-1124.

Mandy Guo, Qinlan Shen, Yinfei Yang, Heming Ge, Daniel Cer, Gustavo Hernandez Abrego, Keith Stevens, Noah Constant, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. Effective parallel corpus mining using bilingual sentence embeddings. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 165–176, Brussels, Belgium, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6317. URL https://www.aclweb.org/anthology/W18-6317.

Viktor Hangya and Alexander Fraser. Unsupervised parallel sentence extraction with parallel segment detection helps machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1224–1234, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1118. URL https://aclanthology.org/P19-1118.

Zellig Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954. doi: 10.1007/978-94-009-8467-7_1. URL https://link.springer.com/chapter/10.1007/978-94-009-8467-7_1.

Mareike Hartmann, Yova Kementchedjhieva, and Anders Sø gaard. Comparing unsupervised word translation methods step by step. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/file/d15426b9c324676610fbb01360473ed8-Paper.pdf.

Zhiwei He, Xing Wang, Rui Wang, Shuming Shi, and Zhaopeng Tu. Bridging the data gap between training and inference for unsupervised neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6611–6623, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.456. URL https://aclanthology.org/2022.acl-long.456.

Kenneth Heafield. KenLM: Faster and smaller language model queries. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, 2011. URL https://www.aclweb.org/anthology/W11-2123.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. Scalable modified Kneser-Ney language model estimation. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 690–696, Sofia, August 2013. URL https://kheafield.com/papers/edinburgh/estimate_paper.pdf.

Kevin Heffernan, Onur Çelebi, and Holger Schwenk. Bitext mining using distilled sentence representations for low-resource languages, 2022. URL https://arxiv.org/abs/2205.12654.

Dan Hendrycks and Kevin Gimpel. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415, 2017. URL https://arxiv.org/abs/1606.08415.

Geert Heyman, Bregt Verreet, Ivan Vulić, and Marie-Francine Moens. Learning unsupervised multilingual word embeddings with incremental multilingual hubs. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1890–1902, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1188. URL https://aclanthology.org/N19-1188.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9 (8):1735–1780, 1997.

John R. Hurley and Raymond B. Cattell. The procrustes program: Producing direct rotation to test a hypothesized factor structure. *Behavioral Science*, 7(2):258–262, 1962. doi: 10.1002/bs.3830070216. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/bs.3830070216.

Ayyoob ImaniGooghari, Peiqin Lin, Amir Hossein Kargaran, Silvia Severini, Masoud Jalili Sabet, Nora Kassner, Chunlan Ma, Helmut Schmid, André Martins, François Yvon, and Hinrich Schütze. Glot500: Scaling multilingual corpora and language models to 500 languages. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1082–1117, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.61. URL https://aclanthology.org/2023.acl-long.61.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In Trevor Cohn, Yulan He, and Yang Liu, editors, *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.147. URL https://aclanthology.org/2020.findings-emnlp.147.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1356. URL https://www.aclweb.org/anthology/P19-1356.

Pratik Jawanpuria, Mayank Meghwanshi, and Bamdev Mishra. A simple approach to learning unsupervised multilingual embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2995–3001, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.240. URL https://aclanthology.org/2020.emnlp-main.240.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistic*, 5:339–351, 2017. doi: 10.1162/tacl_a_00065. URL https://www.aclweb.org/anthology/Q17-1024.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.560. URL https://aclanthology.org/2020.acl-main.560.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F. T. Martins, and Alexandra Birch. Marian: Fast neural machine translation in C++. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2018. URL `http://www.aclweb.org/anthology/P18-4020`.

Kaliyaperumal Karthikeyan, Zihan Wang, Stephen Mayhew, and Dan Roth. Cross-lingual ability of multilingual BERT: An empirical study. *arXiv [e-Print archive]*, abs/1912.07840, 2019. URL `https://arxiv.org/abs/1912.07840`.

Slava M. Katz. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 35 (3):400–401, March 1987. URL `/bib/katz/Katz1987/katz87smoothing.pdf`.

Phillip Keung, Julian Salazar, Yichao Lu, and Noah A. Smith. Unsupervised bitext mining and translation via self-trained contextual embeddings. *Transactions of the Association for Computational Linguistics*, 8:828–841, 2020. doi: 10.1162/tacl_a_00348. URL `https://aclanthology.org/2020.tacl-1.53`.

Jyotsana Khatri and Pushpak Bhattacharyya. Filtering back-translated data in unsupervised neural machine translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4334–4339, Barcelona, Spain (Online), December 2020. International Committee on Computational Linguistics. doi: 10.18653/v1/2020.coling-main.383. URL `https://aclanthology.org/2020.coling-main.383`.

Yunsu Kim, Jiahui Geng, and Hermann Ney. Improving unsupervised word-by-word translation with language model and denoising autoencoder. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 862–868, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1101. URL `https://aclanthology.org/D18-1101`.

Yunsu Kim, Yingbo Gao, and Hermann Ney. Effective cross-lingual transfer of neural machine translation models without shared vocabularies. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1246–1257, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1120. URL `https://aclanthology.org/P19-1120`.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference for Learning Representations*, 2015. URL `http://arxiv.org/abs/1412.6980`.

Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 3294–3302. Curran Associates, Inc., 2015. URL `http://papers.nips.cc/paper/5950-skip-thought-vectors.pdf`.

Tom Kocmi. Exploring benefits of transfer learning in neural machine translation [dissertation thesis], 2020.

Tom Kocmi and Ondřej Bojar. Trivial transfer learning for low-resource neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 244–252, Brussels, October 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-6325. URL `https://www.aclweb.org/anthology/W18-6325`.

Tom Kocmi and Ondřej Bojar. Subgram: Extending skip-gram word representation with substrings. *Lecture Notes in Computer Science*, page 182–189, 2016. ISSN 1611-3349. doi: 10.1007/978-3-319-45510-5_21. URL `http://dx.doi.org/10.1007/978-3-319-45510-5_21`.

Philipp Koehn. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-3250`.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the ACL on Human Language Technology*, NAACL '03, pages 48–54, Stroudsburg, 2003. Association for Computational Linguistics. doi: 10.3115/1073445.1073462. URL `https://doi.org/10.3115/1073445.1073462`.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 177–180, Prague, June 2007. Association for Computational Linguistics. URL `https://www.aclweb.org/anthology/P07-2045`.

Julia Kreutzer, Isaac Caswell, Lisa Wang, Ahsan Wahab, Daan van Esch, Nasanbayar Ulzii-Orshikh, Allahsera Tapo, Nishant Subramani, Artem Sokolov, Claytone Sikasote, Monang Setyawan, Supheakmungkol Sarin, Sokhar Samb, Benoît Sagot, Clara Rivera, Annette Rios, Isabel Papadimitriou, Salomey Osei, Pedro Ortiz Suarez, Iroro Orife, Kelechi Ogueji, Andre Niyongabo Rubungo, Toan Q. Nguyen, Mathias Müller, André Müller, Shamsuddeen Hassan Muhammad, Nanda Muhammad, Ayanda Mnyakeni, Jamshidbek Mirzakhalov, Tapiwanashe Matangira, Colin Leong, Nze Lawson, Sneha Kudugunta, Yacine Jernite, Mathias Jenny, Orhan Firat, Bonaventure F. P. Dossou, Sakhile Dlamini, Nisansa de Silva, Sakine Çabuk Ballı, Stella Biderman, Alessia Battisti, Ahmed Baruwa, Ankur Bapna, Pallavi Baljekar, Israel Abebe Azime, Ayodele Awokoya, Duygu Ataman, Orevaoghene Ahia, Oghenefego Ahia, Sweta Agrawal, and Mofetoluwa Adeyemi. Quality at a glance: An audit of web-crawled multilingual datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72, 2022. doi: 10.1162/tacl_a_00447. URL `https://aclanthology.org/2022.tacl-1.4`.

Ivana Kvapilíková and Ondrej Bojar. CUNI submission to MT4All shared task. In *Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages*, pages 78–82, Marseille, France, June 2022. European Language Resources Association. URL `https://aclanthology.org/2022.sigul-1.10`.

Ivana Kvapilíková and Ondrej Bojar. Boosting unsupervised machine translation with pseudo-parallel data. In *Proceedings of Machine Translation Summit XIX: Research Track*, pages 135–147, Kyoto, Japan, 2023. Asia-Pacific Association for Machine Translation.

Ivana Kvapilíková, Dominik Machacek, and Ondrej Bojar. CUNI systems for the unsupervised news translation task in WMT 2019. In *Proceedings of the Fourth Conference on Machine Translation*, pages 241–248, Florence, August 2019. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/W19-5323`.

Ivana Kvapilíková, Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Ondřej Bojar. Unsupervised multilingual sentence embeddings for parallel corpus mining. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 255–262, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-srw.34. URL `https://aclanthology.org/2020.acl-srw.34`.

Ivana Kvapilíková. Unsupervised machine translation between czech and german language [bachelor's thesis]], 2020.

Guillaume Lample, Ludovic Denoyer, and Marc'Aurelio Ranzato. Unsupervised machine translation using monolingual corpora only. In *Proceedings of the 6th International Conference on Learning Representations*, 2018a. URL `http://arxiv.org/abs/1711.00043`.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium, October-November 2018b. Association for Computational Linguistics. doi: 10.18653/v1/D18-1549. URL `https://aclanthology.org/D18-1549`.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on EMNLP*, pages 5039–5049, 2018c. URL `https://aclanthology.info/papers/D18-1549/d18-1549`.

Yichong Leng, Xu Tan, Tao Qin, Xiang-Yang Li, and Tie-Yan Liu. Unsupervised pivot translation for distant languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 175–183, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1017. URL `https://aclanthology.org/P19-1017`.

Chongman Leong, Derek F. Wong, and Lidia S. Chao. Um-paligner: Neural network-based parallel sentence identification model. In Reinhard Rapp, Pierre Zweigenbaum, and Serge Sharoff, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France, may 2018. European Language Resources Association (ELRA). ISBN 979-10-95546-07-8. URL `http://lrec-conf.org/workshops/lrec2018/W8/pdf/7_W8.pdf`.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL `https://aclanthology.org/2020.acl-main.703`.

Zuchao Li, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. Explicit sentence compression for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34 No. 5*, pages 8311–8318. AAAI Press, 2020a.

Zuchao Li, Hai Zhao, Rui Wang, Kehai Chen, Masao Utiyama, and Eiichiro Sumita. SJTU-NICT's supervised and unsupervised neural machine translation systems for the WMT20 news translation task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 218–229, Online, November 2020b. Association for Computational Linguistics. URL `https://aclanthology.org/2020.wmt-1.22`.

Zuchao Li, Hai Zhao, Rui Wang, Masao Utiyama, and Eiichiro Sumita. Reference language based unsupervised neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4151–4162, Online, November 2020c. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.371. URL `https://aclanthology.org/2020.findings-emnlp.371`.

Zuchao Li, Masao Utiyama, Eiichiro Sumita, and Hai Zhao. Unsupervised neural machine translation with universal grammar. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3249–3264, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.261. URL `https://aclanthology.org/2021.emnlp-main.261`.

Jindřich Libovický, Rudolf Rosa, and Alexander M. Fraser. How language-neutral is multilingual BERT? *CoRR*, abs/1911.03310, 2019. URL `https://arxiv.org/abs/1911.03310`.

Robert Litschko, Goran Glavaš, Ivan Vulic, and Laura Dietz. Evaluating resource-lean cross-lingual embedding models in unsupervised retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR'19, page 1109–1112, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450361729. doi: 10.1145/3331184.3331324. URL `https://doi.org/10.1145/3331184.3331324`.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL `http://arxiv.org/abs/1907.11692`.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742, 2020. doi: 10.1162/tacl_a_00343. URL `https://aclanthology.org/2020.tacl-1.47`.

Jinliang Lu and Jiajun Zhang. Exploiting curriculum learning in unsupervised neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 924–934, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-emnlp.79. URL `https://aclanthology.org/2021.findings-emnlp.79`.

Marco Lui and Timothy Baldwin. langid.py: An off-the-shelf language identification tool. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, 2012. URL `https://www.aclweb.org/anthology/P12-3005`.

Xiaofei Ma, Zhiguo Wang, Patrick Ng, Ramesh Nallapati, and Bing Xiang. Universal text representation from bert: An empirical study. *CoRR*, abs/1910.07973, 2019. URL `https://arxiv.org/abs/1910.07973`.

Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, 1999. ISBN 0-262-13360-1.

Kelly Marchisio, Kevin Duh, and Philipp Koehn. When does unsupervised machine translation work? In *Proceedings of the Fifth Conference on Machine Translation*, pages 571–583, Online, November 2020. Association for Computational Linguistics. URL `https://aclanthology.org/2020.wmt-1.68`.

Benjamin Marie, Haipeng Sun, Rui Wang, Kehai Chen, Atsushi Fujita, Masao Utiyama, and Eiichiro Sumita. NICT's unsupervised neural and statistical machine translation systems for the WMT19 news translation task. In *Proceedings of the Fourth Conference on Machine Translation*, pages 294–301, Florence, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5330. URL `https://www.aclweb.org/anthology/W19-5330`.

Tomas Mikolov, Kai Chen, Greg S. Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013a. URL `http://arxiv.org/abs/1301.3781`.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. Exploiting similarities among languages for machine translation. *CoRR*, abs/1309.4168, 2013b. URL `http://arxiv.org/abs/1309.4168`.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc., 2013c. URL `http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionali pdf`.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Proc. Interspeech 2010*, pages 1045–1048, 2010. doi: 10.21437/Interspeech.2010-343.

Tasnim Mohiuddin, M Saiful Bari, and Shafiq Joty. LNMap: Departures from isomorphic assumption in bilingual lexicon induction through non-linear mapping in latent space, 2020.

Toan Q. Nguyen and David Chiang. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301, Taipei, Taiwan, November 2017. Asian Federation of Natural Language Processing. URL `https://aclanthology.org/I17-2050`.

Sosuke Nishikawa, Ryokan Ri, and Yoshimasa Tsuruoka. Data augmentation with unsupervised machine translation improves the structural similarity of cross-lingual word embeddings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: Student Research Workshop*, pages 163–173, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-srw.17. URL `https://aclanthology.org/2021.acl-srw.17`.

Franz Josef Och. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–

167, Sapporo, July 2003. Association for Computational Linguistics. doi: 10.3115/1075096.1075117. URL https://www.aclweb.org/anthology/P03-1021.

Franz Josef Och and Hermann Ney. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1), 2003.

Aitor Ormazabal, Mikel Artetxe, Gorka Labaka, Aitor Soroa, and Eneko Agirre. Analyzing the limitations of cross-lingual word embedding mappings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4990–4995, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1492. URL https://www.aclweb.org/anthology/P19-1492.

Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. Unsupervised learning of sentence embeddings using compositional n-gram features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 528–540, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1049. URL https://www.aclweb.org/anthology/N18-1049.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://www.aclweb.org/anthology/P02-1040.

Cheonbok Park, Yunwon Tae, TaeHee Kim, Soyoung Yang, Mohammad Azam Khan, Lucy Park, and Jaegul Choo. Unsupervised neural machine translation for low-resource domains via meta-learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2888–2901, Online, August 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.225. URL https://aclanthology.org/2021.acl-long.225.

Barun Patra, Joel Ruben Antony Moniz, Sarthak Garg, Matthew R. Gormley, and Graham Neubig. Bilingual lexicon induction with semi-supervision in non-isometric embedding spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 184–193, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1018. URL https://www.aclweb.org/anthology/P19-1018.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics.*, 2018.

Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1493. URL `https://www.aclweb.org/anthology/P19-1493`.

Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/W15-3049. URL `https://aclanthology.org/W15-3049`.

Maja Popović. chrF++: words helping character n-grams. In *Proceedings of the Second Conference on Machine Translation*, pages 612–618, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4770. URL `https://aclanthology.org/W17-4770`.

Nima Pourdamghani, Nada Aldarrab, Marjan Ghazvininejad, Kevin Knight, and Jonathan May. Translating translationese: A two-step approach to unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3057–3062, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1293. URL `https://aclanthology.org/P19-1293`.

Ofir Press and Lior Wolf. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain, April 2017. Association for Computational Linguistics. URL `https://aclanthology.org/E17-2025`.

Sujith Ravi and Kevin Knight. Deciphering foreign language. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 12–21, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL `https://aclanthology.org/P11-1002`.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.213. URL `https://aclanthology.org/2020.emnlp-main.213`.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2019. doi: 10.18653/v1/d19-1410. URL `http://dx.doi.org/10.18653/v1/d19-1410`.

Nils Reimers and Iryna Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.365. URL `https://aclanthology.org/2020.emnlp-main.365`.

Shuo Ren, Yu Wu, Shujie Liu, Ming Zhou, and Shuai Ma. Explicit cross-lingual pre-training for unsupervised machine translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 770–779, Hong Kong, China, November 2019a. Association for Computational Linguistics. doi: 10.18653/v1/D19-1071. URL `https://aclanthology.org/D19-1071`.

Shuo Ren, Zhirui Zhang, Shujie Liu, Ming Zhou, and Shuai Ma. Unsupervised neural machine translation with SMT as posterior regularization. In *Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33 No. 1*, pages 241–248, Honolulu, Hawaii, USA, 2019b. AAAI Press.

Shuo Ren, Yu Wu, Shujie Liu, Ming Zhou, and Shuai Ma. A retrieve-and-rewrite initialization method for unsupervised machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3498–3504, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.320. URL `https://aclanthology.org/2020.acl-main.320`.

Dana Ruiter, Cristina España-Bonet, and Josef van Genabith. Self-supervised neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1828–1834, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1178. URL `https://www.aclweb.org/anthology/P19-1178`.

Dana Ruiter, Dietrich Klakow, Josef van Genabith, and Cristina España-Bonet. Integrating unsupervised data generation into self-supervised neural machine translation for low-resource languages. In *Proceedings of Machine Translation Summit XVIII: Research Track*, pages 76–91, Virtual, August 2021. Association for Machine Translation in the Americas. URL `https://aclanthology.org/2021.mtsummit-research.7`.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. *Proceedings of the 2019 Conference of the North American Chapter of the ACL*, 2019. doi: 10.18653/v1/n19-1162. URL `http://dx.doi.org/10.18653/v1/n19-1162`.

Holger Schwenk. Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 228–234, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-2037. URL https://www.aclweb.org/anthology/P18-2037.

Holger Schwenk and Matthijs Douze. Learning joint multilingual sentence representations with neural machine translation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 157–167, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2619. URL https://www.aclweb.org/anthology/W17-2619.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. Ccmatrix: Mining billions of high-quality parallel sentences on the web, 2019.

Peter Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966. URL https://EconPapers.repec.org/RePEc:spr:psycho:v:31:y:1966:i:1:p:1-10.

Sukanta Sen, Kamal Kumar Gupta, Asif Ekbal, and Pushpak Bhattacharyya. Multilingual unsupervised NMT using shared encoder and language-specific decoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3083–3089, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1297. URL https://aclanthology.org/P19-1297.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, pages 1715–1725, Berlin, August 2016. Association for Computational Linguistics. doi: 10.18653/v1/P16-1162. URL https://www.aclweb.org/anthology/P16-1162.

Ahmad Shapiro, Mahmoud Salama, Omar Abdelhakim, Mohamed Fayed, Ayman Khalafallah, and Noha Adly. The AIC system for the WMT 2022 unsupervised MT and very low resource supervised MT task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 1117–1121, Abu Dhabi, United Arab Emirates (Hybrid), December 2022. Association for Computational Linguistics. URL https://aclanthology.org/2022.wmt-1.110.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. On the limitations of unsupervised bilingual dictionary induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 778–788, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1072. URL https://aclanthology.org/P18-1072.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. MASS: masked sequence to sequence pre-training for language generation. *CoRR*, abs/1905.02450, 2019. URL http://arxiv.org/abs/1905.02450.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014. URL http://jmlr.org/papers/v15/srivastava14a.html.

Dario Stojanovski, Viktor Hangya, Matthias Huck, and Alexander Fraser. The LMU munich unsupervised machine translation system for WMT19. In *Proceedings of the Fourth Conference on Machine Translation*, pages 393–399, Florence, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-5344. URL https://www.aclweb.org/anthology/W19-5344.

Thomas Storer. Linguistic isomorphisms. *Philosophy of Science*, 19(1):77–85, 1952. ISSN 00318248, 1539767X. URL http://www.jstor.org/stable/185100.

Jana Straková, Milan Straka, and Jan Hajič. Open-Source Tools for Morphology, Lemmatization, POS Tagging and Named Entity Recognition. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 13–18, Baltimore, June 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P/P14/P14-5003.pdf.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. Knowledge distillation for multilingual unsupervised neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3525–3535, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.324. URL https://aclanthology.org/2020.acl-main.324.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. Self-training for unsupervised neural machine translation in unbalanced training data scenarios. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3975–3981, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.311. URL https://aclanthology.org/2021.naacl-main.311.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, pages 3104–3112. Curran Associates, Inc., 2014. URL http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf.

Wilson L Taylor. "cloze procedure": A new tool for measuring readability. *Journalism quarterly*, 30(4):415–433, 1953.

Jörg Tiedemann. Parallel data, tools and interfaces in opus. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Ugur Dogan, Bente Maegaard, Joseph

Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may 2012. European Language Resources Association (ELRA). ISBN 978-2-9517408-7-7.

Chau Tran, Yuqing Tang, Xian Li, and Jiatao Gu. Cross-lingual retrieval for iterative self-supervised training. *CoRR*, abs/2006.09526, 2020. URL `https://arxiv.org/abs/2006.09526`.

Ahmet Üstün, Alexandre Berard, Laurent Besacier, and Matthias Gallé. Multilingual unsupervised neural machine translation with denoising adapters. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6650–6662, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.533. URL `https://aclanthology.org/2021.emnlp-main.533`.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL `http://arxiv.org/abs/1706.03762`.

Pascal Vincent, Hugo Larochelle, Y. Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1096–1103, 01 2008. doi: 10.1145/1390156.1390294.

Ivan Vulić, Goran Glavaš, Roi Reichart, and Anna Korhonen. Do we really need fully unsupervised cross-lingual embeddings? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4407–4418, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1449. URL `https://aclanthology.org/D19-1449`.

Ivan Vulić, Sebastian Ruder, and Anders Søgaard. Are all good word vector spaces isomorphic? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3178–3192, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.257. URL `https://aclanthology.org/2020.emnlp-main.257`.

Takashi Wada, Tomoharu Iwata, and Yuji Matsumoto. Unsupervised multilingual word embedding with limited resources using neural language models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3113–3124, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1300. URL `https://aclanthology.org/P19-1300`.

Hai Wang, Dian Yu, Kai Sun, Jianshu Chen, and Dong Yu. Improving pre-trained multilingual model with vocabulary expansion. In *Proceedings of the 23rd Conference on Computational*

Natural Language Learning (CoNLL), pages 316–327, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/K19-1030. URL `https://aclanthology.org/K19-1030`.

Mingxuan Wang, Hongxiao Bai, Hai Zhao, and Lei Li. Cross-lingual supervision improves unsupervised neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pages 89–96, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-industry.12. URL `https://aclanthology.org/2021.naacl-industry.12`.

Marion Weller-Di Marco and Alexander Fraser. Findings of the wmt 2022 shared tasks in unsupervised mt and very low resource supervised mt. In *Proceedings of the Seventh Conference on Machine Translation*, pages 801–805, Abu Dhabi, December 2022. Association for Computational Linguistics. URL `https://aclanthology.org/2022.wmt-1.73`.

John Wieting, Mohit Bansal, Kevin Gimpel, and Karen Livescu. Towards universal paraphrastic sentence embeddings. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. URL `https://arxiv.org/pdf/1511.08198`.

Jiawei Wu, Xin Wang, and William Yang Wang. Extract and edit: An alternative to back-translation for unsupervised neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1173–1183, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1120. URL `https://aclanthology.org/N19-1120`.

Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1077. URL `https://www.aclweb.org/anthology/D19-1077`.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado, May–June 2015. Association for Computational Linguistics. doi: 10.3115/v1/N15-1104. URL `https://aclanthology.org/N15-1104`.

Yinfei Yang, Gustavo Hernandez Abrego, Steve Yuan, Mandy Guo, Qinlan Shen, Daniel Cer, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. Improving multilingual sentence embedding using bi-directional dual encoder with additive margin softmax. *Proceedings of the*

Twenty-Eighth International Joint Conference on Artificial Intelligence, Aug 2019. doi: 10.24963/ijcai.2019/746. URL `http://dx.doi.org/10.24963/ijcai.2019/746`.

Yinfei Yang, Daniel Cer, Amin Ahmad, Mandy Guo, Jax Law, Noah Constant, Gustavo Hernandez Abrego, Steve Yuan, Chris Tar, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. Multilingual universal sentence encoder for semantic retrieval. In Asli Celikyilmaz and Tsung-Hsien Wen, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 87–94, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-demos.12. URL `https://aclanthology.org/2020.acl-demos.12`.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. Unsupervised neural machine translation with weight sharing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 46–55, Melbourne, Australia, July 2018. Association for Computational Linguistics. doi: 10.18653/v1/P18-1005. URL `https://aclanthology.org/P18-1005`.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1959–1970, Vancouver, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1179. URL `https://www.aclweb.org/anthology/P17-1179`.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on EMNLP*, pages 1568–1575, Austin, November 2016. Association for Computational Linguistics. doi: 10.18653/v1/D16-1163. URL `https://www.aclweb.org/anthology/D16-1163`.

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. Overview of the second BUCC shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2512. URL `https://www.aclweb.org/anthology/W17-2512`.

A. Appendix

A.1 Additional Evaluation (COMET and chrf++)

	DE-HSB	CS-DE	EN-KA	EN-KK	EN-UK			
WMT22 best	45.4	43.8	-	-	-	-	-	-
ChatGPT	30.7	-	28.2	-	30.9	-	**52.4**	-
OBT (baseline)	50.9	55.5	27.4	28.8	2.4	2.9	31.3	37.4
PseudoPar	34.8	37.9	23.1	26.6	17.5	22.9	23.3	32.1
OBT+PseudoPar	53.3	56.5	35.5	39.4	31.4	36.4	35.9	45.7
↦OBT	**56.1**	**59.3**	**37.8**	**40.9**	**33.6**	**37.3**	**41.2**	**49.1**

	DE-HSB	CS-DE	EN-KA	EN-KK	EN-UK			
de Gibert Bonet (2022)	-	-	n/a	-	n/a	-	n/a	-
OBT (baseline)	-	-	32.2	33.8	1.9	2.1	43.2	37.5
PseudoPar	-	-	22.6	27.0	27.3	32.1	39.2	35.3
OBT+PseudoPar	-	-	38.2	25.6	**41.9**	**41.1**	**55.9**	**47.5**
↦OBT	-	-	**43.3**	**47.0**	39.1	38.4	54.7	**48.0**

Table A.1: MT performance of our systems measured by chrF++ scores on the general test set (top) and the legal test set (bottom). Compared to the WMT22 winner [Shapiro et al., 2022] and ChatGPT. The score could not be computed for the system trained by de Gibert Bonet et al. [2022] as we do not have access to their translations.

	DE-HSB	CS-DE	EN-KA	EN-KK	EN-UK			
WMT22 best	0.58	0.52	-	-	-	-	-	-
ChatGPT	0.55	-	0.56	-	0.63	-	0.89	-
OBT (baseline)	0.59	0.68	0.55	0.55	0.40	0.57	0.60	0.60
PseudoPar	0.56	0.56	0.62	0.60	0.62	0.59	0.63	0.62
OBT+PseudoPar	0.62	0.71	0.70	0.70	0.71	0.67	0.71	0.72
↦OBT	**0.63**	**0.72**	**0.71**	**0.72**	**0.71**	**0.68**	**0.73**	**0.74**

	DE-HSB	CS-DE	EN-KA	EN-KK	EN-UK			
de Gibert Bonet (2022)	-	-	n/a	-	n/a	-	n/a	-
OBT (baseline)	-	-	0.58	0.57	0.45	0.65	0.76	0.65
PseudoPar	-	-	0.59	0.58	0.79	0.71	0.77	0.63
OBT+PseudoPar	-	-	0.69	0.69	**0.86**	**0.74**	**0.85**	0.72
↦OBT	-	-	**0.71**	**0.70**	0.85	0.73	0.84	**0.75**

Table A.2: MT performance of our systems measured by COMET scores on the general test set (top) and the legal test set (bottom). Compared to the WMT22 winner [Shapiro et al., 2022] and ChatGPT. The score could not be computed for the system trained by de Gibert Bonet et al. [2022] as we do not have access to their translations.

	EN-AS	AS-EN	EN-MNI	MNI-EN
OBT (baseline)	13.2	16.7	0.5	0.4
OBT+PseudoPar	18.4	**21.8**	11.3	14.5
OBT+PseudoPar (improved)	**19.1**	**21.9**	**16.4**	**16.3**

Table A.3: chrF++ score of EN-AS and EN-MNI unsupervised MT systems on the WMT23 test set.

	EN-KHA	KHA-EN	EN-MZ	MZ-EN
OBT (baseline)	29.9	22.2	20.5	16.5
OBT+PseudoPar	28.1	20.6	**26.8**	**20.5**
OBT+PseudoPar (improved)	**34.7**	**26.2**	24.8	**20.1**

Table A.4: chrF++ score of EN-KHA and EN-MZ unsupervised MT systems on the WMT23 test set.

	EN-AS	AS-EN	EN-MNI	MNI-EN
OBT (baseline)	0.55	0.47	0.26	0.30
OBT+PseudoPar	0.62	**0.54**	0.55	0.40
OBT+PseudoPar (improved)	**0.63**	**0.54**	**0.58**	**0.42**

Table A.5: COMET score of EN-AS and EN-MNI unsupervised MT systems on the WMT23 test set.

	EN-KHA	KHA-EN	EN-MZ	MZ-EN
OBT (baseline)	0.69	0.44	0.57	0.41
OBT+PseudoPar	0.70	0.44	**0.62**	0.45
OBT+PseudoPar (improved)	**0.72**	**0.50**	0.60	**0.46**

Table A.6: COMET score of EN-KHA and EN-MZ unsupervised MT systems on the WMT23 test set.

	EN-AS	EN-KHA	EN-MNI	EN-MZ
AuthPar+OBT (semi-sup)	**37.7**	**38.9**	**55.7**	**52.9**
PseudoPar+AuthPar+OBT (semi-sup)	36.6	37.9	**56.1**	**52.7**
OBT (unsup)	13.2	29.9	0.5	20.5
OBT+PseudoPar$\mapsto$OBT (unsup)	19.1	34.7	16.4	24.8

Table A.7: chrF++ score of EN-AS, EN-KHA, EN-MNI and EN-MZ semi-supervised MT systems on the WMT23 test set.

	EN-AS	EN-KHA	EN-MNI	EN-MZ
AuthPar+OBT (semi-sup)	**0.75**	**0.75**	**0.81**	**0.77**
PseudoPar+AuthPar+OBT (semi-sup)	0.74	**0.75**	**0.81**	0.76
OBT (unsup)	0.55	0.69	0.36	0.67
OBT+PseudoPar$\mapsto$OBT (unsup)	0.63	0.72	0.58	0.60

Table A.8: COMET score of EN-AS, EN-KHA, EN-MNI and EN-MZ semi-supervised MT systems on the WMT23 test set.

A.2 Tools and Configuration

In our experiments, we use the following tools:

- `LASER`[1] for parallel sentence search and creating pseudo-parallel corpora. We modified the original implementation to support similarity search in larger data sets and to support different encoders.

- `Monoses`[2] to create the unsupervised phrase-based system.

- `MUSE`[3] for unsupervised alignment of static embeddings using adversarial training.

- `VecMap`[4] for unsupervised alignment of static embeddings using similarity matrices.

- `XLM`[5] for MT training of most of our translation models (unless stated otherwise in the text). Alternatively, in several experiments we used `Marian`[6] or `fairseq`[7].

  For language model pre-training, we use mini-batches of 64 text streams (256 tokens per stream) per GPU and Adam [Kingma and Ba, 2015] optimization with a learning rate λ=0.0001. For denoising and MT fine-tuning, we use mini-batches of 3,400 tokens per GPU and Adam optimization with a linear warm-up (beta1=0.9, beta2=0.98, λ=0.0001). The models are trained on 8 GPUs, or using gradient accumulation to reach an effective batch size corresponding to 8 GPUs.

  For fine-tuning the XLM-100 model using the TLM objective, we use the batch size of 8 sentences and train on 1 GPU. For fine-tuning the XLM-100 model for unsupported languages using the MLM objective, we use the batch size of 40 sentences per GPU and train on 2 GPUs. We use Adam optimization with a leaning rate λ=0.00005.

  The training hyperparameters were selected based on the related work as tuning them was beyond our computation capacity.

For evaluation, we used the following tools:

- `sacrebleu`[8] to calculate the BLEU and chrF++ metrics with the configuration `sacrebleu -tok '13a' -s 'exp' -m bleu chrf --chrf-word-order 2 --confidence`;

- `COMET`[9] to calculate COMET scores using the default model `wmt22-comet-da`.

[1] https://github.com/facebookresearch/LASER
[2] https://github.com/artetxem/monoses/tree/master
[3] https://github.com/facebookresearch/MUSE
[4] https://github.com/artetxem/vecmap
[5] https://github.com/facebookresearch/XLM
[6] https://github.com/marian-nmt/marian
[7] https://github.com/facebookresearch/fairseq
[8] https://github.com/mjpost/sacrebleu
[9] https://github.com/Unbabel/COMET

List of Figures

List of Tables

List of Abbreviations

BLEU	Bilingual Evaluation Understudy
BP	Brevity Penalty
BERT	Bidirectional Encoder Representations
BPE	Byte Pair Encoding
COMET	Crosslingual Optimized Metric for Evaluation of Translation
chrF	Character-level F-score
DAE	Denoising Autoencoding
ELMo	Embeddings from Language Models
MLM	Masked Language Modelling
MT	Machine Translation
NLP	Natural Language Processing
NMT	Neural Machine Translation
OBT	Online Back-Translation
PBMT	Phrase-Based Machine Translation
PCM	Parallel Corpus Mining
SGD	Stochastic Gradient Descent
TLM	Translation Language Modelling
UMT	Unsupervised Machine Translation
UNMT	Unsupervised Neural Machine Translation
UPBMT	Unsupervised Phrase-Based Machine Translation
WMT	Workshop on Machine Translation

List of publications

IVANA KVAPILÍKOVÁ, ONDŘEJ BOJAR (2023): Boosting Unsupervised Machine Translation with Pseudo-Parallel Data. In: *Proceedings of the Eighth Conference on Machine Translation*, in press, Association for Computational Linguistics: Stroudsburg, PA, USA

IVANA KVAPILÍKOVÁ, ONDŘEJ BOJAR (2023): Low-Resource Machine Translation Systems for Indic Languages. In: *Proceedings of Machine Translation Summit XIX vol. 1: Research Track*, pp. 135-147, Asia-Pacific Association for Machine Translation (AAMT), Kyoto, Japan

IVANA KVAPILÍKOVÁ, ONDŘEJ BOJAR (2022): CUNI Submission to MT4All Shared Task. In: *Proceedings of the LREC 2022 Workshop of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages (SIGUL 2022)*, pp. 78-82, European Language Resources Association (ELRA), Paris, France, ISBN 979-10-95546-91-7

BOREK POŽÁR, KLÁRA TAUCHMANOVÁ, KRISTÝNA NEUMANNOVÁ, IVANA KVAPILÍKOVÁ, ONDŘEJ BOJAR (2022): CUNI Submission to the BUCC 2022 Shared Task on Bilingual Term Alignment. In: *Proceedings of the LREC 2022 15th Workshop on Building and Using Comparable Corpora*, pp. 43-49, European Language Resources Association, Paris, France, ISBN 979-10-95546-94-8

MD MAHFUZ IBN ALAM, IVANA KVAPILÍKOVÁ, ANTONIOS ANASTASOPOULOS, LAURENT BESACIER, GEORGIANA DINU, MARCELLO FEDERICO, MATTHIAS GALLÉ, PHILIPP KOEHN, VASSILINA NIKOULINA, KWEON WOO JUNG (2021): Findings of the WMT Shared Task on Machine Translation Using Terminologies. In: *Proceedings of the Sixth Conference on Machine Translation*, pp. 652-663, Association for Computational Linguistics, Stroudsburg, PA, USA, ISBN 978-1-954085-94-7

IVANA KVAPILÍKOVÁ, ONDŘEJ BOJAR (2021): Machine Translation of Covid-19 Information Resources via Multilingual Transfer. In: *ITAT 2021 2nd Workshop on Automata, Formal and Natural Languages – WAFNL 2021*, pp. 176-181, Faculty of Mathematics and Physics, Praha, Czechia

IVANA KVAPILÍKOVÁ, MIKEL ARTETXE, GORKA LABAKA, ENEKO AGIRRE, ONDŘEJ BOJAR (2020): Unsupervised Multilingual Sentence Embeddings for Parallel Corpus Mining. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pp. 255-262, Association for Computational Linguistics, Stroudsburg, PA, USA, ISBN 978-1-952148-03-3

IVANA KVAPILÍKOVÁ, TOM KOCMI, ONDŘEJ BOJAR (2020): CUNI Systems for the Unsupervised and Very Low Resource Translation Task in WMT20. In: *Fifth Conference* *on Machine Translation - Proceedings of the Conference*, pp. 1123-1128, Association for Computational Linguistics, Stroudsburg, PA, USA, ISBN 978-1-948087-81-0

IVANA KVAPILÍKOVÁ, DOMINIK MACHÁČEK, ONDŘEJ BOJAR (2019): CUNI Systems for the Unsupervised News Translation Task in WMT 2019. In: *Fourth Conference on Machine Translation - Proceedings of the Conference*, pp. 241-248, Association for Computational Linguistics, Stroudsburg, PA, USA, ISBN 978-1-950737-27-7