Effectively Compiling Parallel Corpora for Machine Translation in Resource-Scarce Conditions

by

Steinþór Steingrímsson

Dissertation submitted to the Department of Computer Science
at Reykjavík University in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

May 2023

Thesis Committee:

Hrafn Loftsson, Supervisor
Associate Professor, Reykjavík University, Iceland

Andy Way, Co-supervisor
Professor, Dublin City University, Ireland

Mikel Forcada, Committee Member
Professor, Universitat d'Alacant, Spain

Kepa Sarasola, Committee Member
Professor, Universidad del País Vasco, Spain

Helena Gorete Silva Moniz, External Examiner
Assistant Professor, University of Lisbon, Portugal

ISBN 978-9935-539-18-2 Print version
ISBN 978-9935-539-19-9 Electronic version
Authors ORCID: 0000-0002-9776-9507

The undersigned hereby certify that they recommend to the Department of Computer Science at Reykjavík University for acceptance this Dissertation entitled **Effectively Compiling Parallel Corpora for Machine Translation in Resource-Scarce Conditions** submitted by **Steinþór Steingrímsson** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy (Ph.D.)**

Hrafn Loftsson, Supervisor
Associate Professor, Reykjavík University, Iceland

Andy Way, Co-supervisor
Professor, Dublin City University, Ireland

Signed by MIKEL LORENZO FORCADA
ZUBIZARRETA - NIF:***4487** on
16/05/2023 with a certificate
issued by ACCVCA-120.

Mikel Forcada, Committee Member
Professor, Universitat d'Alacant, Spain

Kepa Sarasola

Kepa Sarasola, Committee Member
Professor, Universidad del País Vasco, Spain

Helena Gorete Silva Moniz, External Examiner
Assistant Professor, University of Lisbon, Portugal

The undersigned hereby grants permission to the Reykjavík University Library to reproduce single copies of this Dissertation entitled **Effectively Compiling Parallel Corpora for Machine Translation in Resource-Scarce Conditions** and to lend or sell such copies for private, scholarly or scientific research purposes only.
The author reserves all other publication and other rights in association with the copyright in the Dissertation, and except as herein before provided, neither the Dissertation nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatsoever without the author's prior written permission.

19. maí 2023
....
date

Steinþór Steingrímsson
....
Steinþór Steingrímsson
Doctor of Philosophy

Effectively Compiling Parallel Corpora for Machine Translation in Resource-Scarce Conditions

Steinþór Steingrímsson

May 2023

Abstract

For machine translation (MT) systems to produce accurate and fluent translations, reliable parallel corpora are key. Errors, due to misalignments or inadequate filtering during compilation of a parallel corpus, can have detrimental effects on the performance of an MT system trained on the data. Moreover, when the corpus is too small, the MT system may not be able to capture the complexities of the source and target languages and produce accurate translations. However, obtaining high-quality parallel data is often a challenging task, even more so for languages with a low number of speakers or rich morphology exacerbating the data sparsity problem. It is thus imperative to develop accurate methods for processing parallel corpora that can help make the most of what is available.

In this thesis, we address this challenge by exploring various methods for processing parallel corpora to maximize their usefulness for MT. First, we investigate a variety of classifiers and scoring mechanisms used for filtering parallel corpora, looking into how efficient they are at removing data detrimental to MT training and retaining useful data. We find that different filtering strategies suit different datasets and that filtering separately for different translation directions can yield better translations in downstream MT tasks. Second, we examine different approaches to sentence alignment, compare their effectiveness, and show that combining multiple methods can improve alignment accuracy. Third, we experiment with comparable corpora mining methods to extract even more useful data from sentences that had previously been discarded, showing that this often overlooked data is a potential source of useful training data. Finally, we manually evaluate translations generated by MT systems trained on our processed datasets, most suitable for each translation direction, confirming the advantages of our applied methods.

Our findings highlight the importance of careful processing and curation of parallel corpora for MT. We propose approaches for maximizing the utility of available parallel data, particularly for scenarios where resources are scarce, contributing to the development of more accurate and reliable MT systems.

Skilvirk smíði samhliða málheilda fyrir þýðingarvélar við gagnarýrar aðstæður

Steinþór Steingrímsson

maí 2023

Útdráttur

Áreiðanlegar samhliða málheildir eru lykillinn að því að hægt sé að þjálfa þýðingarvélar, sem geta myndað nákvæmar þýðingar sem flæða vel á markmálinu. Skekkjur í þjálfunargögnum, sem koma til vegna rangrar samröðunar setninga eða ófullnægjandi síunar við smíði samhliða málheilda, geta spillt gæðum þýðingarvélar sem þjálfuð er á gögnunum. Of lítil samhliða málheild getur jafnframt orðið til þess að þýðingarvélin nái ekki tökum á málfræði eða öðrum blæbrigðum frum- og markmálanna og myndi þess vegna ónákvæmar þýðingar. Það getur hins vegar verið flókið og erfitt að tryggja hámarksgæði þjálfunargagna við úrvinnslu samhliða texta, ekki síst þegar um er að ræða texta á tungumálum sem fáir tala eða þegar flóknar beygingar og virk orðmyndun auka á vandann við að greina rýr gögn. Þegar samhliða málheildir eru settar saman er því afar mikilvægt að þróa nákvæmar aðferðir sem miða að því að nýta sem allra best þau gögn sem til eru.

Í þessari ritgerð tökumst við á við þetta vandamál með því að kanna ýmsar aðferðir til að vinna gögn við smíði samhliða málheilda með það að leiðarljósi að hámarka notagildi gagnanna fyrir vélþýðingar. Í fyrsta lagi rannsökum við nokkrar gerðir flokkara og matsaðferðir sem notaðar eru til að sía samhliða málheildir. Við skoðum hversu árangursríkar þær eru til að fjarlægja setningapör sem geta dregið úr gæðum þýðingarvéla ef þau eru hluti þjálfunargagna og hversu líklegar aðferðirnar eru til að halda eftir þeim setningapörum sem búast má við að séu best fallnar til að bæta þýðingarvélarnar. Við komumst að því að mismunandi síunaraðferðir henta mismunandi gagnasöfnun og að með því að sía sérstaklega fyrir hverja þýðingarátt má bæta gæði þýðinga þeirra véla sem þjálfaðar eru á gögnunum. Í öðru lagi skoðum við mismunandi aðferðir við samröðun setninga, berum saman markvirkni þeirra og sýnum að með því að láta margar mismunandi aðferðir vinna saman getum við aukið nákvæmni samröðunarinnar. Í þriðja lagi gerum við tilraunir með aðferðir til að vinna samhliða gögn úr sambærilegum málheildum, og beitum þeim aðferðum til að draga nýtileg gögn úr setningum og setningapörum sem hafnað hefur verið á fyrri stigum í smíði þjálfunargagnanna. Við sýnum með nokkrum tilraunum að mögulegt er að nýta þessi gögn, sem yfirleitt er litið fram hjá, til að stækka samhliða þjálfunarmálheildir með nýtilegum gögnum og þar með bæta þýðingarvélar sem þjálfaðar eru á þeim. Að lokum metum við handvirkt þýðingar myndaðar af þýðingarvélum sem þjálfaðar eru á gögnum sem unnin hafa verið með okkar aðferðum, en það mat staðfestir gagnsemi þeirra aðferða sem við beitum.

Niðurstöður okkar undirstrika mikilvægi vandaðrar greiningar og gagnavinnslu við smíði samhliða málheilda sem notaðar eru til að þjálfa þýðingarvélar. Við kynnum aðferðir sem hámarka notagildi tiltækra samhliða gagna, ekki sístþegar takmarkað magn gagna er fyrir hendi, og stuðlum þannig að þróun nákvæmari og áreiðanlegri þýðingarvéla.

Acknowledgements

The work described herein has occupied a significant share of my waking hours, and probably sleeping hours too, for the last four years. A large group of people have contributed to the completion of this Ph.D. thesis, with their support, guidance, collaboration and patience. First of all I would like to express my deepest gratitude to my wife, Sólveig and my daughters, Þórey Kristín and Dagrún Hrefna for their love and support, and my daughters especially for their persistent encouragement and motivation to finish in a timely manner.

I am very thankful for the guidance and support of my supervisors, Hrafn Loftsson, associate professor at Reykjavik University, and co-supervisor Andy Way, professor at Dublin City University. I have thoroughly enjoyed discussing my work with them, and their advice, insightful suggestions and general good spirit has been an immense help. I obviously couldn't have done this without them.

The two examiners in the Ph.D. committee, Mikel Forcada, professor at Universitat d'Alacant and Kepa Sarasola, professor at Universidad del País Vasco, made a much appreciated effort to show me how I could improve my work and their detailed comments helped make this thesis better.

I would also like to express my great appreciation to the many collaborators: Örvar Kárason who worked with me on the part-of-speech tagger, Pintu Lohar who worked with me on mining comparable corpora, Luke O'Brien who took part in the work on the bilingual lexicon and Finnur Ágúst Ingimundarson and Árni Davíð Magnússon who helped with evaluation and compiling evaluation sets. They were a joy to work with and their expertise, shared knowledge, and diverse perspectives have greatly enriched my research experience and enhanced the quality of my work. Furthermore, I would like to thank those who participated in the manual evaluation of the MT models, and finally my co-workers at the Árni Magnússon Institute in Icelandic Studies for their understanding and willingness to give me the needed time to work on the thesis, especially Guðrún Nordal, the director of the institute, and Einar Freyr Sigurðsson for his support and encouragement and always being ready to pick up the slack.

Eiríkur Rögnvaldsson taught the first course I attended on language technology more than twenty years ago. Later he hired me to work on a project with him and Sigrún Helgadóttir. Their enthusiasm and dedication towards the subject were not only infectious but also instrumental in shaping my decision to delve deeper into this field and for that I am grateful.

This work was supported in part by the Icelandic Centre for Research (RANNIS), grant number 228654-051, in part by the National Language Technology Programme for Icelandic and in part by Reykjavik University.

Contents

List of Figures

List of Tables

List of Abbreviations

ABLTagger	Augmented BiLSTM Tagger
BERT	Bidirectional Encoder Representations from Transformers
BiLSTM	Bidirectional Long Short-Term Memory
BLEU	Bilingual Evaluation Understudy
BLI	Bilingual Lexicon Induction
BPE	Byte-Pair Encodings
BUCC	Building and Using Comparable Corpora
CLIR	Cross-Language Information Retrieval
CNN	Convolutional Neural Networks
CSLS	Cross-domain Similarity Local Scaling
DaC	Divide-and-Conquer
DA	Direct Assessment
DMII	Database of Modern Icelandic Inflection
DP	Dynamic Programming
EEA	European Economic Area
EMA	European Medicines Agency
ESO	European Southern Observatory
GPT	Generative Pre-trained Transformer
HMM	Hidden Markov Models
IFD	Icelandic Frequency Dictionary
IGC	The Icelandic Gigaword Corpus
LaBSE	Language-agnostic BERT Sentence Embedding
LASER	Language-Agnostic Sentence Representations
LC	Lexical Category
LLM	Large Language Model
LM	Language Model
LSTM	Long Short-Term Memory
mBART	Multilingual BART
MERT	Minimum Error Rate Training
METEOR	Metric for Evaluation of Translation with Explicit ORdering
ML	Machine Learning
MLM	Multilingual Language Model
MODEL	Multilingual Open Data for EU Languages
MQM	Multidimensional Quality Metrics
MRL	Morphologically Rich Language
MT	Machine Translation

NLP	Natural Language Processing
NLTPI	National Language Technology Programme for Icelandic
NMT	Neural Machine Translation
NN	Nearest Neighbour
OCR	Optical Character Recognition
OOV	Out-of-vocabulary
OTIC	One Time Inverse Consultation
PBSMT	Phrase-Based Statistical Machine Translation
PoS	Part-of-Speech
PUD	Parallel Universal Dependencies
RBMT	Rule-Based Machine Translation
RNN	Recurrent Neural Networks
SMT	Statistical Machine Translation
SOTA	State-of-the-art
SVM	Support Vector Machine
TIAD	Translation Inference Across Dictionaries
UD	Universal Dependencies
WMT	Workshop on Machine Translation (now: Conference on Machine Translation)
XLM-R	Crosslingual Language Model-Roberta

Chapter 1

Introduction

1.1 Motivation

Reliable parallel corpora are key to developing good machine translation (MT) systems. However, abundant parallel data can be hard to come by, especially in the case of low-resource languages. Indeed, the lack of parallel data can also be a problem for medium-resource languages, where some data may be available, but often limited to only a few domains. When the data sparsity problem is exacerbated by rich morphology, in particular where the task is to translate into a morphologically rich language (MRL), it can become a major limitation (Dhar et al., 2022). When the corpus is too small, the MT system may not be able to capture the complexities of the source and target languages and produce accurate translations. Web-scraped sentence pairs are often used to supplement training data to mitigate the low-resource problem. Larger corpora are generally better for training, but having a large corpus is not enough.

Parallel corpora commonly contain errors due to misalignments, inaccurate translations and machine translated content or inadequate filtering during compilation (see e.g. Kaalep and Veskis (2007); Zariņa et al. (2015)). The errors can potentially have detrimental effects on the performance of MT systems trained on the corpora. Neural machine translation (NMT) systems have been shown to be sensitive to noise in the training data (Khayrallah and Koehn, 2018), where noise is defined as erroneous segments that degrade the performance of systems trained on the data. Noisy data can introduce inconsistencies and ambiguities, leading to inaccurate translations. Web-scraped corpora are also typically noisy and need extensive cleaning and filtering to be useful (Kreutzer et al., 2022). Where parallel data is not plentiful, it is therefore highly important to be able to align, filter and mine available datasets effectively, if we want our MT models to be as accurate as our chosen architecture can offer.

In this thesis, we investigate different approaches to sentence alignment and parallel corpus filtering with the aim of making the most of what data is available. The goal is to maximise the number of correctly translated segments in a corpus and minimise noise. In order to do that, it is imperative to be able to accurately align and filter the data.

When aligning parallel documents on the sentence level, the aim is to pair all semantically equivalent segments without misalignments or extraneous or missing data in either language. We want to investigate if approaches to alleviate data sparsity problems when we have an MRL on one side in a bilingual corpus can help with alignment. Moreover, we want to explore whether a state-of-the-art (SOTA) multilingual sentence embeddings model can be used to efficiently score sentences when an exhaustive shortest-path algorithm is used to maximize accuracy for the sentence-alignment problem.

In filtering aligned data, various scoring mechanisms are typically used to decide whether a sentence pair should be included in the training data being compiled or if it should be excluded. Working with MRLs, the number of word forms can decrease scoring accuracy due to out-of-vocabulary (OOV) problems, especially when data is sparse. We experiment with lemmatizing the corpora to facilitate filtering, specifically when working with statistical approaches that do not make use of sentence embedding models, as lemmas may give more accurate information on frequency of word usage than inflected word forms when resources are limited.

In most existing work on compiling parallel corpora, a corpus is compiled for a language pair without regard to the translation direction. The same dataset is then used to train models for translating both directions. We challenge this approach. Back-translation (Sennrich et al., 2016a) is a widely used data augmentation technique in which synthetic parallel data is created by translating target language texts to the source language using an available MT system. The target language texts are fluent and grammatically correct, but the machine-translated sentences in the source language may in some cases be incorrect or inaccurate renditions of the target language. Similarly, we hypothesise that different requirements should apply to source and target data in a parallel corpus and thus the same filtering approaches may not necessarily be suitable for different translation directions.

Comparable corpora can be mined for parallel sentence pairs, which in turn can be used as a part of training data for MT. Another data source, often overlooked, is the data discarded during the alignment and filtering process when training data are compiled from parallel documents. We investigate whether we can make use of approaches shown to work for comparable corpora to mine such discarded data for parallel segments.

Having explored approaches for increasing the quality of parallel corpora as well as enlarging the corpora, we inspect the effects of different alignment and filtering approaches and select those that produce the highest quality training sets. We evaluate these different approaches to see if they not only increase the translation quality of MT systems as measured by automatic metrics, but also as perceived by humans.

1.2 Definitions

In Section 1.1, we introduced terminology that will be used extensively throughout this thesis. In this section, we aim to provide definitions for these terms.

When discussing the output of MT systems, we are primarily concerned with conveying the meaning without adding to or removing from the original text. We talk about *high-quality*, *accurate* or *correct* translations when referring to these qualities. Incorrect or inaccurate translations lack some or all of these qualities. They may not convey the meaning of the source correctly, they may add parts which do not exist in the source, or remove something important in the translation.

Data used to train MT systems in this thesis is made up of parallel sentence pairs. A *parallel sentence pair* is a source sentence and its translation in the target language. In a good sentence pair, the target sentence should contain an accurate translation of the source and nothing else. If the translation does not convey the meaning of the original, we define it as an *erroneous sentence pair*. If the content of either one or both of the sentences are only partially represented in the other sentence, we talk about *partial alignments*. These may occur for a number of reasons, e.g. mistakes in the sentence alignment process, differences in sentence structure or omissions in translation.

Parallel sentence pairs are usually extracted either from parallel corpora or comparable corpora. *Parallel corpora* are collections of texts composed of pairs of documents, where each document in the source language has a corresponding translation in the target language. *Comparable corpora*, are collections of documents in two or more languages that share similar content, domain or theme, but are not direct translations of one another.

When we talk about *good machine translation systems*, we are referring to a system that produces high-quality translations as defined above, translations that convey the meaning of the source without adding to it or removing information from it. A good machine translation system should also produce *fluent texts*, easily understood by a native speaker in the target language.

Throughout the thesis we talk about noise in the training data. In this context, we define *noise* as sentence pairs that do not help improve the translation quality of a machine translation system, but instead are detrimental and may degrade the performance of the systems trained on the data, leading to lower quality translations.

1.3 Research Questions

The aim of this thesis is to address four main research questions on the topic of compiling training data for MT by aligning, filtering and mining bilingual texts:

RQ1: How can we filter parallel corpora to minimize noise, and still lose little or no useful data from the original texts?
Different approaches have been taken for filtering parallel corpora, from rule-based to classifiers and score-based methods. In order to gain an insight into the effectiveness of various approaches, we compare them by manually evaluating random samples of data after applying each filtering approach. It is also important to know how the manual evaluation aligns with results of downstream MT tasks, as the final goal is to build better MT systems and this may tell us something about what constitutes useful data for NMT training.

RQ2: To what degree should we consider filtering parallel corpora for MT training to be independent of the dataset and languages being filtered, and the intended translation direction of the MT system being built?
The objective here is to try to understand to what extent selection of methods for processing parallel corpora should be dependent on the data. The usefulness of back-translations for improving MT quality suggests that we should be more concerned with the quality of texts in the target language, i.e. that they are fluent, grammatically correct and accurate translations of the source.

When source and target languages have different levels of morphological complexity, it is reasonable to ask whether data sparsity may be more of a concern for the morphologically richer language. Considering this, we want to investigate whether different approaches might be appropriate for compiling training data for different translation directions.

RQ3: Is sentence alignment accuracy important for the results of a downstream MT task, or is effective filtering of the training data enough?
While accurate sentence alignment is important, it is the role of the filtering process to remove misalignments from the training set. Vecalign, the SOTA in sentence alignment, previous to the work in this thesis, achieves an F_1 score of 0.9 (Thompson and Koehn, 2019)

when tested on a popular German–French evaluation set (Volk et al., 2010). Even if other approaches could improve on that score, the number of correctly aligned sentence pairs would only increase slightly. Would a small improvement translate into better quality translations in a downstream MT task?

RQ4: Are text segments discarded during sentence alignment and filtering suitable as a source for mining useful sentence pairs for MT training?
Traditionally, MT training data is compiled from parallel documents by aligning the data on the sentence level and then filtering out sentence pairs deemed unlikely to improve system performance, as well as sentences that do not meet the requirements of the MT training approach. This discarded data could be considered as comparable corpora, and, as such, an interesting potential source for mining parallel sentence pairs.

In attempting to answer these four questions, we develop tools and resources necessary to aid our research. That work brings up its own research questions, which will be introduced and discussed in the relevant chapters and sections.

1.4 Contributions

We summarise our main contributions, some of which have previously been described in reviewed publications. In our work, we mainly work with the English–Icelandic language pair and some of our contributions relate mainly to working with Icelandic. We start by describing tools and datasets made in order to facilitate our research on alignment and filtering parallel data, some of which are aimed at work on Icelandic but others are more generic:

- We developed a Part-of-Speech (PoS) tagger for Icelandic, outperforming previous taggers for the language by a large margin. Our tagger uses a bidirectional long short-term memory (BiLSTM) model, augmented with a morphological lexicon. In order to increase its accuracy, we devised a two-step process: First, tagging with a highly accurate coarse-grained tagset and, then in a second step, refining the results using a more fine-grained tagset (Steingrímsson et al., 2019).

- We created a tool, CombAlign, for generating more accurate word alignments. The aligner runs an ensemble of available word alignment tools and models and uses a voting system to select the most likely alignments. Our tool outperforms previously available tools when they are run individually, obtaining higher F_1-scores for multiple language pairs (Steingrímsson et al., 2021a).

- We devised an approach, PivotAlign, for inducing dictionaries using a combination of word alignments over parallel corpora and pivoting through available dictionaries. Our approach scored highest in the Translation Induction Across Dictionaries (TIAD) shared task in 2021 (Steingrímsson et al., 2021c).

- We published a new version of ParIce, a previously available English–Icelandic parallel corpus (Barkarson and Steingrímsson, 2019), with better alignments and filtering.

- We compiled an English–Icelandic bilingual lexicon, containing over 230,000 equivalent pairs. Automatic methods, including the ones introduced at TIAD 2021, were used to compile candidate lists that were then manually evaluated (Steingrímsson et al., 2022)

Our contributions in relation to aligning and filtering parallel corpora, and mining comparable corpora, can be summarised as follows:

- We manually inspect the effectiveness of different scoring mechanisms for identifying erroneous sentence pairs in parallel corpora and show that, in order to obtain optimum results when filtering, the threshold for these scores should be set depending on the dataset being filtered.

- We compare various filtering approaches and show that different filters should be selected depending not only on the dataset, but also on translation direction.

- We show that by filtering the ParaCrawl corpus (Bañón et al., 2020) using a different approach for each translation direction, adapted to that translation direction, its usefulness can be improved, at least for the translation directions we experiment with.

- We introduce a new sentence aligner, SentAlign, capable of extensively exploring the possible alignment combinations in pairs of fairly large documents in order to find the best alignments. It uses Language-agnostic BERT sentence embedding (LaBSE) (Feng et al., 2022) for scoring, outperforming previous aligners on a popular evaluation set.

- We compare available alignment tools and evaluate their output on two evaluation sets, one of which was compiled by us for the English–Icelandic language pair. Furthermore, we manually inspect their outputs and evaluate them in a downstream MT task. We find that while SentAlign most often outperforms other aligners, MT-based and lexicon-based approaches also give strong results.

- We experiment with a Cross-language Information Retrieval (CLIR)-based approach for mining parallel sentences from comparable corpora, using a classifier employing word alignment-based scores and sentence embeddings to select the best candidate pairs proposed by the CLIR tool. We use this approach to show the potential of exploiting data usually discarded in the MT training data compilation process.

- We show that by combining the best alignment and filtering approaches for compiling parallel corpora we can significantly increase the quality of MT models trained on that corpora, as measured both in terms of automatic metrics and by manual annotation.

1.5 Structure of the Thesis

In Chapter 2, we start by outlining the background of our work. We review prior work, relevant to our research, introduce important concepts used and provide a brief introduction to the most popular variants of neural MT (NMT) used at the time of writing. We also give a brief description of the state of MT and natural language processing (NLP) in general for English–Icelandic.

Chapter 3 describes our work on supporting tools and data sets developed in order to aid our research on compiling better training data for MT. We describe our work on ABLTagger (Augmented Bi-directional LSTM Tagger), the PoS tagger we developed for Icelandic; CombAlign, a tool for acquiring more accurate word alignments; PivotAlign, a tool we built for helping with inducing bilingual lexicons and our scoring mechanism based on word alignments, which we call WAScore. We also describe ParIce, an English–Icelandic parallel corpus we realigned and filtered and the process of compiling a bilingual lexicon we published.

Chapter 4 describes the different filtering methods we applied. We evaluate the scoring methods used as well as the filtering mechanisms themselves. We report on the results of both manual and automatic evaluations of the methods.

Chapter 5 describes different approaches to sentence alignment, our evaluation of the different approaches and the design and development of our own sentence aligner, SentAlign. We also report on the accuracy of the different approaches using three evaluation methods: measuring F_1-scores on sentence alignment evaluation sets, manual annotation of the output of the aligners, and the effect on translation output in a downstream MT task.

In Chapter 6, we describe our work on mining parallel sentence pairs from comparable corpora. We also show how we can improve an English–Bengali training set by selecting only the best sentence pairs and then split up the other pairs in the training corpus in order to try to extract from them the best matching subsegments. We then use this approach to segment data discarded from our English–Icelandic corpus during the alignment and filtering steps, and then mine the collection of segments for parallel pairs, using the same approaches applied to comparable corpora.

Chapter 7 describes our manual evaluation of MT models trained on data compiled using the approaches described in the previous chapters and compares these models to models trained on data compiled using previous approaches. We also compare the output of models trained using Transformer$_{\mathrm{BASE}}$ (Vaswani et al., 2017) and by fine-tuning mBART (Liu et al., 2020b).

Finally, conclusions are given in Chapter 8 where we summarize our findings and point out possible future directions for our work.

1.6 Publications

Some of the findings presented in this thesis have been published in peer-reviewed conference proceedings and workshops:

1. Steinþór Steingrímsson, Örvar Kárason and Hrafn Loftsson. 2019. Augmenting a BiLSTM Tagger with a Morphological Lexicon and a Lexical Category Identification Step. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1161–1168, Varna, Bulgaria.

2. Steinþór Steingrímsson, Hrafn Loftsson and Andy Way. 2020. Effectively Aligning and Filtering Corpora under Sparse Data Conditions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 182–190, Online.

3. Haukur Páll Jónsson, Haukur Barri Símonarson, Vésteinn Snæbjarnarson, Steinþór Steingrímsson and Hrafn Loftsson. 2020. Experimenting with Different Machine Translation Models in Medium-Resource Settings. In *Text, Speech, and Dialogue: 23rd International Conference, TSD 2020, Proceedings*, pages 95–103, Brno, Czech Republic.

4. Steinþór Steingrímsson, Hrafn Loftsson and Andy Way. 2021. CombAlign: a Tool for Obtaining High-Quality Word Alignments. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 64–73, Reykjavik, Iceland (Online).

5. Steinþór Steingrímsson, Pintu Lohar, Hrafn Loftsson and Andy Way. 2021. Effective Bitext Extraction From Comparable Corpora Using a Combination of Three Different Approaches. In *Proceedings of the 14th Workshop on Building and Using Comparable Corpora (BUCC 2021)*, pages 8–17, Online.

6. Steinþór Steingrímsson, Hrafn Loftsson and Andy Way. 2021. PivotAlign: Leveraging High-Precision Word Alignments for Bilingual Dictionary Inference. In *Proceedings of the Workshops and Tutorials held at LDK 2021*, pages 190-199, Zaragoza, Spain.

7. Steinþór Steingrímsson, Luke O'Brien, Finnur Ingimundarson, Hrafn Loftsson and Andy Way. 2022. Compiling a Highly Accurate Bilingual Lexicon by Combining Different Approaches. In *Proceedings of Globalex Workshop on Linked Lexicography within the 13th Language Resources and Evaluation Conference*, pages 32–41, Marseille, France.

8. Steinþór Steingrímsson, Hrafn Loftsson and Andy Way. 2023. Filtering Matters: Experiments in Filtering Training Sets for Machine Translation. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, Tórshavn, Faroe Islands.

Chapter 2

Background

Machine translation (MT) is the subfield of natural language processing (NLP) that focuses on automatically translating text from one natural language to another. The goal is to build systems that can perform that task accurately and efficiently, outputting fluent and well-formed text in a target language that preserves the meaning of a source language text, without the need of human intervention.

Approaches to MT can broadly be divided into three paradigms: rule-based MT (RBMT), statistical MT (SMT) and neural MT (NMT). RBMT typically uses a set of grammatical rules to analyze the source language input text, breaking it down into smaller units, such as words, phrases or sentences. These units are then matched against transformation rules and rules for the target language as well as dictionaries to produce an output. The first RBMT system was demonstrated in 1954 (Dostert, 1955) and RBMT was the ruling paradigm until the late 1980s. While more recent approaches outperform RBMT systems for most language pairs, rule-based systems are still being developed, mostly for low-resource languages (Forcada et al., 2011; Pirinen et al., 2017; Khanna et al., 2021).

In SMT, information is extracted from a bilingual corpus (made up of sentence aligned documents written in two different languages) to find word or phrase equivalences. A word alignment tool is used to find pairs of words or phrases in the source and target languages. A translation model is derived from these alignments and a language model is based on word sequences in the target language. Translation is determined by probabilities using Bayes' rule, shown in Equation (2.1):

$$p(t|s) = \frac{p(s|t)p(t)}{p(s)} \tag{2.1}$$

In the equation, $p(t|s)$ is the probability of a translation t, given a source sentence s. $p(s|t)$ is the probability of the source sentence given the translation and $p(t)$ and $p(s)$ are the probabilities of the translation and source segments occurring, respectively, given the language model (LM). When looking for a t that maximizes the right side of the equation, $p(s)$ is always the same for the source sentence and thus unnecessary, so an LM for the source language is not needed. This approach is called the noisy-channel model and was put forward by Brown et al. (1993) for word-based translations. Variations of SMT emerged in the following years, with research in phrase-based SMT (PBSMT) (Koehn et al., 2003), depicted in Figure 2.1. This was facilitated by open source toolkits such as Moses (Koehn et al., 2007) in the 2000s and 2010s.

In the standard phrase-based model, a phrase translation table is learned from the parallel corpus, by extracting small word sequences based on word-alignment results and calculating relative frequency for each phrase pair based on how many sentence pairs contain it. A

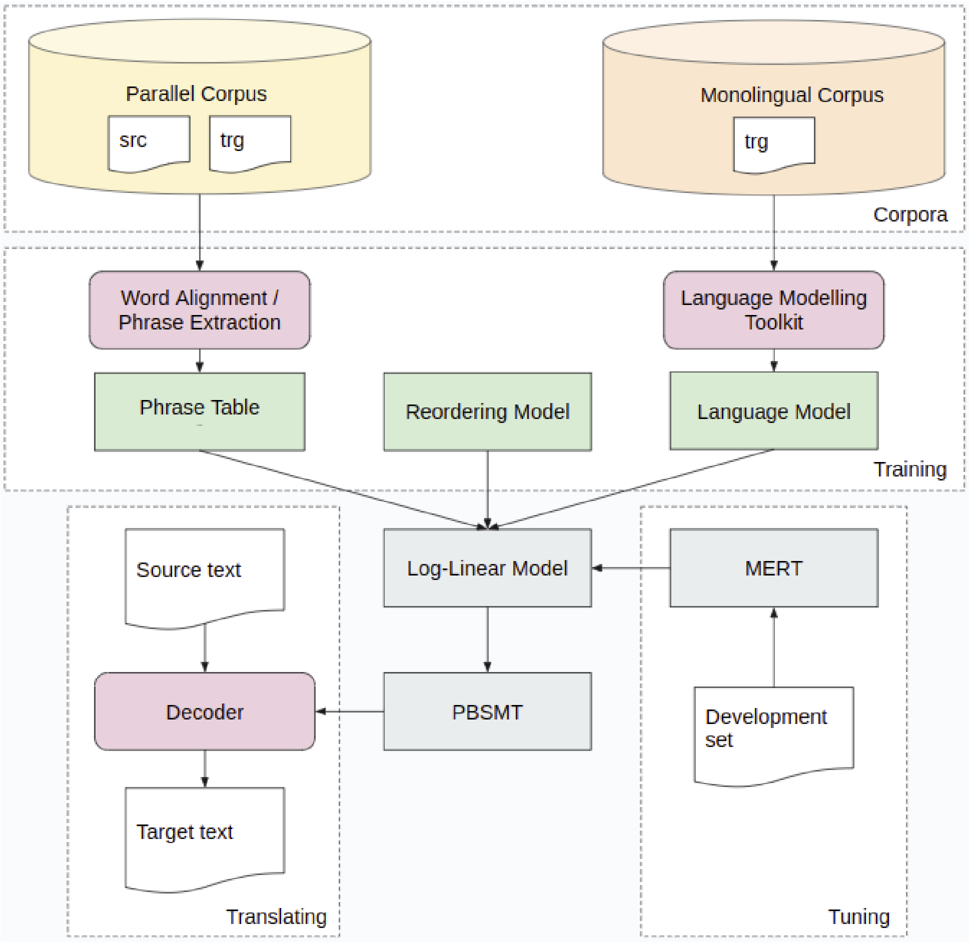


Figure 2.1: The basic architecture of a PBSMT model. The figure is adapted from (Liu et al., 2021).

distance-based reordering model handles reordering of phrases. The reordering distance is the number of words skipped relative to the previous phrase. The reordering cost is based on a decaying cost function, and not learned from data. Finally, an n-gram language model, based on a target language monolingual corpus is used to help with building a fluent target language output. These three components work together to build the optimal translation using a log-linear model. In this framework, sentence translations are viewed as a vector of features and the model a set of feature functions, trained separately and combined assuming they are independent of each other. The log-linear model has the following form:

$$p(e|f) = \exp \sum_{i=1}^{n} \lambda_i h_i(e, f) \tag{2.2}$$

In equation (2.2), n stands for the number of feature functions, in this case $n = 3$. Feature function $h_1 = log\phi$, where ϕ stands for the phrase translation table. Feature function $h_2 = logd$, where d stands for the reordering model, and feature function $h_3 = logp_{\text{LM}}$, standing for the language model. λ_1, λ_2 and λ_3 stand for the feature weights. To obtain better performance, the weights of each feature of the model can be tuned on a separate development data set, to maximize translation performance, usually as measured with BLEU (Papineni et al., 2002). Minimum error rate training (MERT) (Och, 2003) is most often used for parameter tuning. One of the reasons for using this model is that it allows for the weighting of the different model components, which may lead to improvements in translation quality. Another reason for using this model is that it allows for adding additional model

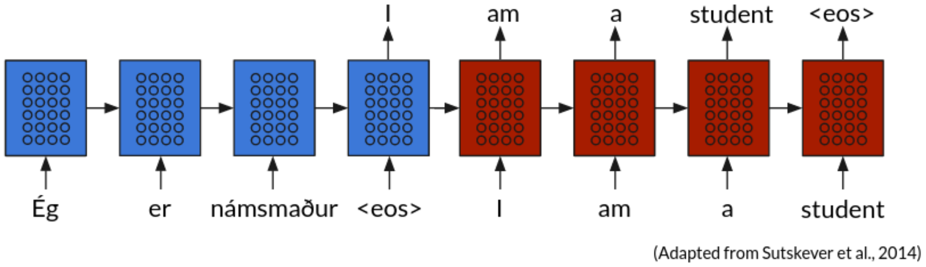


Figure 2.2: The sequence to sequence model proposed by (Sutskever et al., 2014) reads an input sentence and embeds it in vector space. After processing <eos> at the end of the input sentence, the recurrent state holds the embedding for the entire input sentence, which is used by the decoder to generate the output sentence in the target language.

components in the form of feature functions, such as bidirectional translation probabilities (Koehn, 2009). Finally, to find the best translation for a given sentence, a decoding algorithm uses the trained model to search for the best translation using a heuristic search. The system can consult the phrase table to look up all translation options that can apply to the input sentence and then incrementally compute the probability of a translation from left to right by testing out different hypotheses.

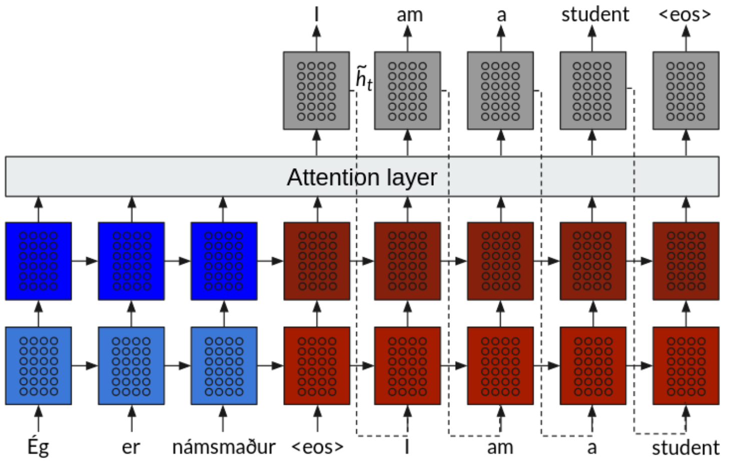
The 2010s saw the rise of a new paradigm, NMT, discussed in the following section.

2.1 Neural Machine Translation

As opposed to SMT, which use text-based models for predicting translations, NMT models represent source and target language text, as well as internal states, as vectors. They have no separate translation models and language models, and are trained end-to-end on raw input (source) and output (target) text sequences.

In the shared translation task at the 2016 Conference on Machine Translation (WMT), for the first time the best-performing systems in terms of BLEU score for some high-resource language pairs were NMT models (Bojar et al., 2016). Since then, it has been shown that NMT can also outperform SMT on low-resource language pairs (Sennrich and Zhang, 2019).

The history of neural networks-based MT research goes as far back as the 1980s when Allen (1987) experimented with using back-propagation for translation, and to the work by Chalmers (1992), Chrisman (1991) and Forcada and Ñeco (1997) in the 1990s, who used two feed-forward neural networks, an encoder and a decoder, to learn internal representations of input sentences that can be decoded to obtain a corresponding output string. With improvement in computing power and an increase in available data, the current wave of applying neural networks to MT emerged in 2013 and 2014 with the work of Kalchbrenner and Blunsom (2013), Sutskever et al. (2014), Cho et al. (2014) and Bahdanau et al. (2015). Kalchbrenner and Blunsom (2013) used a model based on convolutional neural networks (CNN) for encoding the input in a vector space and recurrent neural networks (RNN) for generating the output. Cho et al. (2014) proposed a model consisting of two RNNs, one for the encoder and the other for the decoder, while Sutskever et al. (2014) used long short-term memory (LSTMs) for encoding and decoding sequences of text (represented in Figure 2.2), showing that their architecture could solve sequence-to-sequence problems and learn representations that are sensitive to word order, benefitting from the LSTM's ability to successfully learn on data with long range temporal dependencies. Bahdanau et al. (2015) based



(Adapted from Luong et al., 2015)

Figure 2.3: The input-feeding approach proposed by (Luong et al., 2015b). The attentional vectors $\tilde{h}_t$ are concatenated with inputs at the next time step. This results in the model being aware of previous alignment choices and it creates a deep network spanning both horizontally and vertically.

their architecture on RNNs, both for encoding and decoding, but they also apply an attention mechanism in the decoder which helps the decoder decide which parts of the source sentence to pay attention to. The attention value is computed by a feed-forward layer that takes the previous hidden state and each input word embedding as inputs. They, like Sutskever et al. (2014), report BLEU scores for an English→French translation task comparable to those achieved by SMT. Improvements were made on the attention mechanism, e.g. by Luong et al. (2015b), represented in Figure 2.3, who proposed two classes of attention, a global one and a local one, which respectively attend to all source words or only a limited window of words at a time.

Vaswani et al. (2017) introduced the transformer architecture, represented in Figure 2.4, based solely on attention with no recurrence or convolutions. In particular, this influential paper[1] introduced using self-attention for MT. Self-attention extends the idea of attention to the encoder; instead of searching for alignment between input words and output words, it refines the representation of input words with respect to other relevant words in the input string (Koehn, 2020). The transformer models require significantly less time to train than previous models, while still improving output quality, which has made them the dominant NMT architecture. Vaswani et al. (2017) presented a few variations on the architecture, based on the hyperparameters chosen for training. In the experiments presented in this thesis, we employ the Transformer$_{\text{BASE}}$ model, detailed in Section 3.5.

Various improvements have been made to the transformer models, such as altering the depth of the models and by allowing for translation of larger contexts. Examples of that include Junczys-Dowmunt (2019), who uses document boundaries in his training data to mark out long text sequences for training the models, Dehghani et al. (2019) who propose a variant that, instead of having a fixed number of layers, has variable depth, and Liu et al. (2020a), who build transformer-based models with very many layers, up to 60 encoder layers and 12 decoder layers.

[1] At the time of writing this thesis, Vaswani et al. (2017) have more than 70,000 citations, according to Google Scholar.

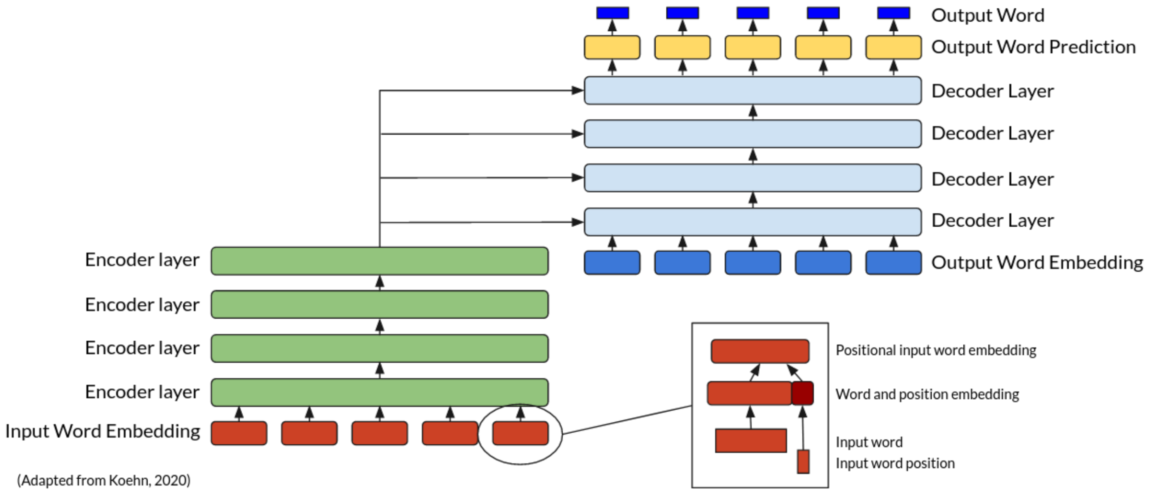


Figure 2.4: The full attention-based transformer model. This representation is adapted from Koehn (2020). The input words are represented as a vector representation of both the word itself and its position. In the encoder layer, a self-attention mechanism is used to first mix the word embedding with the embeddings of related words in the input, found by calculating the dot product of the word embedding and the embeddings of all words in the sentence. Then this representation is refined with a feed-forward layer. The decoder layer also has self-attention, but adds attention, computed between the decoder states and the final encoder states, before refining the representation again.

2.1.1 Representing Tokens in an NMT Model

When training NMT models, source and target sentences are represented as a sequence of tokens. As the size of the vocabulary is limited due to computational constraints, usually to tens of thousands of different tokens, representing word forms as separate tokens would imply limiting the vocabulary only to the most frequent words, creating a large OOV-problem. It would also mean treating the word forms as distinct and unrelated, even though they may only be different inflectional forms of the same word, exacerbating greatly the OOV-problem for MRLs.

To address this, Sennrich et al. (2016b) proposed segmenting input to NMT models using byte pair encoding (BPE) (Gage, 1994), allowing for representing an open vocabulary through a fixed-size vocabulary of variable-length sequences of subword units. Unseen and rare words can then be represented by sequences of these subword units, while frequent words often get their own tokens. Kudo and Richardson (2018) presented SentencePiece, a language-independent subword tokenizer and detokenizer which tokenizes and then converts text into an id sequence, simplifying the process of building end-to-end systems, trained on subword units.

2.2 Large Language Models

Fine-tuning large language models (LLMs), which means updating their weights by training them in a supervised[2] manner on a dataset with examples for the intended task, have been shown to produce significant performance gains for a range of MT tasks. mBART25 (Liu et al., 2020b) is an LLM trained on 25 languages. For some language pairs, by fine-tuning the

[2] Supervised MT training requires sentence pairs from a parallel corpora for the system to learn how to perform its task.

model it can outperform MT models trained from scratch on the same parallel data (Liu et al., 2020b), in terms of BLEU. The model can also generalize to unseen languages, although to a different extent depending on the language. The authors hypothesize that this is due to the pre-trained transformer layers learning universal properties of language, even without much lexical overlap with the languages in the training set.

Generative Pre-trained Transformer (GPT) models have been shown to be able to translate in zero-shot to few-shot scenarios. Zero-shot means that the model is told to translate without being shown any translation examples, but in few-shot scenarios the model is given a few demonstrations of the task at hand. In either case, the model weights are not updated. Brown et al. (2020) show that GPT-3, the third-generation of models in the GPT series, outperforms previous unsupervised[3] NMT work when translating into English, which reflects its strength as an English LM. When translating into other languages, the model does not perform as well. In another study (Hendy et al., 2023), GPT-3 was shown to be able to reach reasonable accuracy for some language pairs on evaluation datasets from WMT shared translation tasks, as measured by automatic metrics, especially when translating into English, although in most cases the results lagged substantially behind the best WMT results for these same datasets. The evaluation sets used were from WMT21 and WMT22. While the WMT22 datasets should not overlap with the GPT models training data, collected until June 2021, the WMT21 might, which could have affected the results of the paper.

2.3 Compiling Training Data for MT

In Section 1.2, we defined noise as segments that may degrade the performance of an MT system trained on that data. As NMT systems are sensitive to noise (Khayrallah and Koehn, 2018), various efforts have been made in recent years to improve the quality of the data, i.e. minimize noise. The training data are compiled from parallel corpora, which are collections of documents in one language paired with their translations into another. In compiling the training data, the parallel corpora are usually first aligned on a sentence level, and then the sentence pairs that are likely to be detrimental to the training process are filtered out. Alignment and filtering are discussed in Sections 2.3.1 and 2.3.2. Comparable corpora are collections of documents in two or more languages that share similar content, as defined in Section 1.2. In Section 2.3.3, we discuss the main issues in mining such corpora for parallel sentence pairs.

2.3.1 Sentence Alignment

The objective of sentence alignment (represented in Figure 2.5) is to find parallel sentences in aligned documents, ideally all semantically equivalent sentence pairs without any misalignments and extraneous or missing data in either language. When sentence alignment is incorrect or inaccurate, everything that uses the aligned parallel corpora will be less reliable, be it MT systems or word alignment tools used for inducing bilingual dictionaries (the focus of Section 3.6). It is therefore important to select the best available algorithm for a given dataset to be aligned.

While a wide variety of approaches have been used for automatic sentence alignment over the last 30 years, the predominant ones have been statistical methods using length-based features (Brown et al., 1991; Gale and Church, 1991), lexicon-based techniques (Varga

[3] Unsupervised MT does not require training data in two languages, but instead relies on monolingual data in both languages and learns representations from them.

et al., 2005), methods using MT (Volk et al., 2010) and, most recently, multilingual sentence embedding-based approaches (Thompson and Koehn, 2019).

All of these approaches have their merits, but also some possible disadvantages. The length-based methods make the assumption that a set number of characters in one language give rise to a set number of characters in another and that the languages are proportional to each other. This can work reasonably well for related languages, but some research have shown the assumption to be less likely to hold for unrelated languages, especially those using different writing systems (Wu, 1994; Martin et al., 2003; Samy et al., 2006). Lexicon-based approaches can use simple statistical methods or word alignment to bootstrap a bilingual dictionary if it is not available, but for best results, it is important to have access to good external dictionaries. Bilingual dictionaries usually only contain word lemmas and thus good lemmatizers or stemmers are necessary for them to be useful. MT-based approaches require a pre-existing good quality MT system, and sentence embedding-based systems require cross-lingual sentence embeddings, trained on data in the languages to be aligned, for the results not to be significantly deteriorated (Chimoto and Bassett, 2022). All of these resources are not often available and some or all are lacking for many language pairs.

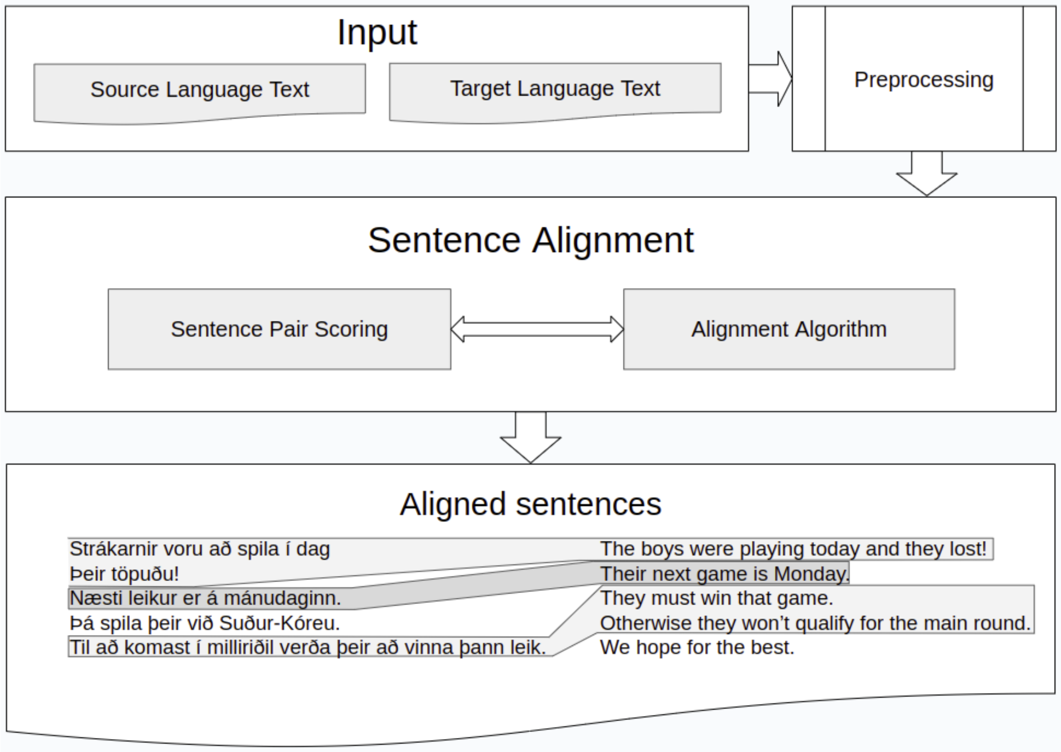

Input
Source Language Text
Target Language Text
Preprocessing
Sentence Alignment
Sentence Pair Scoring
Alignment Algorithm
Aligned sentences
Strákarnir voru að spila í dag
Þeir töpuðu!
Næsti leikur er á mánudaginn.
Þá spila þeir við Suður-Kóreu.
Til að komast í milliriðil verða þeir að vinna þann leik.
The boys were playing today and they lost!
Their next game is Monday.
They must win that game.
Otherwise they won't qualify for the main round.
We hope for the best.

Figure 2.5: A sentence alignment system takes in parallel documents in two languages. Preprocessing usually involves sentence splitting and sometimes cleaning the text of non-linguistic content. The sentence alignment module needs to have a way of scoring how good a given alignment is, and a way of finding the optimal set of alignments between two documents. After running all processing steps, the system then outputs a set of aligned sentences. For successful alignment, the system needs to be able to perform a variety of functions: Contraction ($n \leftrightarrow 1$) and expansion ($1 \leftrightarrow n$), in which one sentence aligns to more than one sentence in the other language, deletion ($1 \leftrightarrow 0$) and insertion ($0 \leftrightarrow 1$), where a sentence does not align to any sentence in the other language, and substitution ($1 \leftrightarrow 1$) where a sentence aligns to exactly one sentence in the other language. Most alignment systems also allow mergers ($n \leftrightarrow m$), in which multiple sentences in both languages are merged before alignment.

In Chapter 5, we will provide a more detailed account of relevant work in this area, and describe the most common approaches and tools in more detail.

2.3.2 Filtering

Incorrect translations, as defined in Section 1.2, untranslated target text, misalignments, and other noisy segments in a parallel corpus have been shown to have a detrimental effect on the output quality of NMT systems trained on that corpus (Khayrallah and Koehn, 2018), as measured using BLEU. In recent years, machine learning (ML) research has generally focused more on creating better models, rather than better datasets, where a focus on benchmarking model performance prompts researchers into adapting the largest existing datasets without fully considering fidelity to the underlying problem the model should solve (Mazumder et al., 2022). The effectiveness of ML models, however, depends on both algorithms and data. Aroyo et al. (2022) argue that as the datasets define the entire world within which models exist and operate, more work is needed on how the data can be optimized for more effective use. Filtering parallel data for MT is the task of removing possible detrimental segments from the data used for training MT models. Filtering is usually carried out by using a set of rules, often accompanied with scoring and/or classifying sentence pairs, to remove the segments with the lowest perceived quality.

WMT hosted annual shared tasks on parallel corpus filtering for three years, 2018–2020 (Koehn et al., 2018, 2019, 2020). There, methods based on crosslingual sentence embeddings trained from parallel sentence pairs did well (e.g. Chaudhary et al. (2019) and Artetxe and Schwenk (2019a)). Two versions of Bicleaner (Sánchez-Cartagena et al., 2018; Esplà-Gomis et al., 2020) were submitted, both times ranking highly. Zaragoza-Bernabeu et al. (2022) introduced a third version, Bicleaner-AI, implementing a neural classifier based on pre-trained transformer-based language models fine-tuned on a binary classification task. In Chapter 4 we discuss these systems further, as well as giving a more detailed account of relevant work. We experiment with different versions of Bicleaner as well as various scoring and classification approaches and try to compare their merits using both manual and automatic evaluation methods.

2.3.3 Mining Comparable Corpora

Comparable corpora are more abundant than parallel texts. They have been shown to be a useful source for mining parallel segments that can be useful as additional training data for MT systems (Wolk et al., 2016; Hangya and Fraser, 2019). This is important in low-resource scenarios, where parallel corpora are scarce. In contrast to parallel corpora, where it can be assumed that the sentence order in two parallel texts is the same, potential parallel sentence candidates in comparable corpora can come from anywhere in two comparable documents. This means that the search space becomes very large as the comparable documents grow in size, thus necessitating methods for reducing the search space and finding parallel sentence candidates effectively.

Considerable work has been carried out on developing methods for extracting parallel sentences or phrases from comparable corpora. Numerous approaches have been proposed: lexicon-based (Zhao and Vogel, 2002), MT-based (Sarikaya et al., 2009), and using bilingual word embeddings, as in the most prominent systems in the 2017 and 2018 shared tasks on identifying parallel sentences in comparable corpora in the Workshop on Building and Using Comparable Corpora (BUCC) (Zweigenbaum et al., 2017, 2018).

In Chapter 6, we present our approach using a cross-lingual information retrieval (CLIR) tool, both for mining traditional comparable corpora and for mining data usually discarded in the parallel corpora alignment process. There, we also discuss related work in more detail.

2.4 Assessing Quality in Machine Translation

There is a range of different use cases for which MT is utilized. Common scenarios include: gisting, i.e. helping readers find the most important points in a text written in languages they can't read; various uses in relation to tourism, both for tourists when communicating with people that do not share a common language with them and when trying to understand words or short texts in a foreign languge, as well as for service providers looking to provide information in multiple languages; professional translators using MT to become more productive by post-editing MT-generated translations; and the output of MT systems may even be useful on its own for various domain-specific scenarios.

To understand the strengths and weaknesses of a given MT system we need to be able to assess its quality. A variety of different metrics have been introduced. Subjective human judgements have commonly been measured along two dimensions: 1) "adequacy", measuring the extent to which a translated text contains the same information as the source text, and 2) "fluency", the extent to which the sentence is well-formed in the target language (Way, 2018). These are traditionally evaluated on a graded scale, e.g. from 1–5. Another approach is to rank two or more systems on the sentence level by human evaluators, giving statistics on which system is preferred over the others. More recently, human evaluation in evaluation campaigns, such as for the WMT News Translation shared tasks, has used "direct assessment" (Graham et al., 2016). Using this approach, a single translated sentence is evaluated at a time, using a 100-point scale. The judgments on this scale can more easily be normalized than judgments on the scale of 1–5, which is important as subjective biases of individual human evaluators can skew the results. In recent years, more fine-grained approaches have been employed for evaluating MT output. Multidimensional Quality Metrics (MQM) (Lommel et al., 2014) is a framework for analytic translation quality evaluation. When using MQM, an evaluator identifies errors in a translation and classifies them. If applied successfully, the approach generates data that reduces subjectivity and enhances comparability. The data can then be used both to identify what kind of quality issues an MT system has and to compare the strengths and weaknesses of different systems.

While human evaluation has obvious benefits – it shows how MT users are likely to perceive the output – it is also expensive, slow and can be inconsistent. Automatic metrics designed specifically for MT evaluation were introduced at the start of the century, as SMT was becoming the ruling paradigm in MT. BLEU and NIST (Doddington, 2002) both ignored the source sentence and used n-grams to compute similarity between a reference and the MT output. For developers building MT systems, a fast, automatic system that can tell them whether the system improves after a change is highly useful and, as these approaches seemed to serve that purpose without any need for human evaluation, they were quickly adopted. BLEU, which became the standard for measuring MT quality, has since early on been criticized for various shortcomings. Callison-Burch et al. (2006) show that BLEU has a rather low correlation with human judgment and that an improvement in BLEU does not necessarily reflect a genuine improvement in translation quality, as perceived by human evaluators. Other important aspects of BLEU that have been criticized include that BLEU ignores the relative relevance of different words; that it does not address grammatical coherence beyond n-grams; and that the actual BLEU scores have no meaning in themselves but depend on multiple different factors, including the reference translations and their quality, the number of reference translations, the language pair, domain and even the tokenization scheme (Koehn, 2020). METEOR (Banerjee and Lavie, 2005) introduces the idea of using stemming and synonyms. It counts the number of word matches between the system output and a reference, first matching surface forms of words and then backs off to stems and finally to using WordNet (Miller, 1994) to match semantic classes. This has the drawback that stemmers and synonym databases are required and thus it cannot easily be applied to all languages. Translation error rate (TER) (Snover et al., 2006) measures how much editing a human would have to perform to match a reference translation. It inspects addition, deletion and substitution of words as well as shifts of word sequences. In comparison to BLEU, TER scores more accurately score a single sentence, but as the problem of calculating edit-distance with a move operation is NP-complete, calculating the score can be computationally expensive. Popović (2015) proposes using a character n-gram F-score for automatic evaluation, when introducing chrF, showing that it has higher correlation with human judgments than BLEU, TER and METEOR, and that it is language- and tokenisation-independent. While BLEU, TER, METEOR and chrF have been some of the more popular evaluation metrics, recent Metrics Shared Tasks at WMT (Freitag et al., 2021b, 2022) have shown neural-based learned metrics to better correlate with human evaluation. Multiple different neural-based metrics have been introduced. We employ one of these, COMET-22 (Rei et al., 2022) in Chapter 7, where we evaluate the final MT models trained in our experiments.

The metrics discussed in this section are only few of many metrics that have been developed for assessing MT quality. They are some of the most commonly used, and despite the widely acknowledged issues with BLEU, it remains the primary measure of translation quality, not least with developers of MT systems.

2.5 Machine Translation for Icelandic

In this thesis, we will mostly be working with the English–Icelandic language pair. Before the commencement of this Ph.D. project in 2018, development in MT for Icelandic had been limited. In the META-NET White Paper Series from 2012, the report on Icelandic (Rögnvaldsson et al., 2012) mentions a rule-based system, now defunct, offering translations between Icelandic and three languages, English, Danish and Esperanto, as well as an Apertium-based (Forcada et al., 2011) Icelandic–English system (Brandt et al., 2011).

A parallel corpus, ParIce (Barkarson and Steingrímsson, 2019), was compiled and published in 2018. ParIce is partly a collection of previously available parallel data, which has been realigned and filtered, as well as new parallel data with the largest source being regulatory texts published in relation with the European Economic Area (EEA) agreement. The original work on ParIce was mainly carried out by myself in cooperation with Starkaður Barkarson, prior to starting the work described in this thesis.

Jónsson et al. (2020) was the first published work describing SMT and NMT for Icelandic. It compares a PBSMT model trained using Moses (Koehn et al., 2007) and two NMT models: a sequence-to-sequence model as described in Sutskever et al. (2014) and a Transformer$_{\text{BASE}}$ model (Vaswani et al., 2017). I participated in this paper as part of this Ph.D. project.

English–Icelandic was one of the language pairs of the shared news translation task at WMT 2021 (Akhbardeh et al., 2021). Multiple teams participated, with a team from Facebook achieving the highest scores (Tran et al., 2021). They submitted a multilingual system employing backtranslation (Sennrich et al., 2016a), in-domain fine-tuning, averaging model parameters across the last five checkpoints, noisy channel re-ranking and post-processing of punctuation. Two of the other systems submitted to the shared task were the Allegro.eu submission (Koszowski et al., 2021), based on the Transformer$_{\mathrm{BIG}}$ architecture, and Miðeind's[4] submission (Símonarson et al., 2021) based on mBART. Due to the similarity of their models to ours, we will be comparing our results to theirs in Chapter 7.

A National Language Technology Programme for Icelandic (NLTPI) started in 2019 (Nikulásdóttir et al., 2017; Nikulásdóttir et al., 2020; Nikulásdóttir et al., 2022). That programme included building MT models and compiling datasets for training MT systems. This Ph.D. project was in part funded by that programme and some of the work on supporting tools and data, described in Chapter 3, resulted from cooperation within the programme, as detailed in the relevant sections.

Most NLP tools and language resources useful for building Icelandic resources for MT were of limited quality before the start of the NLTPI. The best PoS-tagger available (Loftsson and Östling, 2013) still had ample room for improvement, and, at the beginning of this Ph.D. project, we built a new tagger outperforming the previous one by a large margin (see Section 3.1). That tagger was then further improved upon within the NLTPI. A lemmatizer was available, accurately predicting lemmas given the correct PoS-tags (Ingólfsdóttir et al., 2019). In terms of data, the first version of the Icelandic Gigaword Corpus was published in 2018 (Steingrímsson et al., 2018), a large database of Icelandic inflection had been in development for over 15 years (Bjarnadóttir, 2012), and the first version of the previously mentioned ParIce corpus had been published. All of these projects were further developed within the NLTPI, thus improving in size and quality as the project described in this thesis developed. Furthermore, a small English–Icelandic dictionary was available, created as a part of the Apertium project. For some of our experiments, we wanted to use a larger dictionary and, in Section 3.6, we describe how we compiled a reasonably large bilingual dictionary using bilingual lexicon induction (BLI) methods.

[4] A private software company focusing on NLP for Icelandic.

Chapter 3

Supporting tools and data

In order to carry out the intended experiments for the English–Icelandic language pair, some tools and datasets needed to be built, as the resources for Icelandic were either unavailable or inadequate when we started our project. Available parallel corpora that included Icelandic were distributed between multiple sources until the first version of the ParIce corpus was published in 2018. The parallel data had been aligned and filtered along with other language pairs, using generic methods without any regard to Icelandic in particular. Icelandic datasets for use with various aspects of language technology and tools to process Icelandic have progressed substantially during the period this thesis is written in, which largely coincides with the NLTPI, discussed in Section 2.5. Some of the tools and data discussed in this chapter stem from the NLTPI and were either built by myself or by me in cooperation with others. When the work was done in cooperation with others, my contribution is described in a footnote.

In this chapter, we first describe a part-of-speech (PoS) tagger which significantly raised the accuracy for Icelandic PoS tagging (Section 3.1). Second (in Section 3.2), we describe a word alignment tool, which uses an ensemble of word aligners to acquire word alignments that obtain higher F_1-scores than those produced by the best aligners in the ensemble. Third, we describe an approach to induce a bilingual lexicon using dictionary pivoting in combination with word alignments on a parallel corpus (Section 3.3). Fourth, we describe a scoring mechanism based on word alignments (Section 3.4). Fifth, we discuss the compilation of two versions of an English–Icelandic parallel corpus (Section 3.5). Finally, we describe how a large English–Icelandic lexicon was induced and evaluated (Section 3.6).

3.1 ABLTagger

Tagging and lemmatizing Icelandic texts with as much accuracy as possible are necessary for multiple aspects of our work, including building a bilingual lexicon (Section 3.6), training a lemmatized Bicleaner model (Section 4.2.2), sentence alignments (Chapter 5), and more. For tagging Icelandic texts, the most accurate tagger previous to our work was IceStagger (Loftsson and Östling, 2013), which obtained an accuracy of 93.84% using an averaged perceptron method.

Bidirectional long short-term memory (BiLSTM) models have been shown to be effective for various sequential labelling tasks, including PoS tagging (Ling et al., 2015; Plank et al., 2016). They are an extension of general LSTMs (Hochreiter and Schmidhuber, 1997) that perform better on sequences where the complete input sequence is available. Two LSTMs are trained on the input sequence, one on its natural reading order and the other on its reverse (Graves and Schmidhuber, 2005). Santos and Zadrozny (2014) were first to join word embeddings, vector representations of words based on their context in training data, with character embeddings when tagging with BiLSTMs. In our model, both word embeddings and recurrent character embeddings are used as input. For each word, both forward and backward expressions are generated, containing the sequence of characters in the word, as well as word initial and word final markers. The character embeddings for a given word are input into a BiLSTM. The output from the BiLSTM is concatenated to the word embedding. This helps the model grasp morphological details.

In an effort to raise the accuracy of Icelandic PoS tagging, we developed a new tagger, ABLTagger[1] (Steingrímsson et al., 2019) using a BiLSTM model which we augmented with the Database of Modern Icelandic Inflection (DMII) a morphological lexicon (Bjarnadóttir, 2012; Bjarnadóttir et al., 2019) and a lexical category identification step. In our work on the tagger, presented in Steingrímsson et al. (2019), we evaluated three models. First, we confirmed the effectiveness of a BiLSTM model for PoS tagging using a fine-grained tagset. Second, we supplemented the base model with an external morphological lexicon by encoding the morphological features for each word as an n-hot vector and concatenating it to the word and character embeddings input into the model, thereby obtaining SOTA results. Third, we proposed an approach to further increase the accuracy by creating a coarse-grained tagset from the fine-grained one and using the resulting tagset to devise a two-step process. Specifically, we trained a separate model on only the lexical category and used the coarse-grained output tag as an input into the main model. This approach was, to the best of our knowledge, novel in the context of neural network tagging. Combined, this resulted in an overall tagging accuracy of 95.15%, which is equivalent to an error reduction of 21.3% compared to the previous state of the art.

3.1.1 The Three Models

We trained our models on the Icelandic Frequency Dictionary (IFD) corpus (Pind et al., 1991), which contains about 590 thousand tokens, predominantly from literary texts. All previous taggers developed for Icelandic were trained and tested on this corpus. As the developers of IceStagger did, we used the so-called *corrected version* of the corpus, with a tagset of 565 morphosyntactic tags, and ten-fold split from Loftsson et al. (2009). The morphosyntactic tags in the tagset are mnemonic encodings, i.e. character strings where each character has a particular function. The first character denotes the *lexical category*. For each category there is a predefined number of additional characters (at most six), which describe morphological features, like *gender*, *number* and *case* for nouns, etc. To illustrate, consider the word form *maður* “man”. The corresponding tag is *nken*, denoting noun (*n*), masculine (*k*), singular (*e*), and nominative (*n*) case.

We also experiment with a more recent corpus, MIM-GOLD (Loftsson et al., 2010). It uses the same tagset as the IFD but contains a greater diversity of texts. In addition to texts from published books, it contains texts from news media, blogs, parliamentary speeches and more. Furthermore, MIM-GOLD is about twice as large as the IFD, i.e. containing approximately 1 million running words.

Baseline Model

In our baseline model, which is similar to Plank et al. (2016), both word embeddings and

[1] ABLTagger was written in collaboration with Örvar Kárason, an MSc student at Reykjavik University. Örvar and I both worked on all aspects of programming and testing of the program, with equal contribution from both. It is available at `https://github.com/steinst/ABLTagger`.

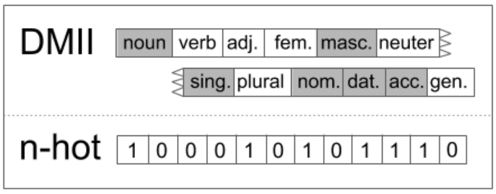


Figure 3.1: A partial n-hot vector and the corresponding features from the DMII. The example shows 12 features, including the active features for the word form *strætó* "bus". All possible features for the word form are activated. In this case, the word, a noun, has the same form for nominative, dative and accusative and therefore all corresponding labels are activated. An actual vector in our model has 61 labels, which are either active, `1`, or inactive, `0`.

recurrent character embeddings are used as input. The character embeddings for a given word are input into a BiLSTM. The output from the BiLSTM is concatenated to the word embedding and the combined vector input into another BiLSTM, whose output is input into a hidden layer. The hidden layer feeds the output layer, which selects a PoS tag.

Adding an External Morphological Lexicon

Horsmann and Zesch (2017) replicated the work of Plank et al. (2016) using a collection of corpora annotated with fine-grained tagsets of varying sizes, in contrast to the coarse-grained Universal Dependencies (UD) tagset in the previous study (17 tags). The replication confirmed the superior performance of the BiLSTM tagger, also on fine-grained tagsets. Furthermore, they found that the advantages of the BiLSTM tagger over other taggers grow proportionally with the tagset size of the corpus. However, they also claimed that for large tagsets of morphologically rich languages, hand-crafted morphological lexicons are still necessary to reach state-of-the-art performance. Using a morphological lexicon has become common practice for enriching training data for PoS taggers. Hajič (2000) marked the importance of this for morphologically rich languages and it was first done for Icelandic in Loftsson et al. (2011).

Sagot and Martínez Alonso (2017) first used morphological lexicons as supplemental input for PoS tagging with BiLSTM taggers and showed that it yields consistent improvement. Following their work, we extended the baseline model by adding an input layer that contains token-wise features obtained from the DMII lexicon, which contains over 300 thousand paradigms and six million inflectional forms. The input vector for a given word is an n-hot vector where each active value corresponds to one of 61 possible labels in the lexicon. This vector is concatenated to the two vectors described in the previous section, i.e. the word embedding and the character embedding, and the result is then fed into the BiLSTM layer. An example of an n-hot vector is given in Figure 3.1.

Previous taggers using DMII have had to map the information to the IFD tagset. As the tagsets of IFD and DMII are not completely compatible, some information has been lost in the mapping process. Our method on the other hand allows the model to use and learn from all the information encoded in the morphological lexicon, even though it uses a tagset slightly different from the training data.

Lexical Category Identification Step

When employing a fine-grained tagset with mnemonic encoding, the model does not place different significance on two tags when they differ in lexical category, on one hand, or share a lexical category but differ in morphological features, on the other. A human, however, would consider the former a more significant error than the latter. A PoS tagger is especially

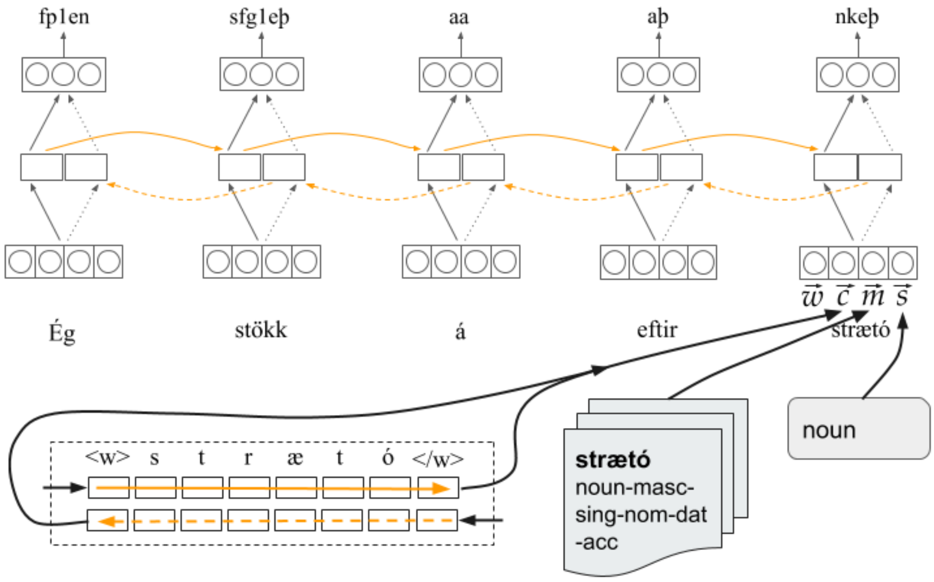


Figure 3.2: Our full tagging model, employing word embeddings, character embeddings, a morphological lexicon, and the output of the lexical category identification step. The hidden layer is omitted for simplicity. Figure adapted from Plank et al. (2016).

prone to such errors when the tagset is large and the amount of training data is insufficient to detect all the subtle differences between labels.

To place a higher emphasis on assigning the correct lexical category, we devised a two-step process. First, we simplified the tagset from 565 to 10 tags by using only the first letter of the fine-grained tag mnemonic, i.e. the letter denoting the lexical category. We then trained our third and final model on this new coarse-grained tagset, using word and character embeddings as well as the morphological lexicon. This resulted in a lexical category tagger with very high accuracy, 98.97% in our case. In the second step, the output of that tagger is embedded as a one-hot vector and concatenated to the vectors input into the BiLSTM layer of the main model. This guides the tagger to the correct lexical category and eliminates some of the errors caused by insufficient training data. This final model is shown in Figure 3.2.

3.1.2 Part-of-Speech Tagging Results

The test results for all three models are shown in Table 3.1. The substantial gain achieved by using DMII confirms the advantages of using an external morphological lexicon. Employing the stepwise model further increases accuracy by helping in assigning rare or ambiguous

	Accuracy	Known	Unknown
Baseline	93.25%	95.19%	**66.84%**
+ DMII	94.84%	95.17%	54.61%
+ LC	**95.15%**	**95.48%**	54.06%

Table 3.1: Accuracy of the three models trained and tested on IFD. The results show how the accuracy improves when the model is augmented with the Database of Modern Icelandic Inflection (DFII), and again when the lexical category (LC) identification step is added.

	Accuracy	Known	Unknown
MIM-GOLD	94.04%	95.13%	68.34%
+ IFD	94.17%	95.62%	68.18%

Table 3.2: Accuracy when training and testing on MIM-GOLD.

tags in the fine-grained tagset guided by the highly accurate lexical category. Note that the baseline model achieves the highest accuracy for unknown words. This is because on average, the IFD ten-fold splits contain 58,977 words and by incorporating DMII, the average unknown word rate in testing falls from 4,036 to 476, a drop from 6.8% to 0.8%. The words that are still unknown are mostly foreign words and rare or unorthographical word forms, which are the word classes the taggers struggle most with.

While our tagger achieved a significant gain in accuracy over previous taggers, using the same training and testing data they used, these datasets, containing mainly literary texts, are not necessarily characteristic of texts that need to be tagged for language technology or research purposes. We thus also evaluated our system using a more recent gold standard, MIM-GOLD, which contains more diverse texts.

Table 3.2 shows that there is a substantial drop in accuracy compared to training and testing on the IFD (see Table 3.1). The lower accuracy may, at least partly, be due to a greater variety in texts than before and a larger proportion of unknown words and word forms in the MIM-GOLD test set compared to IFD (Steingrímsson et al., 2015). This also shows the importance of the choice of training and testing data. In practice, PoS tagging is not carried out primarily on literary fiction and the training data should reflect that. For further discussion, comparison to previous taggers for Icelandic and error analysis, refer to Steingrímsson et al. (2019).

Since the first version of ABLTagger was published in 2019 it has been developed further by others to reach even more accuracy. The latest version uses a BERT-like model and an updated version of MIM-GOLD with a somewhat revised tagset (Jónsson and Loftsson, 2022). While the results are not entirely comparable due to differences in the tagset, the developers report the latest version to reach 97.8% accuracy.[2]

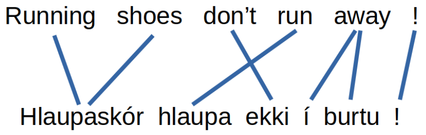

Running shoes don't run away !
Hlaupaskór hlaupa ekki í burtu !

Figure 3.3: A simple example of English–Icelandic word alignments. Corresponding words are connected by edges.

3.2 CombAlign

Word alignment, the task of finding corresponding words in a bilingual sentence pair (see Figure 3.3) was a key component of statistical machine translation (SMT) systems. While word alignments are not necessary for neural machine translation (NMT), various methods incorporating word alignment have been found to achieve significant improvements in performance. Alkhouli et al. (2018) and Liu et al. (2016) use alignments as a prior; Arthur et al.

[2] Latest model and reported accuracy released on `https://github.com/cadia-lvl/POS`

Language Pair	Gold Standard	Sentence Pairs	Edges
en-cs	Mareček (2008)	2,501	67,424
en-de	Europarl[6]	508	10,534
en-fr	Och and Ney (2000)	447	17,438
en-is	*new*	384	5,517

Table 3.3: Gold standard alignments used for evaluation. The en-is gold standard contains further 220 sentence pairs that were used as a development set for grid search.

(2016) augment NMT systems with discrete translation lexicons that encode low-frequency words; Press and Smith (2018) infer a correspondence between words in sentence pairs before encoding/decoding and, as demonstrated by Poncelas et al. (2019), back-translated data created using SMT systems, requiring word alignments, can be valuable to augment NMT systems. Word alignments have also been utilized to improve automatic post-editing (Pal et al., 2017) as well as to preserve markup in machine-translated texts (Müller, 2017). Shi et al. (2021) show that by simply pipelining word alignment with unsupervised bitext mining, BLI efficiency can be improved significantly. For BLI, Artetxe et al. (2019) use an unsupervised MT pipeline, also employing word alignments. Kurfalı and Östling (2019) use word alignments to filter noisy parallel corpora, and Paetzold et al. (2017) include word alignment as a part of their pipeline to align monolingual comparable documents.

We use word alignments as a part of multiple pipelines. We use it to help create a bilingual lexicon (Section 3.6), for scoring sentence pairs when filtering parallel corpora (Chapter 4) and as a part of a pipeline to extract mutual translations from comparable corpora or data that can be regarded as comparable corpora (Chapter 6).

A variety of different available word alignment tools, based on different approaches, are able to attain a fairly high F_1-score on a variety of evaluation sets, as shown in Figure 3.4. It is reasonable to expect that combining their results in a sensible way could give better results than using any one of the individual systems. We thus create an experiment where we compare the results from common aligners and then compare these to ensemble alignments created by *CombAlign* (Steingrímsson et al., 2021a), a tool we developed to combine the output of multiple word aligners in order to try to maximize precision or recall, depending on the use case.[3] The tool runs the five aligners and returns all alignments that the majority of the aligners agree upon. In order to raise recall we can relax the demands for agreement, accepting alignments that two or more of the five aligners agree upon.

We evaluate on four language pairs, using known test sets that have been used in multiple previous work for three of the language pairs. For English–Czech we used the evaluation set provided by Mareček (2008), for English–German we used the Europarl evaluation set,[4] and for English–French we used the evaluation set provided by Och and Ney (2000). Additionally, we compiled a new gold alignment test set for English–Icelandic.[5]

[3] Available at: `https://github.com/steinst/CombAlign`.

[4] Available at: `https://www-i6.informatik.rwth-aachen.de/goldAlignment/`

[5] Available at: `https://repository.clarin.is/repository/xmlui/handle/20.500.12537/103`. The test set was compiled by me with Hjalti Daníelsson assisting on manually annotating the alignments. A detailed description of the dataset and how it was created is provided in Steingrímsson et al. (2021a).

3.2.1 Experimental Setup

There is a variety of word aligners available. For our experiments we use four different systems. Giza++ (Och and Ney, 2003) and fast_align (Dyer et al., 2013) are easy to use implementations of the IBM models (Brown et al., 1993). While fast_align builds on IBM model 2, Giza++ iterates on a number of the models in sequence, as well as using an HMM model. eflomal (Östling and Tiedemann, 2016), using a Bayesian model with Markov Chain Monte Carlo inference on the IBM models, is fast and gives competitive results. SimAlign (Masoud et al., 2020) takes advantage of the rising availability of contextualized embeddings and leverages them by extracting alignments from similarity matrices.

Giza++, fast_align and eflomal are trained on parallel data. For all language pairs except English–Icelandic, we use a a subset of $512,000$ sentences from Europarl (Koehn, 2005) to train the models. For English–Icelandic we use sentence pairs from the first version of the ParIce corpus (see Section 3.5.1).

In order to find the best settings for Giza++, fast_align and eflomal, we run the systems using different heuristics and compare the results. With SimAlign, we induce alignments from two different contextualized embedding models, multilingual BERT (mBERT) (Devlin et al., 2019), and XLM-R (Conneau et al., 2020). We are thus working with five different aligners/alignment models. For SimAlign, we carry out a grid search to find the best set of hyperparameters for each model and run the experiments both for whole words and BPE. Detailed results of the grid search is provided in Steingrímsson et al. (2021a). The paper also describes how different hyperparameters and different levels of agreement should be chosen, depending on whether the objective is to reach high recall, high precision or a high F_1-score. For the comparison in the next section, we run CombAlign using the setting that we expect to obtain the highest F_1-score.

3.2.2 Results

As shown in Table 3.4, the results vary considerably between language pairs. This can, at least in part, be explained by the fact that the gold alignment data the aligners are evaluated on are all created by different researchers, at different times using different source material. The criteria for what constitutes an alignment may thus have varied to some extent when the gold alignments were compiled. The sets also vary in size (see Table 3.3). For all language pairs, the CombAlign approach based on an ensemble of five alignment models gives the highest F_1-scores. The downside to this approach is that it requires all aligners to be run, being by far the most time and computing power intensive. When comparing individual aligners, SimAlign employing BERT gives the highest score for three out of four language

Method	**en-cs**	**en-fr**	**en-de**	**en-is**
eflomal	.86	.91	.73	*.91*
fast_align	.78	.86	.70	.89
Giza++	.81	.89	.73	.88
SimAlign: XLM-R	*.87*	.93	.78	.90
SimAlign: BERT	*.87*	*.94*	*.81*	.86
CombAlign	**.91**	**.95**	**.83**	**.95**

Table 3.4: Word alignment F_1-scores for the four language pairs. The highest scoring alignments are in bold, all ensemble alignments created by CombAlign. The highest scoring individual aligners for each language pair is in italics.

pairs. For one pair, English–Icelandic, eflomal gave a slightly higher F_1 score than SimAlign. A possible reason for this is that the contextualized models SimAlign uses were trained on less Icelandic data and so have more 'knowledge' of the other languages than of Icelandic.

Based on these results, we proceed to use CombAlign and an ensemble of available word aligners when taking advantage of word alignment information in our further work.

3.3 PivotAlign

When filtering parallel data (Chapter 4) , aligning parallel corpora (Chapter 5), and extracting parallel sentence pair candidates from comparable corpora and data discarded when aligning and filtering (Chapter 6), we use lexicon-based methods as well as other approaches. In order to induce an English–Icelandic bilingual lexicon (Section 3.6), we apply various approaches, amongst them pivoting through intermediary dictionaries and extracting translation candidates from a parallel corpus using word alignments. We developed a tool, PivotAlign (Steingrímsson et al., 2021c), to experiment with combining these two approaches. We tested PivotAlign and compared against other approaches by participating in the Translation Inference Across Dictionaries (TIAD) 2021 shared task (Gracia et al., 2021), obtaining a very competitive result.[7]

Word alignments have previously been used for automatically inducing bilingual dictionaries, see e.g. (Melamed, 2000; Caseli et al., 2006; Shi et al., 2021). It is simple to regard the outputs of word alignment models as hypotheses for translation equivalence. However, the problem with word alignments has been that these hypotheses are not necessarily very accurate, both due to the limitations of the aligners themselves and to the limitations of the data being aligned. We try to circumvent these limitations by using CombAlign (Section 3.2) to obtain sets of alignment suiting our purposes, whether our goal is high precision, high recall or high F_1-score.

3.3.1 Experimental Setup

Our approach is based on two methods applied in conjunction to induce a bilingual dictionary: word alignments and pivoting through intermediary dictionaries. We created three versions of our system for three different goals: high precision, high recall and a high F_1-score. In the experiment, we worked with all translation directions between three languages, English, French and Portuguese, resulting in six induced dictionaries: en→pt, pt→en, pt→fr, fr→pt, en→fr and fr→en.

We started by collecting as many lexical translations as possible, using a subset of Apertium RDF v2 (Gracia et al., 2020) (see Figure 3.4). Our main approach is pivoting through either one or two intermediary languages for each language pair. In order to score the candidate lexical translations, we extract sentence pairs from a parallel corpus, align them at the word level and calculate a word alignment score for each aligned pair of words, in effect using the extracted word alignment pairs as a filter for the candidates obtained by pivoting through Apertium.

It has been demonstrated that by using a method called One Time Inverse Consultation (OTIC) it is possible to get a translation candidate or a list of translation candidates in the target language with a good likelihood of the candidates being relevant (Tanaka and Umemura, 1994). OTIC induces a candidate list through a pivot language, but sets restrictions that result

[7] Available at: https://github.com/steinst/pivotalign.

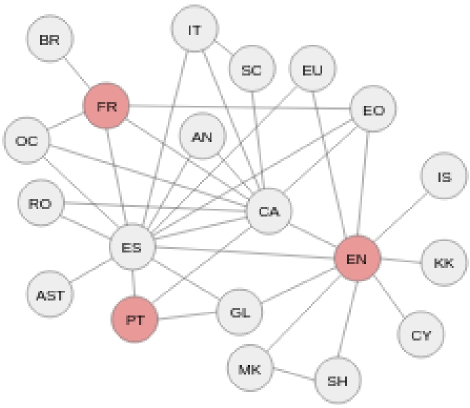


Figure 3.4: The subset of the Apertium dictionaries used in our experiment. Each bilingual dictionary is represented by an edge between vertices in the graph. In the experiment we aim to infer translations between the languages in the red circles.

in pruning of unlikely candidates based on information in the available dictionaries. OTIC was used in one of the baseline systems for the TIAD 2021 shared task.

As our method relies on a scoring mechanism external to the Apertium dictionaries, we want to extract as many potential candidates as possible. We thus opt for taking a more naive approach to collecting translation candidates and simply accept all words inferred through the intermediary dictionaries. This gives us lists of translation candidates for each of the language pairs we are working with, which are larger than the pruned lists OTIC generates. Before proceeding to filter the data, we iterate using our new induced and unfiltered dictionaries. By connecting the three languages, English, French and Portuguese, in all possible combinations, with one as a source language, another as a target language and the third as an intermediary, we obtain even more translation candidates.

While our system investigates all possible paths through one or two intermediary dictionaries between our source and target languages, we compile another list of word pairs by running word alignments on parallel corpora, with 1 million sentence pairs for each language pair. The corpora were obtained from OPUS[8] (Tiedemann, 2012), with the sentence pairs selected from larger corpora only if they contained a word from the source or target language intermediary dictionaries. In order to get a decent coverage, after a word was found 10 times it was removed from the list. We lemmatized the sentences using spaCy[9] before doing the word alignments. After aligning the parallel corpora using six word alignment systems and models, the five described in Section 3.2 as well as AWESoME (Dou and Neubig, 2021), which exploits multilingual BERT to extract the word alignments, different settings of CombAlign (Section 3.2) were used to select the final word alignments depending on whether the aim was high precision, high recall or high F_1-score. Different goals can suit different needs. in our work on building a bilingual lexicon, described in Section 3.6, automatically generated candidates for lexicon entries are manually checked and to obtain a high coverage we aim for high recall. In Chapter 4 and Chapter 6, where we experiment with using WAScore (see Section 3.4), a word-alignment-based approach to assist with estimating the likelihood of a sentence pair containing mutual translations, we aim for high precision word alignments.

[8] https://opus.nlpl.eu/

[9] https://spacy.io

After obtaining the alignments from CombAlign, we use the word alignment frequency combined with a count of word co-occurrences in the sentence pairs to score the candidates. The score is calculated for each word pair $\langle s, t\rangle$ using Equation 3.1:

$$\rho\left(s, t\right) = \frac{\text{mat}\left(s, t\right)}{\text{coc}\left(s, t\right) + \lambda} \tag{3.1}$$

where $\text{mat}\left(s, t\right)$ is the one-to-one matching count, i.e. how often the words are aligned in the corpus, and $\text{coc}\left(s, t\right)$ is the number of one-to-one co-occurrences, i.e. count of $\langle s, t\rangle$ appearing in a sentence pair in the corpus. λ is a non-negative smoothing term. The equation was proposed by Shi et al. (2021) but we use it with a slight variation. While Shi et al. (2021) set the smoothing variable λ to 20, we set it to $\log_2 n$ where n is the number of sentence pairs in the corpus under consideration. This makes the score more comparable between corpora of different sizes. The scores are in the range $[0, 1]$.

3.3.2 Results

We submitted three variants of PivotAlign to the TIAD 2021 shared task, one aiming for high precision, another aiming for high recall and the third aiming for a high F_1-score. The detailed settings are described in (Steingrímsson et al., 2021c). As shown in Table 3.5, two of our submitted system variants outperformed all other participating systems with respect to F_1 score, showing the usefulness of a scoring mechanism based on accurate word alignments extracted from parallel corpora. Our third variant, PivotAlign-P, achieved a precision of 0.85, but it had much lower recall, 0.24, and came in seventh in terms of F_1 score.

The candidate translations accepted by PivotAlign-R, aiming for high recall, had the minimum threshold for $\rho\left(s, t\right)$ set to 0.15, thus including candidate translations with $\rho\left(s, t\right)$ from 0.15 and up to 1.00. These scores, measured against the evaluation sets, show how the precision score rises almost linearly as the threshold rises, while recall goes down, see Figure 3.5. This indicates a good correlation between our alignment score and translation inference, showing that a scoring mechanism based on accurate alignments from an ensemble of word alignment tools can be highly valuable for tasks such as this one.

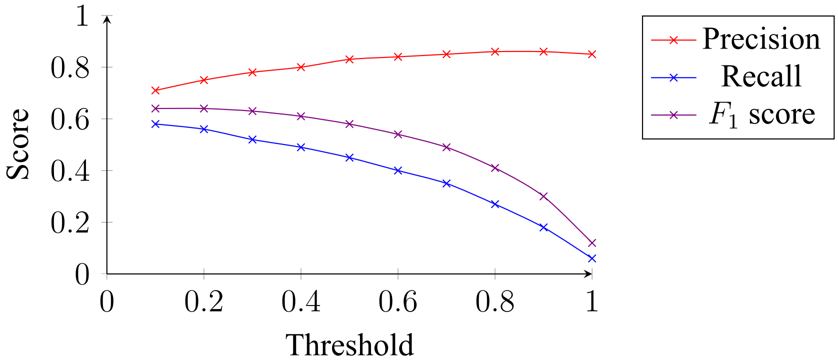


Figure 3.5: Precision, recall and F_1 score charted against various threshold settings for PivotAlign-R. The threshold is the minimum value of the confidence score, $\rho\left(s, t\right)$. It is highly correlated with all three scores.

Top 5 Systems				
System	Precision	Recall	F_1-score	Coverage
PivotAlign-R	0.71	**0.58**	**0.64**	0.77
PivotAlign-F	**0.81**	0.51	0.62	0.68
ACDcat	0.75	0.53	0.61	0.75
TUANWEsg	**0.81**	0.47	0.59	0.76
TUANWEcb	**0.81**	0.47	0.59	0.76

Table 3.5: The five highest ranking systems, in terms of F_1, submitted to the TIAD 2021 shared task. Our systems are the PivotAlign systems.

3.4 WAScore

In order to be able to take advantage of word alignments as a scoring mechanism, when filtering parallel corpora or when mining parallel sentence pairs from comparable corpora, we need high-precision alignments and a scoring formula.

Word alignments have previously been used for parallel sentence extraction, generally under the assumption that if a pair of sentences is equivalent in two languages, there should be a number of word alignments between the sentences, and, in contrast, non-parallel sentences should have few or no word alignments. Stymne et al. (2013) use alignment-based heuristics to filter out sentence pairs. They hypothesize that sentence pairs with very few word alignments common to both directions would likely not be corresponding sentences. They use GIZA++ to find the alignments and set three thresholds for accepting valid sentence pairs: 1) Ratio between number of alignment points and maximum sentence length; 2) Absolute number of alignment points in a sentence pair; 3) Length ratio of the sentences. Lu et al. (2020) use a word alignment based translation score as a part of their scoring ensemble for filtering a noisy parallel corpus. Their translation score is a simplified version of the translation score introduced by Khadivi and Ney (2005) and based on fast_align probability scores. Zariņa et al. (2015) identify parallel sentences using word alignments, experimenting with five different alignment based scores, four of which are based on fast_align probability scores. They find that the best out of the five approaches is a geometric mean of the alignment probability scores for each token.

Our approach to creating alignment-based scores is to collect high precision word alignments using CombAlign (Section 3.2). As we are combining outputs from multiple aligners to create a final alignment using CombAlign, we do not have probability scores for each alignment. In light of the results of our word alignment experiment (see Table 3.4), we work under the assumption that our alignment approach eliminates most extraneous alignments without missing too many of the correct ones. In Steingrímsson et al. (2021b) we calculate a word alignment score by multiplying the ratio of aligned tokens in the source sentence to the aligned tokens in the target sentence and call it *WAScore*:

$$\frac{s_a}{s} \times \frac{t_a}{t} \tag{3.2}$$

In Equation (3.2), s is the number of words in the source sentence and s_a is the number of source words that are aligned to some word in the target sentence, t is the number of words in the target sentence, and t_a is the number of target words that are aligned to some word in the source sentence.

With a set of highly likely alignments for each sentence pair, the WAScore tends to favour sentences of similar length as a much longer sentence on one side usually has proportionately few alignment edges on that side which lowers the score substantially. In contrast, if a shorter sentence on one side has all tokens aligned to a longer sentence on the other side, it can result in a reasonable score. Such pairs are often partially parallel, meaning that a part of either sentence can align perfectly with either the whole or a part of the other sentence. In Steingrímsson et al. (2021b), we show that this approach is suitable for extracting partially parallel sentence pairs as well as truly parallel ones. In order to differentiate between these two, when filtering or extracting sentence pairs from comparable corpora, we use additional scoring mechanisms to raise the accuracy further.

3.5 ParIce

The ParIce English–Icelandic parallel corpus was compiled in order to facilitate work on MT for that language pair. Before it was published, MT training data was fragmented and compiled using generic approaches not necessarily focusing on making the sentence pairs for this particular language pair as accurate as possible. The available data was part of multilingual datasets, shown in Table 3.6. The largest of these datasets was the OpenSubtitles corpus, (Tiedemann, 2016) available from the OPUS project. The second largest was data obtained from the European Medicines Agency (EMA), as part of the Tilde MODEL (Multilingual Open Data for EU Languages) corpus (Rozis and Skadiņš, 2017).

The aim of the ParIce corpus project was to collect available corpora in one place and refilter them, as well as compiling additional data. Two versions of the corpus have been published. They are described in the following sections, as well as an automatic evaluation of training NMT systems using the published versions.[10]

3.5.1 The First Version

The first version of the corpus was published in 2018 (Barkarson and Steingrímsson, 2019). While the work was carried out in part by me, it was not part of the research carried out for this thesis. Nevertheless, as the work done on the second version of the corpus, ParIce 21.10, was part of the research for this thesis, I will give an overview of how the first version was compiled.

Previously available data was collected. An examination of random samples of sentence pairs indicated that for some of the subcorpora, in particular KDE4, OpenSubtitles and Statistics Iceland, a rather large portion of the sentence pairs were faulty. For KDE4 almost half the pairs were deemed faulty, while approximately 8% were deemed faulty for the other two subcorpora. This is most commonly due to misalignment, semantic mismatch or inadequate optical character recognition (OCR) of the texts being aligned. Between 200 and 800 sentence pairs were manually assessed, depending on size of the subcorpus. The results are shown in Table 3.6.

[10] The work on the corpus was joint work between me and Starkaður Barkarson. For the first version, published in 2018, I collected the data and Starkaður and I worked together on the alignments and evaluation. For the second version of the corpus, published in 2021, Starkaður collected more data and I aligned it, scored the sentence pairs and filtered the corpus.

[11] `http://christos-c.com/bible/`

[12] `https://tilde-model.s3-eu-west-1.amazonaws.com/Tilde_MODEL_Corpus.html`

[13] `http://opus.nlpl.eu`

[14] Available at the ELRC-SHARE repository

Corpus	Sentence Pairs	Faulty Pairs (%)
The Bible[11]	31,085	0.5
EMA[12]	420,297	3.3
Gnome[13]	5,431	n/a
KDE4[13]	87,575	45.0
OpenSubtitles[13]	1,368,170	8.3
Statistics Iceland[14]	2,360	8.0
Tatoeba[13]	8,139	0.0
Ubuntu[13]	2,127	2.5
TOTAL	**1,923,060**	

Table 3.6: Pair count and ratio of bad alignments in the available parallel corpora collected for ParIce. A large part of the Icelandic translation of the KDE4 dataset was untranslated or containing placeholders, explaining the high ratio of faulty pairs. An inspection of the Gnome dataset revealed similar issues, resulting in a decision not to work further with that data.

The corpus was realigned when possible and refiltered in order to increase the quality of the corpus. Furthermore, new data was added. For the first version, regulations and directives in relation to the EEA agreement were collected in English and Icelandic and aligned on sentence level using LFAligner,[15] which is based on Hunalign (Varga et al., 2005). Furthermore, some out-of-copyright books were collected from Project Gutenberg and a similar website specializing in Icelandic books,[16] as well as news from the European Southern Observatory (ESO),[17] and aligned in the same way.

In order to filter out the misalignments, a scoring and filtering mechanism was devised. It was based on an NMT system and a makeshift dictionary containing all possible Icelandic word forms for any given English word form. The dictionary is based on the Apertium dictionary (Brandt et al., 2011), Icelandic Wiktionary,[18] a bilingual dictionary built from the parallel data using bitextor-builddics (Esplà-Gomis, 2009), and a set of wordforms extracted from DMII. OpenNMT (Klein et al., 2017) was used to train the NMT system on data from a one-million-segment translation memory acquired from the Translation Centre of the Ministry for Foreign Affairs, in addition to the parallel data obtained from OPUS.

All sentences in the corpus were translated using the NMT system and scored by counting how many words in the source sentence were represented in the target sentence and vice versa using this equation:

$$score = \frac{(s_r/s) + (t_r/t)}{2} \tag{3.3}$$

where s_r are the number of words in the source sentence present in the translation of the target sentence, s the total number of words in the source sentence, t_r the number of words in the target sentence present in the translation of the source sentence and t the total number of words in the target sentence. Consider the sentence pair in (1), where (a) is the source sentence in Icelandic and (b) is the target sentence in English: As each word in the Icelandic sentence is found in the translation of the target sentence and translations of only three of

[15] http://sourceforge.net/projects/aligner
[16] https://rafbokavefur.is/
[17] https://www.eso.org/
[18] https://www.wiktionary.org/

Subcorpus	**Before Filtering**	**Accepted Pairs**	**Accepted Pairs (%)**	**Faulty Accepted Pairs (%)**	**Faulty Deleted Pairs (%)**
The Bible	32,964	32,964	100.0	0.0	n/a
Books	16,976	12,416	73.1	3.5	38.0
EEA	2,093,803	1,701,172	81.3	5.0	63.5
EMA	420,297	404,333	96.2	1.3	45.0
ESO	12,900	12,633	97.9	0.5	46.0
KDE4	137,724	49,912	36.2	9.0	n/a
OpenSubtitles	1,620,037	1,305,827	80.6	1.4	37.0
Sagas	43,113	17,597	40.8	11.0	55.5
Statistics Iceland	2,481	2,288	92.2	5.0	56.0
Tatoeba	8,263	8,263	100.0	0.0	n/a
Ubuntu	11,025	10,572	95.9	2.0	n/a
TOTAL	**4,399,582**	**3,557,977**	**80.9**		

Table 3.7: Pair count before and after filtering as well as ratio of accepted pairs and deleted pairs that were deemed faulty during the assessment of the first version of ParIce.

eight English words are represented in the corresponding Icelandic sentence, the score would be $(1 + 0.38)/2 = 0.69$.

(1) a. Hann gekk inn. *(e. He walked in)*

b. As he walked in he sang a song.

Furthermore, the average of all sentence pair scores in each document is calculated, and if the document score is below a threshold it is discarded. For other documents, the sentence pairs are deleted if multiple pairs in a row have a score under a given threshold. This is to allow for limitations in the scoring mechanism, which sometimes gives low scores for good sentence pairs, and thus the low scoring sentence pairs are accepted unless they appear in clusters.

Before filtering the texts, the corpus contained 4,399,582 sentence pairs in total. After filtering it had 3,557,977 pairs, including repeated sentence pairs. When duplicates had been removed, 2,774,942 were left for MT training. A manual evaluation was carried out and sentence pairs that were accepted and discarded were evaluated for each data source. If the sentence pairs were deemed to be a mutual translation by the annotator they were accepted, otherwise they were rejected. Table 3.7 gives the number of sentence pairs for each data source in the corpus and shows that while the filtering process usually did not accept many bad pairs, for many of the text sources over half of what was discarded were good sentence pairs. Barkarson and Steingrímsson (2019) describe the compilation of the first version of ParIce in more detail.

In our experiments with different MT models, we filtered the first version of ParIce further (Jónsson et al., 2020).[19] The additional filtering was primarily based on rules, such as sentence length ratio, minimum and maximum length, digit mismatch, a character whitelist filter, removing sentence pairs with corrupt symbols, e.g. '?' inside words, removing identical and close to identical source and target sequences and other shallow filters. After applying these filters 1,642,927 sentence pairs were left for training.

[19] I was one of five authors of this paper. My main contribution was overseeing the human evaluation of the models.

Corpus	**Sentence pairs**
The Bible	32,423
EMA	405,088
EEA	3,418,465
ESO	10,079
KDE4	17,654
Norden	11,441
OpenSubtitles	1,412,940
Statistics Iceland	2,288
Tatoeba	9,440
Ted	2,392
Ubuntu	7,431
TOTAL	**5,329,641**

Table 3.8: Number of sentence pairs from each source of ParIce 21.10. These numbers are before deduplication and filtering.

3.5.2 The Second Version

A second version of the ParIce corpus, ParIce 21.10, was published in 2021.[20] It contained updated data from the same data sources as before with some additions: News texts published by the Nordic Council of Ministers[21] were acquired and subtitles of TED talks available at OPUS were added. All texts were checked for OCR errors and whether the texts had an expected ratio of the Icelandic letters 'þ' and 'ð', an indicator of whether the texts have been processed and saved correctly. Documents with such errors were discarded and all other texts aligned using Vecalign (Thompson and Koehn, 2019). Table 3.8 shows the number of sentences from the different sources.

We score all sentence pairs using three different scoring mechanisms: Language-Agnostic SEntence Representations (LASER) (Schwenk and Douze, 2017) and LaBSE (Feng et al., 2022), described in Section 4.2.2, and WAScore (Steingrímsson et al., 2021b) described in Section 3.2.

Three filters were created for the aligned data: 1) A character filter that filters out sentence pairs where more than 40% of tokens in either language contain characters that are not punctuation marks or alphabetic letters in the English or Icelandic alphabets; 2) A length filter that filters out all sentence pairs where either of the sentences is shorter than 4 tokens or longer than 200 tokens; 3) A score filter using a logistic regression classifier trained on the three scoring mechanisms. The classifier and training set used for training it are described in more detail in Section 4.2.2.

The dataset is published with all the sentence pairs, accompanied with flags representing the outcome of the three filters. Table 3.9 shows the total number of sentences when evaluation sets (Section 3.5.3) and duplicates have been removed and different filters have been applied.

[20] The work was carried out in cooperation between me and Starkaður Barkarson. Starkaður collected additional data and I aligned, scored and filtered the data.

[21] https://norden.org

Filter	No. of pairs
Unfiltered	3,464,789
Character filter (CF)	2,735,354
Length filter (LF)	2,896,886
Score filter (SF)	2,470,838
Shallow filters (CF + LF)	2,453,135
All filters (CF + LF + SF)	1,864,679

Table 3.9: Number of distinct sentence pairs in ParIce 21.10 training data after different filters have been applied.

Data Source	Valid pairs	Test	Dev	No. of files
Open Subtitles	7,388	3,694	3,694	N/A
EMA	6,279	3,139	3,140	N/A
EEA	5,812	2,189	2,190	25
Norden	2,317	974	974	116
ESO	2,561	1,213	1,213	106
Total	24,357	11,209	11,211	247

Table 3.10: Number of distinct sentence pairs in ParIce 21.10 development and evaluation datasets after different filters have been applied.

3.5.3 Evaluation Sets

We created development/test sets out of subsets from five different sources included in the ParIce 21.10 corpus.[22] These sets can be used to run automatic evaluations on different MT models for these different domains. From the EEA, Norden and ESO datasets, we randomly selected a set of files for these datasets and removed these files from the training data. For training sets from Open Subtitles and EMA, we randomly selected sentence pairs to be evaluated for inclusion in the evaluation sets. The data was divided between four annotators who evaluated whether sentence pairs were valid or invalid and selecting only valid sentences for the evaluation sets.[23] Table 3.10 shows the number of sentence pairs accepted as valid for each source.

3.5.4 Evaluation

To establish a baseline for comparing our experiments, we evaluate the different versions of ParIce by training MT models using identical hyperparameters. We use fairseq (Ott et al., 2019) to train Transformer$_{\text{BASE}}$ models, as described in Vaswani et al. (2017), except that we set dropout to 0.2, in line with Sennrich and Zhang (2019). Their results indicate that a more aggressive dropout than applied in the original transformer paper leads to higher BLEU scores in low and medium resource settings, and we use byte pair encoding with a shared vocabulary size of $32,000$. We train each model on a single A100 GPU and use early stopping with the patience set to 10 epochs.

We evaluated six versions of ParIce training data against the WMT 2021 news translation evaluation sets (Akhbardeh et al., 2021). For the first version of ParIce, we train two models. One using all the corpus and the other one using a filtered version of that corpus,

[22] Available at `http://hdl.handle.net/20.500.12537/146`

[23] I organized the work and annotated the sentence pairs with the help of Finnur Ágúst Ingimundarson, Árni Davíð Magnússon and Hildur Hafsteinsdóttir. Starkaður Barkarson packaged the data for publication.

Training Set	No. Pairs	en-is BLEU	is-en BLEU
ParIce 1	2,774,942	13.5	*25.1*
As filtered by Jónsson et al. (2000)	1,642,927	18.0	24.7
PI 21.10 raw	3,464,789	17.5	*25.4*
PI 21.10 Character filter	2,735,354	18.5	24.5
PI 21.10 Shallow filters	2,453,135	*19.0*	24.7
PI 21.10 All filters	1,864,679	**19.2**	**25.7**

Table 3.11: BLEU scores for Transformer$_{\text{BASE}}$ NMT models trained on different versions of ParIce. The systems are evaluated on WMT 2021 news translation test data. Scores in bold are highest. Scores in italics are lower but not significantly lower than the highest scores ($p > 0.05$).

which was published by Jónsson et al. (2020). We trained four models on Parice 21.10, one on the unfiltered corpus, the second one using only the character filter, the third one using the character filter and the length filter, and the fourth one on both the shallow filtering approaches and the classifier based on the three scoring mechanisms described in 3.5.2. All models are evaluated using SacreBLEU (Post, 2018).[24] ParIce 21.10 generally performed better than ParIce 1. This is more prominent when translating from English into Icelandic, although there are some improvements in the other translation direction also. The filters applied to ParIce 21.10 are also beneficial for English→Icelandic, but the shallow filters seem to harm the translation quality in the other direction, at least as measured by BLEU. Table 3.11 shows BLEU scores, statistical significance calculated using the pairwise bootstrap test (Koehn, 2004).

Furthermore, we evaluate the four models, trained on different subsets of ParIce 21.10, on the five ParIce evaluation sets described in section 3.5.3. Note that during training we have not used the development data for these datasets, only WMT 2021 development data. We do not evaluate models trained on data from the first version of ParIce because some sentence pairs in the evaluation sets may be present in these training data. Results of the evaluation on the ParIce 21.10 evaluation sets are shown in Table 3.12.

As evident from tables 3.11 and 3.12, different evaluation sets give very different BLEU scores. The WMT test sets are made of sentences extracted from news media and translated especially to be used for evaluating MT systems. The domain is quite open, the sentences are rarely short, on average over 20 words for both Icelandic and English, and can be complicated. The Norden data also has sentence pairs from news texts and thus has a resemblance to the WMT test sets. While OpenSubtitles also has texts from a rather open domain, the sentences extracted from movie or television subtitles are, in contrast, considerably shorter, less than nine words on average, and therefore often simpler. This may explain some of the difference in BLEU scores between these data sets. The EEA and EMA data on the other hand are from rather closed domains, but are well represented in the training data. The MT systems can therefore obtain high scores for these datasets. It is also evident from Table 3.12 that the is→en translation direction scores higher than en→is. BLEU matches words and n-grams between the MT output and a reference translation. For morphologically rich languages this may be too simplistic. Belinkov et al. (2017) suggest that while translating from morphologically rich languages is challenging, translating into such languages is even harder, with BLEU scores in their experiments consistently being lower when translating from English

[24] SacreBLEU Signature: BLEU+numrefs.1+case.mixed+tok.13a+smooth.exp +version.2.2.0

		en-is				**is-en**			
		Filters				**Filters**			
Evaluation Set	**No. Pairs**	**No**	**Ch**	**C+L**	**All**	**No**	**Ch**	**C+L**	**All**
EEA	2,189	39.1	39.2	39.8	**41.2**	48.0	49.3	49.7	**50.6**
EMA	3,139	46.0	46.2	47.0	**48.3**	53.9	55.3	55.5	**56.8**
ESO	1,213	23.9	24.5	*25.8*	**26.4**	30.0	30.9	**31.9**	**31.9**
Norden	974	19.3	20.0	20.6	**21.3**	26.9	*27.8*	*27.8*	**28.0**
OpenSubtitles	3,694	33.2	*33.4*	*33.4*	**34.0**	34.3	*35.8*	*35.8*	**36.0**

Table 3.12: Evaluation of Transformer$_{\text{BASE}}$ NMT models trained on ParIce 21.10 using different levels of filtering. BLEU scores are given for five different evaluation sets distributed with the corpus. The highest scores are in bold and scores that are lower but not significantly lower in italics ($p > 0.05$).

into morphologically rich languages than the other way around. This may partly be due to better source-side representations when translating from the morphologically rich language.

In order to make our results as comparable as possible to other work, we will use the WMT 2021 evaluation set for further experiments in this work, when the nature of the experiments do not necessitate the use of other evaluation sets.

3.6 Bilingual Lexicon

Bilingual lexicons are useful for an array of different tasks. In our work, we use a bilingual lexicon when filtering parallel sentences (Chapter 4), in sentence alignment (Chapter 5), when working with comparable corpora (Chapter 6), and when extracting sentence pairs from discarded data (Chapter 6). In order for the approaches taking advantage of a lexicon to be useful, we want a fairly large lexicon. Previously, only the Wiktionary and Apertium dictionaries were publicly available for this language pair, containing approximately 18,000 and 23,000 word pairs, respectively. We used a variety of approaches to BLI in order to generate a larger lexicon. The bilingual lexicon project was carried out under my supervision within the NLTPI.[25] The aim of the project was to build a highly accurate glossary list. As different BLI approaches have different merits, we wanted to compare the effectiveness of multiple different methods and see if they could be used jointly in order to increase accuracy.

We designed a number of experiments to explore three research questions: 1) How accurately can we produce equivalence pairs using four different methods: using cross-lingual word embeddings trained on comparable corpora, pivoting through available dictionaries, mining bitexts using word alignments, and translating using available MT systems? 2) To what extent does the frequency of words affect the results in corpus-based approaches? 3) How can we best combine the different approaches to increase accuracy while not reducing the size of the resulting lexicon too much?

3.6.1 Related Work

A wide range of approaches to BLI have been shown to be effective. Bilingual lexicons have been mined from parallel corpora using word alignments (Mihalcea and Pedersen, 2003;

[25] Others working on the project were Luke O'Brien who executed word alignments using a combination of word alignment tools and CombAlign (Section 3.2), and Finnur Ágúst Ingimundarson, Árni Davíð Magnússon, Þórdís Dröfn Andrésdóttir and Inga Guðrún Eiríksdóttir who evaluated translation candidates. All other work was carried out by me.

Vulić and Moens, 2012), and from comparable corpora with various methods, in recent years commonly by learning cross-lingual word embeddings (Lample et al., 2018; Rapp et al., 2020). Artetxe et al. (2019) use an unsupervised MT system to create a synthetic corpus from which they extract a lexicon. Comparable corpora can also be exploited by identifying word pairs in the corpus using word alignments. For this purpose, sentence pairs first have to be extracted from the comparable corpora. This has been carried out using a range of approaches. Bouamor and Sajjad (2018) produced candidate lists using bilingual sentence level embedding models and cosine similarity, filter the lists using a threshold based on sentence level BLEU, with an MT generated translation as a reference and finally select the final pairs using a classifier exploiting features such as part-of-speech, named entities and sentence length. Feng et al. (2022) use a BERT model to generate a similarity score based on contextualized sentence embeddings and, as we do in one of our experiments, discussed in Section 6.2, by using CLIR to limit the search space and a classifier, based on a word alignment score and a contextualized embedding score, to select the sentence pairs (Steingrímsson et al., 2021b).

Shi et al. (2021) show that the performance of lexicon induction from bitexts correlates with bitext quality, although they are still able to induce a reasonably good bilingual lexicon from their lowest quality bitexts. They also observe that a better word aligner, in terms of F_1-score, usually leads to a better induced lexicon.

It is also common to pivot through existing dictionaries to infer translations between two languages using an intermediary language, as we did with PivotAlign (Section 3.3). The problem has also been approached by using MT systems to translate the words between languages (Arcan et al., 2019).

3.6.2 Methodology

We based our experiments on four different approaches: 1) Using word alignments to extract candidate pairs from bilingual corpora, i.e. parallel or comparable texts in two languages; 2) Pivoting through existing dictionaries; 3) Using MT; and 4) Using cross-lingual word embeddings. Each experiment resulted in a list of translation candidates from which we extracted a random sample for evaluation.

The evaluation was carried out by first comparing the list against the following manually curated Icelandic–English/English–Icelandic dictionaries and word lists: English–Icelandic Wiktionary and Apertium dictionaries, titles of common pages in the Icelandic and English Wikipedia, the Icelandic Term Bank,[26] and the Terminology Database of the Ministry of Foreign Affairs.[27]

If the candidate pairs were found in these data sets they were automatically accepted. Other candidate pairs were divided between five human annotators who manually evaluated them. Each candidate pair was only evaluated by one person. The annotators were all Icelandic native speakers, educated in linguistics and with excellent knowledge of English. The annotators were to categorize a pair as *acceptable* if the word in either language could be translated to the other word, in any environment the annotators could think of. The *rectifiable/partial* category was used when there was a minor error in one of the words, e.g. a spelling error, lemmatization error or a typo, or when a word in one language had to be translated into a multiword unit, and the translation given only has a part of that unit. Words that fell into neither of these categories were categorized as *unacceptable*.

[26] `https://idordabanki.arnastofnun.is/`

[27] `https://hugtakasafn.utn.stjr.is/`

Translation Pair		Probabilities	
Icelandic	**English**	**is→en**	**en→is**
ananas	pineapple	1.0	0.82
ananasjurt	pineapple	1.0	0.15
granaldin	pineapple	1.0	0.03
regnhlíf	umbrella	0.70	0.73
regnhlíf	brolly	0.30	1.0
hlífð	umbrella	0.02	0.01
sólhlíf	umbrella	0.31	0.26
sólhlíf	parasol	0.48	1.0
sólhlíf	sunshade	0.21	0.46

Table 3.13: Example of translation pairs with probability scores from the lexicon resulting from the project. If there is only one translation for a word, the probability is 1.0, if there are many translations the probabilities sum to 1.0, as for the English word *pineapple* or the Icelandic word *regnhlíf*.

Our work resulted in a manually verified lexicon of over $232,000$ pairs, with a probability score attached to each pair for both translation directions. The probability scores are an attempt to order the translations for a given source word from most common to least common. The probability is based on relative frequency, calculated by tallying the number of times the pair was suggested by our methods and comparing that to how often other translations for the same word were suggested. An example of the lexicon format is shown in Table 3.13.

3.6.3 Extracting Candidate Pairs from Bilingual Corpora

We use six different bilingual corpora. One parallel corpus: ParIce (version 1, see Section 3.5). Three collections of parallel sentences extracted from comparable corpora: WikiMatrix (Schwenk et al., 2021) and two versions of ParaCrawl (Bañón et al., 2020). Furthermore, we use two synthetic corpora with back-translated source sentences on one side and human-written target sentences on the other. These corpora were generated by translating monolingual data, one by translating Icelandic data into English and the other by translating English data into Icelandic.[28]

We extracted accurate word alignments using six word alignment models and CombAlign (Section 3.2). If four out of six models agreed on an alignment, it was accepted. In order to increase alignment accuracy and to reduce noise, we lemmatized all the data and collected lemma pairs from the lemmatized sentence pairs. We used spaCy for lemmatizing English, and after PoS-tagging the Icelandic texts using ABLTagger (Section 3.1), we lemmatized them using Nefnir (Ingólfsdóttir et al., 2019). We then calculated a confidence score for each aligned word pair (see Equation 3.1 in Section 3.3.1). By evaluating samples of 250 candidate pairs at different levels of the confidence score, we found cutoff thresholds for each of the bilingual corpora we used.

As Figure 3.6 shows, the acceptance ratio differs between different corpora. This has to do with both the smoothing mechanism and the nature of the different corpora. The smoothing mechanism is intended to raise the score for rare occurrences if a corpus is small, but lower the confidence as the corpus gets larger. But if a rare occurrence is repeated in proportion to the growth of the corpus, the confidence score is raised. An example could be a wrong translation extracted four times from a corpus of four million sentence pairs,

[28] The synthetic corpora are available here: `http://hdl.handle.net/20.500.12537/70`

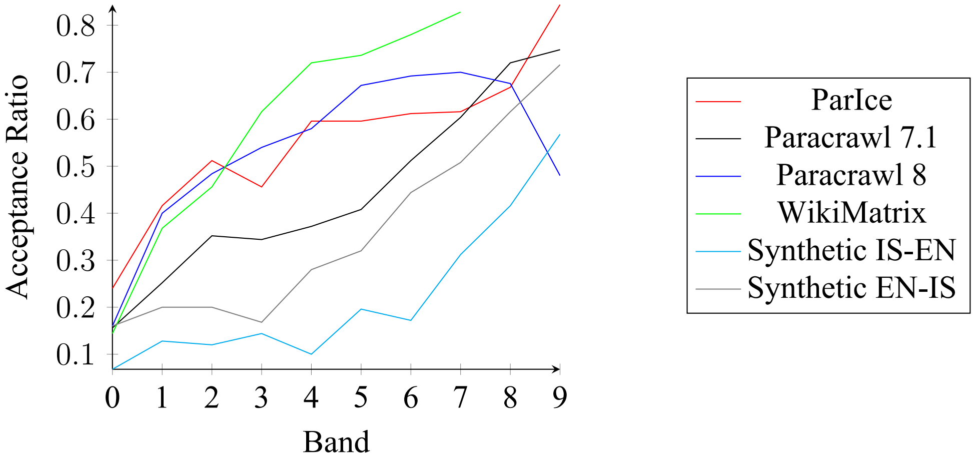


Figure 3.6: Bilingual corpora. Manually evaluated acceptability of candidate pairs at different bands of confidence, as automatically assessed by our confidence score.

with eight co-occurrences of the words in sentence pairs in the corpus. This pair would obtain a confidence score of $4/(8 + \log_2(4,000,000)) = 0.13$. The same proportion of alignments and occurrences in a corpus of 50 million sentence pairs, with the erroneous pair extracted 50 times and the words co-occurring 100 times, would give the a score of $50/(100 + \log_2(50,000,000)) = 0.4$. In a genuine parallel corpus, this would probably be the correct assumption, but in a synthetic corpus, containing back-translations on one side, it might not be. The synthetic corpora may not have as rich a vocabulary and they sometimes generate made up words or get the inflections incorrect. In a large synthetic corpus these patterns are more likely to repeat than in a genuine parallel corpus.

3.6.4 Pivoting

We used dictionaries with Icelandic as the source language and pivoted through an intermediary language into English. For collecting translations from Icelandic into intermediary languages, we used the ISLEX (Úlfarsdóttir, 2014) and LEXIA dictionaries (Icelandic–Danish / Swedish / Norwegian / Finnish / French) and dict.cc[29] for Icelandic–German. For collecting translations from the intermediary languages into English, we used Apertium (Forcada et al., 2011) (Finnish / French / Norwegian / Swedish-English) and dict.cc (German/Finnish/Norwegian/ Swedish/French/English). For each Icelandic source word, we collected all possible translations in the intermediary languages and, for each of the intermediary translations, we collected all English translations, as we did previously with PivotAlign (Section 3.3).

We compiled candidate lists for each of the intermediary languages, using both Apertium and dict.cc to obtain English translations from the intermediary language words. A random sample from each list was evaluated and the acceptance ratio calculated. As seen in Table 3.14, up to 76% of the translations are rated acceptable, depending on the intermediary dictionary and language used. In order to increase the accuracy even further, we can require a pair to be suggested by two or more pivoting paths. While this raises the accuracy it produces fewer candidate pairs.

[29] `https://www.dict.cc/`

	Apertium		dict.cc	
	acc. ratio	**no. pairs**	**acc. ratio**	**no. pairs**
se	0.64	34,915	0.76	26,622
fi	0.43	214,659	0.75	19,304
no	0.53	15,261	0.74	31,213
fr	0.63	20,865	0.64	39,590
de			0.54	137,970

Table 3.14: Pivoting. Acceptance ratio and number of pairs yielded by pivoting from Icelandic to English via an intermediary language in ISLEX and the Apertium and dict.cc dictionaries.

	Opus	**M2M**	**Google**	**MS**	**no. pairs**
is			0.59	0.60	53,151
da	0.52		0.59	0.63	80,074
sv	0.56	0.32	0.65	0.65	69,884
fi	0.53	0.27	0.66	0.62	62,876
no			0.59	0.61	66,129
fr	0.56	0.35	0.67	0.71	48,533

Table 3.15: Machine translation. Acceptance ratio in 250 randomly selected candidate pairs for each language and system. For all languages except Icelandic, we pivoted through intermediary languages using dictionaries and translated the intermediary languages to English using MT.

3.6.5 Machine Translation

Our most simple approach was translating words into English using four available MT models: Google Translate,[30] Microsoft Translator,[31] OPUS-MT (Tiedemann and Thottingal, 2020) and M2M-100 (Fan et al., 2021). First, we translated the Icelandic source words of the ISLEX/LEXIA dictionaries into English, thereby creating a candidate list. Second, we also translated into English the target language equivalents in the same dictionaries, Danish, Swedish, Norwegian, Finnish and French, and then paired the source Icelandic word to the MT output for the target words.

While this method is simple and accessible for many languages, using existing commercial MT services can make it difficult to replicate the results of the experiments. As one of our goals was to compile a lexicon at minimal cost, we decided to use these services anyway to see if they could be useful for this purpose.[32]

All the systems except M2M-100 resulted in over 50% acceptable translations for all languages. A high acceptability rate should be expected as each model only produces one translation for each word, often selecting a common translation. When doing word alignment on parallel corpora or when pivoting through intermediary dictionaries more examples are produced and thus we may expect to see a higher number of rare translations using those approaches. Out of the different translation engines, Microsoft Translator gave the best results, as shown in Table 3.15, and for all translation engines, translating through French gave the best results, even better than translating straight from Icelandic to English.

[30] `https://translate.google.com/`

[31] `https://translator.microsoft.com/`

[32] The translation services, Google Translate and Microsoft Translator, were used via an API in May 2021.

Translation Direction	Retrieval method	Classification		
		High	Medium	Low
en-is	NN	0.39	0.20	0.03
	CSLS	0.59	0.38	0.14
	freq.	0.71	0.50	0.14
is-en	NN	0.48	0.26	0.15
	CSLS	0.63	0.40	0.19
	freq.	0.67	0.44	0.22

Table 3.16: Cross-lingual word embeddings. Acceptance ratio for candidate lists in different similarity or frequency classes, for each of the methods employed.

3.6.6 Cross-lingual Word Embeddings

Icelandic news texts collected from the Icelandic Gigaword Corpus (Steingrímsson et al., 2018) and English news texts collected from Newscrawl[33] were used to train two word2vec models (Mikolov et al., 2013), one for English and the other for Icelandic. VecMap (Artetxe et al., 2018) was then used to build cross-lingual word embeddings by mapping the models to a common vector space.

Three candidate lists were generated. One is a list of the most frequent English and Icelandic words in their respective corpora, with the target word being the nearest neighbour (NN) in the target language, as calculated by cosine distance in the common vector space. For the second list the nearest neighbour is found for all words represented in the word embeddings, and the ones with the lowest cosine distance to a word in the other language, i.e. NN, are selected.

Dinu and Baroni (2015) show that few vectors in the embedding space tend to be nearest neighbours of many other points, pushing the correct ones down the neighbour list. This is called the hubness problem. The Cross-domain Similarity Local Scaling (CSLS) method, proposed by Lample et al. (2018) alleviates this problem by subtracting the mean similarity of the source embedding to its target neighbourhood and the mean similarity of the target embedding to its source neighbourhood, from twice the cosine distance between points in the hubs. This expands the space where there is a high density of points, increasing accuracy. For our third list we look for the lowest distance as for the second list, but using CSLS instead of NN.

For each of these approaches, we divided the results into three classes: *High*, for the top 2,000 pairs, *Medium*, for the next 8,000 pairs, and *Low* for the 90,000 pairs after that. Table 3.16 shows that while we obtain decent scores for the most frequent words in the corpora and the most similar ones in terms of the model and scoring methods, the results fall sharply as word frequency and similarity decrease. This approach thus seems to be the least useful one.

3.6.7 Combining Different Approaches

Based on the results presented above, we created two lists. One containing all candidate pairs obtained through either pivoting or MT, being in classes where the acceptance rate for the manual evaluation was above 50%. The other list was created from all six bilingual corpora, but only from confidence bands with over 50% acceptance rate (see Figure 3.6). Taking an

[33] `https://data.statmt.org/news-crawl/en/`

		Evaluator 1			
		C	P	I	Total
Evaluator 2	C	883	20	23	926
	P	8	15	6	29
	I	17	0	28	45
	Total	908	35	57	1000

Table 3.17: Evaluation matrix for the two evaluators. C=Correct; P=Partially correct; I=Incorrect.

intersection of these two lists resulted in a list of 29,609 candidates, of which 93.2% were accepted after manual evaluation. Furthermore, if the confidence bands are ignored and the second list contains all pairs from the six bilingual corpora, and is used to filter the results of the previous list in much the same way we do with PivotAlign (Section 3.3), this results in a list of 57,818 candidates, of which 84.1% were accepted.

3.6.8 Results

We applied and compared four different approaches to automatically compile an English-Icelandic bilingual lexicon. We showed that by using a combination of bilingual corpora, pivoting and MT approaches, we can build a highly accurate candidate list for lexicon translations between languages. While using an unsupervised approach such as cross-lingual word embeddings did not result in many useful candidate pairs, extracting candidate pairs from the back-translated synthetic corpora using word alignments did give promising results.

Our resulting lexicon contains $232,950$ pairs, with $105,442$ different Icelandic lexical items, of which $84,812$ are single words and $20,630$ multiword units, and $116,744$ different English items, of which $45,147$ are unique English words and $71,597$ multiword units.[34] The published dataset includes the probability scores described in Section 3.6.2 and word class information, in cases where that could be retrieved automatically from Wiktionary or the DMII. The published dataset also contains information on which methods produced the pairs included in the dataset and how often. More detailed results and description of the process can be found in Steingrímsson et al. (2022).

A random sample of $1,000$ pairs in the final lexicon was evaluated independently by two of the annotators, me and Finnur Ágúst Ingimundarson. As before, the pairs were evaluated to be in one of three categories: acceptable, unacceptable, rectifiable/partial. Results are shown in Table 3.17.

The two annotators agreed on a correct translation 92.6% of the time. We calculated the Cohen's Kappa coefficient (Cohen, 1960) to measure inter-rater reliability between the two.

$$\kappa = \frac{P_o - P_e}{1 - P_e}. \tag{3.4}$$

Equation 3.4 uses P_o, the proportion of observed agreement and P_e, the proportion of chance agreement to calculate a coefficient that accounts for the possibility of the raters guessing at the variables due to uncertainty. It can range from -1 (no agreement) to +1 (perfect agreement). When κ is equal to 0, the agreement is the same that might obtained by chance. If it is negative, it is less than the agreement expected by chance. While there is some debate on how to interpret the kappa-value for subjective labels, such as fair, moderate or

[34] Available at `http://hdl.handle.net/20.500.12537/144`

substantial agreement (see e.g. Landis and Koch (1977), Viera and Garrett (2005), McHugh (2012)), Fleiss (1973) argues, in line with most other work, that values between 0.40 and 0.75 may be taken to represent fair to good agreement beyond chance. Our calculated κ of 0.52 falls in that range, indicating that the inter annotator agreement is reasonable.

After evaluating independently, the two annotators had another look at the pairs they disagreed upon and tried to come to a common conclusion for each of them. This final evaluation of the lexicons quality resulted in 91.6% correct pairs, 4.1% incorrect and 4.3% which were almost correct and needed a minor amendment. The ‘almost correct’ ones were most often incorrectly lemmatized word forms, nouns in an oblique case or lemmas containing spelling errors, and thus not suitable for being added to the lexicon without correcting.

3.7 Conclusion

We have introduced the data and tools we created to facilitate our research on how best to process data in order to make the best use of bilingual data for training MT systems. In the following chapters, we will work further with the data and methods introduced here to align parallel corpora on a sentence level, filter parallel sentence pairs, and use data that is traditionally discarded in these processes as well as comparable corpora. In the next chapter, we will start with filtering.

Chapter 4

Filtering Parallel Corpora

In the previous chapter, we provided an overview of the tools and datasets we have developed in order to be better able to study different approaches to compiling parallel training data for English–Icelandic MT. In this chapter, we will look into the effectiveness of different filtering approaches and whether the dataset and/or language direction matter when choosing a filtering approach.

Detrimental segments, in our context, are sentence pairs in training data that may degrade the performance of an MT system trained on that data. Filtering parallel data for MT, the task of removing possible detrimental segments from the data, is usually performed by applying rules and sometimes a scoring mechanism or a classifier. The aim is then to remove pairs most likely to weaken performance of an MT system in some way, usually in terms of translation quality as measured by automatic metrics.

We work with the English–Icelandic language pair, and raw data from two parallel corpora, ParIce and ParaCrawl (Bañón et al., 2020), with the goal of minimising data detrimental to translation performance while losing little or no useful data from the original texts, thus building a data set better suited for MT training. We experiment with basic shallow filters, scoring mechanisms and classifiers based on some of the scores, as well as other classifiers not dependent on extrinsic scoring mechanisms. We investigate whether lemmatizing the morphologically rich Icelandic texts helps to increase filtering accuracy, and whether we can further improve the datasets by using multiple classifiers in combination. The work described in this Chapter is also partly presented in Steingrímsson et al. (2023).

Recent literature on parallel corpus filtering has largely focused on filtering noisy data collected from the web. This was, for example, the objective of the parallel filtering shared tasks at WMT 2018–2020 (Koehn et al., 2018, 2019, 2020). We want to inspect whether we should apply the same filtering approaches to noisy datasets and to cleaner parallel corpora compiled from document pairs where one document is known to be a translation of the other, or where both are a translation of a third original text. Finally, for a given dataset, the same training data is usually used for training both translation directions, src→trg and trg→src, instead of filtering especially for each translation direction. We want to investigate if whether filtering separately for each translation direction is likely to bring improvements in a downstream MT task.

Our aim in this chapter is thus to find out how to filter parallel corpora so as to compile a training set that potentially gives the best results when used to train an MT system. We seek to answer our first two research questions, the first one being a more general search for good filtering approaches: **RQ1: How can we filter parallel corpora to minimize noise, and still lose little or no useful data from the original texts?** And the second question being more specific about whether something is gained by adapting the filtering approach more to the problem at hand: **RQ2: To what degree should we consider filtering parallel corpora for MT training to be independent of the dataset and languages being filtered, and the intended translation direction of the MT system being built?** In order to answer these questions, we build MT models for both translation directions and multiple different filtering approaches for each one, and evaluate the results, both manually and automatically.

4.1 Related Work

In their paper, Khayrallah and Koehn (2018) specify five general classes of noise commonly found in the German–English part of the Paracrawl corpus: misaligned sentences, disfluent text, wrong language, short segments, and untranslated sentences. They find this distinction to be useful to give a good general idea of which types of errors seem to have the least impact on MT systems (short segments, untranslated source sentences and wrong source language) and which have the most dramatic effect (untranslated target sentence). In the paper, misalignments, misordered words, and wrong language, in source or target texts, are also shown to be harmful, but not as harmful.

As this classification is rather coarse, some variation can be expected within each class; a misalignment in one sentence pair does not have to be equivalent to a misalignment in another sentence pair. Briakou and Carpuat (2021) focus on fine-grained semantic divergences within mostly equivalent pairs (pairs of words, phrases or sentences that have similar meanings and connotations), instead of looking at broader and perhaps more coarse definitions of noise as Khayrallah and Koehn (2018) define it. An example given in the paper is fr: "votre père est français" → en: "your parent is french", where the correct translation should be: "your father is french". These fine-grained divergences can be found in even high-quality parallel corpora. For lexical substitution the authors of the paper corrupt equivalents by substituting words with their hypernyms or hyponyms. For phrase replacement they replace sequences of words with phrases of matching PoS tags and for subtree deletion they randomly delete subtrees in the dependency parse tree of either source or target. They find that the divergences cause degradation on the MT output of a system trained on the data, as measured by BLEU and METEOR, and that divergences impact model confidence in their predictions. Corrupting training data by lexical substitution causes the largest degradation in MT output and subtree deletion the least. Nevertheless, the impact of divergences seem to be smaller than that of noise. They argue that this suggests that noise-filtering techniques are suboptimal to deal with fine-grained divergences. While these are recent papers, the call to better quality data for MT training is not only confined to NMT. Ozdowska and Way (2009) observe that the original source language has considerable impact on French–English phrase-based SMT, with a decrease in translation quality if the original source language was neither French nor English, and best results when the original language is the source language being translated. They argue that better quality data is more important than more data, and that more attention has to be paid to the role of the human translator.

A wide array of approaches for parallel data filtering has been employed. Early work used the IBM models (Brown et al., 1993) for word alignment. For example, Khadivi and Ney (2005) filter out the noisy part of a corpus based on IBM models 1 and 4 and length-based models, and score the alignments on a linear combination of these. Taghipour et al. (2011) do outlier detection and show that their filtered corpus results in improved translation quality as measured by BLEU, even though sentences have been removed. Sarikaya et al. (2009) use context extrapolation to boost the sentence pair coverage, checking whether the distance of the sentences from an anchor point is the same, and whether the sentences have the highest similarity score compared to other pairs within a window, despite being below a defined threshold. In an early work on filtering web-scraped parallel corpora, Rarrick et al. (2011) filter machine-translated content from web-scraped corpora employing a maximum entropy classifier, using a variety of sentence-level and document-level features, and show that it is possible to improve performance of an MT system by removing large amounts of training data, challenging conventional wisdom at the time that more data is better data.

Cross-lingual word embeddings have been used to calculate distance between equivalences in different languages (Luong et al., 2015a; Artetxe et al., 2016). Defauw et al. (2019) treat filtering as a supervised regression problem and show that Levenshtein distance (Levenshtein, 1966; Wagner and Fischer, 1974) between the target and MT-translated source, as well as cosine distance between sentence embeddings of the source and target, are important features. While they use InferSent (Conneau et al., 2017), in more recent work BERT has been employed for calculating crosslingual semantic textual similarity to detect misalignment with good results (Lo and Simard, 2019).

In the three shared tasks on parallel corpus filtering at the WMT (Koehn et al., 2018, 2019, 2020), some very promising tools and approaches were submitted. Methods based on crosslingual sentence embeddings trained from parallel sentence pairs did well, such as Chaudhary et al. (2019) based on LASER and Artetxe and Schwenk (2019a) which uses a BiLSTM model. Both papers tackle the scale inconsistencies of cosine similarity, the problem that cosine similarity is not globally consistent and that potentially different scales of target candidates for a given source sentence may affect their relative ranking, causing the hubness problem discussed in Section 3.6.6. They do that by considering the margin between a given sentence pair and its closest candidates to normalize the similarity scores. Zipporah (Xu and Koehn, 2017) uses probabilistic translation dictionaries, language models and a logistic regression model trained to classify sentence pairs. Noisy data is synthesized and used as negative samples in training. Bicleaner (Sánchez-Cartagena et al., 2018) uses a set of handcrafted hard rules to detect flawed sentences and then proceeds to use a random forest classifier based on lexical translations and several shallow features such as respective length, matching numbers and punctuation. It also scores sentences based on fluency using 5-gram language models. The tool ranked highly on the first two parallel corpus filtering tasks at WMT. Bicleaner AI (Zaragoza-Bernabeu et al., 2022) is a fork of Bicleaner using a neural classifier. It has been shown to give significant improvements in translation quality as measured by BLEU when used for filtering training data for multiple language pairs, as compared to the previous version of the tool.

Herold et al. (2022) revisit the noise classes specified by Khayrallah and Koehn (2018) to investigate how accurately two of the strongest filtering approaches to date, according to them, cross entropy (Rossenbach et al., 2018) and LASER, can filter out different noise classes. They find that for a common language pair, German–English, most types of noise can be detected with over 90% accuracy, although misalignments and poor synthetic translation can only be detected with an accuracy of less than 70%. For a less common language pair, Khmer–English, the performance of the filtering system degraded significantly and the accuracy of identifying noise was lowered by 8–19%, depending on the type of noise.

4.2 Experiments

In order to answer the research questions set out in the beginning of this chapter, we compare a number of approaches and scoring mechanisms when applied to a set of sentence pairs derived from web-crawled corpora, on the one hand, and from parallel corpora compiled from known parallel documents, on the other. For each approach, a sample of the data is manually evaluated using the taxonomy developed by Kreutzer et al. (2022) to gain an understanding of what sort of data each approach and scoring mechanism filters out. We then train MT systems using datasets filtered using different filtering approaches, and compare the quality of the resulting systems in terms of BLEU score. Furthermore, the scores are compared to the output of the unfiltered systems described in Section 3.5.4. We measure BLEU scores on the test set provided for the English–Icelandic language pair in the WMT 2021 shared task (Akhbardeh et al., 2021), using SacreBLEU (Post, 2018).

4.2.1 Data Sets

The two different data sets we use for our experiments both contain English–Icelandic parallel sentences: ParaCrawl and ParIce. We carry out the same experiments using both corpora and compare the results in order to answer our research question about whether the same methods work as well for both types of data and both language directions, or whether different approaches should be considered depending on the data set and/or intended translation direction.

ParaCrawl

ParaCrawl is compiled from web-crawled data. Based on the evaluation by Kreutzer et al. (2022), approximately 76% of sentence pairs are acceptable mutual translations, on average, in 21 language pairs from the ParaCrawl 7.1 datasets cleaned for publication. This is still substantially higher than for two other common datasets compiled from web-crawled data and also evaluated in the same paper, CCAligned (El-Kishky et al., 2020) and Wiki-Matrix (Schwenk et al., 2021). Nonetheless, there is high variance between languages and low-resource datasets tend to have be of the lowest quality as judged by human annotators. Rikters (2018) inspects the quality of the first version of ParaCrawl and filters out 85% of the English–Estonian Paracrawl dataset. Although it should be noted that there are differences in noise ratio between different versions of the corpus, it is clear that for ParaCrawl to be of good use for training MT models it has to be filtered thoroughly, especially in the case of languages where the corpus quality has not been evaluated. This has also been emphasized by the results of the three shared tasks at the WMT focusing on filtering parallel corpora. In our work, we start with the raw data from version 9 of the corpus, consisting of 65,373,727 sentence pairs in total. Our goal is to extract from the corpus sentence pairs useful for training MT systems on its own or to complement other data sets, and leave out sentence pairs likely to be detrimental.

ParIce

The English–Icelandic parallel corpus ParIce differs from ParaCrawl in that it is compiled from known parallel documents, which have been aligned at the sentence level. We work with the 21.10 version of the corpus (Steingrímsson and Barkarson, 2021). It is available unfiltered, accompanied with semantic similarity scores for each sentence pair and flags indicating whether it is recommended to filter out the pair or not. The corpus is described in more detail in Section 3.5.

4.2.2 Filtering and Scoring Tools

In order to decide which sentence pairs are useful and which ones should be filtered out, we use an array of tools for scoring sentence pairs to find the highest-quality data within the corpora. We start with shallow filters, mostly rule-based, to remove pairs that are very likely to be noise, and then proceed to run different tools, both made available by others and of our own device.

Shallow Filters

In order to remove duplicates, near duplicates and other redundant data, we apply a few shallow filters. Most of these are rules inspired by Pinnis (2018), who applies 17 different filters in his work. We do not use all his filtering approaches but select the ones likely to remove the highest portion of detrimental pairs as outlined by Khayrallah and Koehn (2018). Our shallow filters are:

1. ParaCrawl Zero Score: ParaCrawl is distributed with *Bicleaner* scores for each sentence pair. We remove all pairs where this score is 0, the lowest possible.

2. Minimum Sentence Length: If both source and target sentence have 3 tokens or less, the pair is discarded. Khayrallah and Koehn (2018) show that very short sentences can have a small detrimental effect on MT translation quality.

3. High Overlap: We remove all pairs where 60% or more of the tokens in one language are also present in the other language.

4. Symbol Filter: At least 70% of characters in both sentences should be alphabetical, when whitespace has been removed. Otherwise the pair is discarded.

5. Language Filter: We use the *fasttext* model (Joulin et al., 2017) to identify the language of each sentence. The model predicts the two most likely languages for each sentence, and if the expected language is not one of them, we discard the pair.

6. Similar Pairs: Our last two shallow filters remove near-duplicates. Lee et al. (2022) show that removing such segment pairs allows for training models that require fewer training steps to achieve the same or higher accuracy. Our first step for this is to remove all spaces and symbols, except for alphabetical letters, and create one string from both segments. We find all pairs with identical such strings and keep only the one with the highest *Bicleaner* score.

7. Similar Segments: In our second step for removing near-duplicates, we start by calculating four scores for each pair, *WAScore*, *LaBSE*, *NMTscore* (Vamvas and Sennrich, 2022) and *LASER*. (See more details about the scores in Section 4.2.2). Then all segments are tokenized. Tokens starting with a capital letter or containing non-alphabetical letters are removed. This removes most named entities and the resulting string should thus primarily include other content words and function words. We create these strings for all sentences, in the source and target language. For each of these strings, we search for identical strings in the same language and select only one pair for each string, the one with the highest *Bicleaner* score. The aim of this step is to reduce the number of sentences that are identical, except for proper nouns or numbers, a common phenomenon in ParaCrawl. A large portion of such *almost* identical segments in the corpus seem to originate from tourism websites, commonly referring to hotels, locations or companies.

While all filters are applied to the ParaCrawl data, all except the first one are applied to the ParIce corpus as the three scores published with each sentence pair in that corpus, LASER, LaBSE and WAScore, almost never receive the value 0. For the sixth filter, we use these three scores published with ParIce, instead of a Bicleaner score.

Bicleaner Models

Bicleaner (Sánchez-Cartagena et al., 2018; Ramírez-Sánchez et al., 2020) is an open-source noise filter and classification tool to clean parallel corpora. It was released as part of the ParaCrawl project and is used to compile the clean datasets for ParaCrawl. Bicleaner uses a set of hard rules for pre-filtering, n-gram language models for fluency scoring, and a random-forest classifier to produce a probability score using features such as lexical similarity and shallow properties like sentence length, punctuation and capitalization. Bicleaner AI (Zaragoza-Bernabeu et al., 2022) the Bicleaner fork, uses a fine-tuned XLM-RoBERTa classifier to produce probability scores by training it on positive samples from existing parallel corpora and negative samples which are created by corrupting the same positive samples. In synthesising the noise, the tool tries to emulate errors commonly introduced by sentence segmentation and alignment.

We use two publicly available Bicleaner models for English–Icelandic, version 1.5 of the original Bicleaner and Bicleaner AI 1.0 full model. In addition, we train two new models using Bicleaner v0.15.2, one that classifies lemmatized data and the other unlemmatized. We follow the instructions provided in the Bicleaner repository when training the models.[1] For training each model, we need word frequency information, probabilistic dictionaries and a parallel training corpus.

En–Is: Retrained

We create word frequency lists from monolingual corpora. For Icelandic we use the 20.05 version of The Icelandic Gigaword Corpus (IGC) (Steingrímsson et al., 2018) and for English we use News Crawl (Barrault et al., 2020) with data from 2012–2019. This resulted in lists of 6.5M unique Icelandic word forms and 2.8M unique English word forms. For a probabilistic dictionary, we use the bilingual lexicon described in Section 3.6.

The lexicon contains 232,950 pairs of both single words and multiword units. All single word units are lemmatized. For training the Bicleaner model, we only use pairs of single words, approximately 145,000 pairs in total. The published dictionary contains probabilities for each equivalency pair. As we have removed all pairs containing multiword units, we recalculate the probabilities so the probabilities of all translations for each word add up to 1. This results in a dataset of approximately 52,000 unique Icelandic words and 44,000 unique English words with one or more translations and a probability assigned to each one.

For a bilingual training corpus, we extract sentence pairs from the 21.10 version of ParIce. The corpus is published with LaBSE, LASER and WAScore calculated for each sentence pair, as described in Section 3.5. We calculated an average of these three scores and used the highest-scoring 250,000 sentence pairs for training.

En–Is Lemmatized

As our probability dictionary is lemmatized, and the proportion of unlemmatized word forms is very high in running Icelandic texts, we want to investigate whether training a Bicleaner model with lemmatized data would improve the results. We train a new model using the same datasets as before, but with the Icelandic IGC and English News Crawl lemmatized before creating the word frequency lists, and the 250,000 sentence pairs extracted from ParIce lemmatized before training. For lemmatizing English texts we use spaCy, and for Icelandic texts

[1] `https://github.com/bitextor/bicleaner/wiki/How-to-train-your-Bicleaner`

we first tag them using ABLTagger and then proceed to lemmatize using Nefnir (Ingólfsdóttir et al., 2019). The same tools are used to process the data to be classified using the model.

Scoring

Various scoring mechanisms have been developed to automatically assess the quality of bilingual sentence pairs in parallel corpus filtering. We employ six approaches to score and manually evaluate sentence pairs and compare these mechanisms in filtering out data. We use the scores to train classifiers. We also experiment with using them to set cut-off rates for adding and removing sentence pairs from the final datasets.

LASER (Artetxe and Schwenk, 2019b), is an architecture to learn joint multilingual sentence representations. It uses a single BiLSTM encoder with a shared BPE vocabulary for all languages. This is coupled with an auxiliary decoder and trained on parallel corpora. LASER can be applied to various tasks, e.g. multilingual similarity search (Artetxe and Schwenk, 2019b) and bitext mining (Artetxe and Schwenk, 2019a). It was trained on 93 languages, including Icelandic and English.

LaBSE (Feng et al., 2022) is a model trained and optimized to produce similar representations for bilingual sentence pairs. It uses dual encoder models, with the encoder architecture following the BERT Base model, and additive margin softmax which extends the scoring function in the model by introducing a large margin around positive pairs, improving the separation between translations and nearby non-translations (Yang et al., 2019). Monolingual and bilingual data are used to pre-train the model with a masked language model (Devlin et al., 2019) and a translation language model (Conneau and Lample, 2019), respectively. A publicly released pre-trained model was trained on 109 languages, including Icelandic and English.[2] Feng et al. (2022) show the LaBSE model to give state-of-the-art results on a number of bitext retrieval tasks.

NMTScore (Vamvas and Sennrich, 2022) is based on *translation cross-likelihood*, the likelihood that a translation of segment *A* into some language, could also be a translation of segment *B* into the same language. An example could be the translation of the French 'Bonjour!' into the Swedish 'Hej!'. To calculate translation cross-likelihood, the French segment would first be translated to a third language, say English, and the score is based on the probability of the model getting the same translation for the Swedish segment. The score is symmetrized by averaging the translation probabilities in both directions. We use the M2M100 multilingual translation model (Fan et al., 2021) to calculate NMTScore in our work.

WAScore is the word alignment-based score we devised to measure word-level parallelism, described in Section 3.4.

Furthermore, we experiment with additional **translation-based measures**, which do not use any resources other than the dataset to be filtered. We do that by training NMT models, for both translation directions, on the corpus to be filtered. We then use these models to translate the training dataset. The assumption is that incorrect pairs in the corpus are irregular and arbitrary and thus a model trained on data, albeit noisy, would in translation not create the same errors, while translations of sentences in pairs of mutual translations would at least to some extent be close to the original. For each sentence in the training corpus, we create five translations and calculate two types of scores. ChrF (Popović, 2015) is more forgiving than BLEU when it comes to minor errors due to morphology, and correlates better with human judgement on the segment-level (Ma et al., 2019). We therefore opt to calculate ChrF for

[2] `https://huggingface.co/sentence-transformers/LaBSE`

all five translations and select the highest score. We do this for both translation directions. We also do a simple comparison of the tokens in the target sentences against tokens in the five most likely translations from the source, and calculate the ratio of overlapping tokens between the target sentence and the translated source sentences. Again we select the highest score of the five and call this *BOWScore*, for bag-of-words score. We calculate separate scores for each translation direction, four scores in total.

Three Score-based Classifiers

Using four of the scores described in the previous section, LASER, LaBSE, NMTScore and WAScore, we train three different classifiers to determine whether the sentence pairs are useful or not. We adapt a training set we compiled for a classifier used in mining comparable corpora (Steingrímsson et al., 2021b). The dataset was compiled of 50,000 randomly sampled non-parallel pairs from two monolingual news corpora for negative examples. We use these and 1,000 parallel segments selected from the Icelandic part of the Parallel Universal Dependencies (PUD) treebanks.[3] LASER, LaBSE, NMTScore and WAScore were calculated for all 51k sentence pairs and used to train the classifiers. We used scikit-learn (Pedregosa et al., 2011) to train random-forest, support vector machine and logistic regression classifiers.

A random-forest classifier is an averaging algorithm that combines a number of decision tree classifiers, fitted on sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. Using a random selection of features to split each node, they become more robust with respect to noise (Breiman, 2001). In our training, we use the scikit-learn default number of estimators, i.e. number of trees in the forest = 100.

Support vector machines (SVM) (Cortes and Vapnik, 1995) use multiple features to separate two classes by a hyperplane. We use a support vector classifier using a radial base function kernel that allows some observations to be on the incorrect side of the hyperplane to be able to generalize better on the data.

Logistic regression (Cox, 1958) uses maximum likelihood to fit a model applying logistic functions to the training dataset. We use the default settings in scikit-learn.

For all sentence pairs we want to classify, we calculate the same scores as were calculated for the training data and run each classifier on all that data.

Sentence Perplexity using GPT-2

The manual evaluation of ParaCrawl sentence pairs revealed that the Icelandic sentences in ParaCrawl are frequently ungrammatical or have erratic syntax, even though some, and in some cases most or all, of the lexical semantics of the translations are correct. This is likely because many web pages, scraped by the ParaCrawl project, use MT models to generate texts in multiple languages, even though the MT models do not generate fluent results. We try to find these badly formed sentences by training a classifier to recognize fluent and disfluent sentences. The classifier uses a pre-trained GPT-2 model (Radford et al., 2019), trained on the IGC.[4] We selected sentences to train the classifier randomly from WikiMatrix (Schwenk et al., 2021) and ParaCrawl v8, and manually classified them in two groups, *coherent* and *incoherent*. 6,570 sentences were classified as coherent and 3,430 sentences as incoherent.[5] The classifier uses the GPT-2 model to calculate perplexity for the sentences, and chooses potential thresholds as the average between two adjacent perplexity values. It then uses a

[3] `https://universaldependencies.org/treebanks/is_pud/index.html`

[4] The model, trained by Jón Friðrik Daðason, a PhD student at Reykjavik University, is available on Hugging Face: `https://huggingface.co/jonfd/gpt2-igc-is/tree/v1.0`.

[5] Dataset available here: `https://github.com/steinst/filter-align-datasets`

Correct Codes	
C: *Correct translation, any*	*Combined label for CC, CB, CS*
CC: *Correct translation, natural sentence*	
`en`: I caught a big fish yesterday.	`is`: Ég veiddi stóran fisk í gær. *en: I cought a big fish yesterday.*
CB: *Correct translation, boilerplate, partial alignments or grammatical errors*	
`en`: Dust can not get to the engine itself.	`is`: Ryk getur ekki fá til vélarinnar sjálfrar. *en: Dust can not receive to the engine itself.*
`en`: Search for hotels in Liausson	`is`: Leita að hótelum - Liausson *en: Search for hotels - Liausson.*
`en`: I feel something, I'm telling you.	`is`: Ég er að segja þér það. *en: I'm telling you.*
CS: *Correct translation, short*	
`en`: All right.	`is`: Allt í lagi. *en: All right.*
Error Codes	
X: *Incorrect translation, but both correct languages*	
`en`: What a gorgeous image.	`is`: Þetta er glæsilegur árangur hjá þér. *en: This is an impressive accomplishment for you.*
WL: *Source OR target wrong language, but both still linguistic content*	
`en`: Alamin ang mga detalye...	`is`: Nánari upplýsingar... *en: Further information...*
NL: *Not a language: at least one of source and target are not linguistic content*	
`en`: 7.7 / 7.0 knots	`is`: 7.7 / 6.5

Table 4.1: Annotation codes for sentence pairs. Taxonomy developed by Kreutzer et al. (2022), with slight adaptations to the CB class. An English gloss of the Icelandic segments is provided in italics.

maximization function to decide on a threshold that yields the most accurate prediction based on the training set. This approach was selected in collaboration with Jón Friðrik Daðason.

4.2.3 Manual Evaluation

In order to gain some understanding of how good the translations in our parallel corpora are, and to see what kind of data our filters weed out, we manually annotated samples of the data sets compiled by each filtering approach. We found that ParaCrawl contained a lot of boilerplate, as well as a large portion of data most likely translated from English to Icelandic using inadequate MT systems. ParIce, in contrast, commonly includes misaligned segments where a segment in one language contains not only the translation of the segment in the other language, but also extraneous data. In our evaluation, we followed the taxonomy developed by Kreutzer et al. (2022), using three codes for correct segment pairs and three error codes (see Table 4.1). In order to accommodate for the characteristics of the two corpora, we decided to amend one category, CB (*correct translation, but boilerplate or grammatical errors*). We use that category as carried out by Kreutzer et al. (2022), but we also use it for partially aligned sentence pairs and segment pairs where a sentence in one of the languages is grammatically incorrect, while the meaning is still conveyed. Many of the translations in these sentence pairs are probably generated by MT systems.

ParaCrawl shallow filtering					
Filter	Dataset Size	CC (%)	3C (%)	X (%)	3X (%)
0. ParaCrawl v9 Raw	65,373,727	14.40	69.20	8.00	30.80
0b. ParaCrawl v9 Clean	2,967,519	13.60	89.20	8.80	10.80
1.-3. Non-zero / low overlap (accepted)	31,094,385	23.60	94.80	4.40	5.20
1.-3. Non-zero / low overlap (discarded)	34,285,591	1.60	46.80	9.20	53.20
4.-5. Symbol+Language filter (accepted)	26,609,214	25.00	97.20	2.80	2.80
4.-5. Symbol+Language filter (discarded)	4,485,171	11.20	85.60	9.20	14.40
6. Similar pairs (accepted)	4,666,464	12.00	86.80	12.80	13.20
7. Similar segments (accepted)	2,081,354	14.80	95.60	3.60	4.40
ParIce shallow filtering					
0. ParIce 21.10 filtered	1,776,049	73.60	95.20	4.80	4.80

Table 4.2: Sizes and manual evaluation results for the shallow filtering approaches. For each dataset 250 randomly sampled pairs are evaluated. 3C stands for all correct codes: CC, CB and CS. 3X stands for all error codes: X, WL and NL. For comparison, we also evaluate the clean version of the corpus as published by the ParaCrawl project. Note that we evaluated both accepted and discarded pairs for two of the filtering steps.

For the datasets created by applying different shallow filters to ParaCrawl, we annotated 250 randomly selected pairs from each dataset. Two annotators carried out the evaluation, myself and a linguist, fluent in English, Finnur Ágúst Ingimundarson. Sentence pairs for all the different approaches were collected and the order randomized. We worked together to come to a mutual understanding on how to annotate the sentence pairs. Finnur then annotated all the sentences and I checked the annotations to look for errors and to make sure they were as standardized as possible, with similar sentence pairs annotated in the same way. We then went on to randomly sample and annotate 100 pairs from each group of stochastic filtering approaches using the same approach as before, as well as carrying out a more thorough evaluation of the scoring approaches, to discover whether they are good at making a distinction between mutual translations and erroneous or lower-quality segments.

Table 4.2 shows the size of the datasets when shallow filtering has been applied, and the percentage of sentence pairs in different categories. The evaluation indicates that almost 70% of the raw ParaCrawl data are potentially useful data, while over 30% is in the best case useless and possibly detrimental. That data has not been cleaned at all and contains duplicates and many pairs that are very similar. ParaCrawl also distributes a cleaned version of the corpus,[6] containing approximately three million sentence pairs. In that version, over 10% of sentence pairs are still erroneous, and while almost 90% are potentially useful, only 13.6% are evaluated to be good mutual translations. We filter the raw data and show how the accuracy changes with every shallow filtering step. All the filters discard some mutual translations but proportionally a lot more inadequate pairs. While the 3C column indicates the ratio of all pairs in the correct category, some sentences are boilerplate or ungrammatical and may or may not be useful for MT. With our filtering procedures we want to keep as many sentence pairs from the CC category and remove all from the X-categories. When the last two filters are applied, the number of pairs annotated as correct, CC, is lowered from 25% down to 14.8%. This is because sentences that are identical except for numbers or other named entities have been reduced to one example.

[6] `https://web-language-models.s3.us-east-1.amazonaws.com/paracrawl/release9/en-is/en-is.deferred.tmx.gz`

	Laser								**LaBSE**							
	ParaCrawl				**ParIce**				**ParaCrawl**				**ParIce**			
	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X
0.0 => 0.1	10	100	0	0	95	100	0	0	0	7	93	93	1	9	91	91
0.1 => 0.2	9	99	1	1	93	99	1	1	0	5	95	95	4	12	88	88
0.2 => 0.3	8	99	1	1	92	100	0	0	1	6	94	94	11	26	74	74
0.3 => 0.4	16	100	0	0	87	100	0	0	0	7	93	93	14	50	50	50
0.4 => 0.5	16	99	1	1	83	99	1	1	2	13	85	87	24	75	25	25
0.5 => 0.6	20	85	14	15	75	98	2	2	4	42	57	58	46	93	7	7
0.6 => 0.7	15	69	31	31	61	90	10	10	16	71	29	29	64	98	2	2
0.7 => 0.8	10	43	57	57	58	93	7	7	26	94	6	6	82	100	0	0
0.8 => 0.9	13	56	42	44	63	75	25	25	15	98	2	2	89	99	1	1
0.9 => 1.0	27	63	36	37	51	65	36	36	11	99	1	1	99	100	0	0
	NMTScore								**WAScore**							
	ParaCrawl				**ParIce**				**ParaCrawl**				**ParIce**			
	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X
0.0 => 0.1	22	76	24	24	65	92	8	8	1	17	81	83	8	45	55	55
0.1 => 0.2	20	96	4	4	87	100	0	0	12	46	53	54	43	91	9	9
0.2 => 0.3	12	98	2	2	85	100	0	0	28	72	21	28	57	95	5	5
0.3 => 0.4	9	100	0	0	94	100	0	0	27	88	9	12	73	97	3	3
0.4 => 0.5	9	100	0	0	97	100	0	0	39	96	4	4	80	100	0	0
0.5 => 0.6	12	99	1	1	97	100	0	0	33	95	5	5	92	100	0	0
0.6 => 0.7	13	100	0	0	93	100	0	0	27	93	7	7	93	99	1	1
0.7 => 0.8	11	99	0	1	99	100	0	0	10	99	1	1	94	99	1	1
0.8 => 0.9	23	100	0	0	100	100	0	0	7	97	3	3	94	99	1	1
0.9 => 1.0	20	100	0	0	100	100	0	0	5	98	2	2	95	100	0	0
	ChrF translation score – ParaCrawl								**ChrF translation score – ParIce**							
	en–is				**is–en**				**en–is**				**is–en**			
	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X
0.0 => 0.1	0	11	89	89	4	28	72	72	13	29	69	71	6	14	85	86
0.1 => 0.2	1	6	94	94	1	13	87	87	18	48	52	52	11	39	60	61
0.2 => 0.3	9	38	62	62	8	26	73	74	40	83	16	17	41	66	34	34
0.3 => 0.4	25	84	16	16	22	67	33	33	56	89	10	11	48	85	15	15
0.4 => 0.5	34	96	4	4	36	91	9	9	70	100	0	0	74	96	3	4
0.5 => 0.6	24	97	3	3	30	95	5	5	85	99	0	1	79	97	3	3
0.6 => 0.7	21	96	4	4	25	99	1	1	80	99	1	1	87	100	0	0
0.7 => 0.8	9	97	3	3	9	95	4	5	89	100	0	0	79	99	1	1
0.8 => 0.9	8	100	0	0	11	100	0	0	93	100	0	0	93	100	0	0
0.9 => 1.0	19	100	0	0	8	99	0	1	91	97	3	3	94	99	0	1
	Bag-of-words translation score – ParaCrawl								**Bag-of-words translation score – ParIce**							
	en–is				**is–en**				**en–is**				**is–en**			
	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X	CC	3C	X	3X
0.0 => 0.1	7	26	73	74	6	20	79	80	25	46	54	54	24	45	53	55
0.1 => 0.2	10	40	60	60	7	22	77	78	49	78	22	22	19	49	57	57
0.2 => 0.3	30	70	29	30	10	44	56	56	49	86	13	14	25	69	31	31
0.3 => 0.4	36	86	14	14	27	72	27	28	74	97	3	3	49	85	15	15
0.4 => 0.5	37	97	3	3	33	80	20	20	65	97	3	3	61	90	9	10
0.5 => 0.6	19	97	3	3	29	91	8	9	78	97	3	3	73	96	4	4
0.6 => 0.7	16	98	2	2	19	99	1	1	75	98	0	2	78	97	2	3
0.7 => 0.8	6	98	2	2	11	100	0	0	85	99	0	1	85	100	0	0
0.8 => 0.9	3	95	5	5	7	97	3	3	84	99	0	1	91	99	1	1
0.9 => 1.0	8	98	2	2	9	96	3	4	81	99	0	1	90	99	0	1

Table 4.3: Results of the manual evaluation of samples of 100 randomly selected sentence pairs from each of ten bands for the scoring mechanisms used.

For the ParIce corpus, we only evaluated a dataset that had been filtered using the shallow filters described in 4.2.2, and did not investigate the changes at each stage. This is because the ParIce corpus is smaller and the data in the corpus all comes from known document sources and should not contain as much noisy data as ParaCrawl. We found that about 5% of sentence pairs in the corpus were erroneous, a number largely in line with the original ParIce paper, where the evaluation indicated that 3.5% of the alignments were bad, but we also found that only about three out of every four sentence pairs were mutual translations, with about 20% being accepted as correct but not annotated as CC, indicating they were imperfect in some way, usually due to misalignments.

Due to the large differences in the results of the evaluation of the two corpora, we wanted to investigate further whether different filtering approaches might suit the different corpora. However, first we wanted to investigate how good our chosen scoring mechanisms were by evaluating sentence pairs for each score, selecting samples for ten bands for each scoring approach.

As detailed in Section 4.2.2, we used a variety of scoring mechanisms, based on different approaches. In order to see if they are effective at identifying good translations from inferior ones and erroneous pairs, Finnur and myself manually evaluated 1000 sentence pairs for each scoring mechanism, divided into 10 scoring bands with 100 pairs in each. Evaluation results are shown in Table 4.3. The evaluation indicates that all the scoring methods have some merit and could probably be useful to a classifier. On their own, the results usually differ depending on the parallel corpora used, with the accuracy of the same scoring mechanism varying for different corpora. For example, for more than 90% of sentence pairs in a scoring band to be acceptable (3C), the LaBSE score has to be more than 0.7 for ParaCrawl, but 0.5 for ParIce. This may be because a large part of the ParIce corpus comes from two domain specific subcorpora, EEA regulations and directives, and texts from the European Medicines Agency document portal, and these domains may not be well represented in the LaBSE training data. As LaBSE is not as confident with sentences from these domains, they may score slightly lower even though they are accepted as correct in manual evaluation. The distribution of the scores are also quite different between scoring approaches, which can effect their usefulness. While NMTScore seems to be very accurate when looking at the bands in the table, 83% of the ParaCrawl sentences have a score of less than 0.3, and 25% of the ParIce sentences have a score of less than 0.1, indicating that even though the results seem very good, using only this scoring method may not be enough for accurate filtering. It should also be noted that most of the approaches do not seem to be very good at discerning finer nuances such as whether a sentence pair contains only mutual translations or if there is additional content in at least one of the sentences. The ratio of CC thus usually does not change as consistently with rising scores as the 3C or 3X ratio. This may indicate that we need other approaches to identify these sentence pairs and filter them out if we find that some of the sentence pairs classified as CB are detrimental to MT training.

Our next step is thus to train classifiers based on these scores and described in Section 4.2.2, in order to find if we can more effectively filter the sentence pairs using multiple scoring mechanisms at the same time. We also use four different Bicleaner models, and set the cutoff score at two different levels for each model, the default 0.5 threshold, and a higher threshold of 0.67 to try to discover whether detrimental sentence pairs can still be found at such a high level.

As is evident in Table 4.4, the filtering mechanisms are quite adept at removing the most erroneous sentence pairs. We can see that for both corpora in our study, all but two filters return over 90% accepted sentence pairs, and a low rate of erroneous data, and for ParIce in particular almost all erroneous data is removed with the 3X category (any erroneous data)

Filter	ParaCrawl Filters									
	accepted (%)					rejected (%)				
	No. pairs	CC	3C	X	3X	No. pairs	CC	3C	X	3X
GPT-2	1,218,256	15	93	7	7	863.098	5	91	8	9
Logistic Regression	1,940,385	38	85	4	15	140,969	18	37	61	63
Random Forest	1,981,405	7	98	0	2	99,949	2	22	77	78
Support Vector Machine	1,991,924	12	98	2	2	89,430	0	22	78	78
Bicleaner baseline (0.50)	1,973,885	22	96	4	4	107,469	10	41	58	59
Bicleaner baseline (0.67)	1,705,042	15	98	2	2	376,312	20	80	20	20
Bicleaner retrained (0.50)	1,898,209	25	97	3	3	183,145	27	75	25	25
Bicleaner retrained (0.67)	1,615,913	20	98	2	2	465,441	24	81	18	19
Bicleaner lemmatized (0.50)	1,850,884	18	88	8	12	230,470	14	66	28	34
Bicleaner lemmatized (0.67)	1,512,437	30	93	5	7	568,917	21	70	29	30
Bicleaner AI (0.50)	1,235,771	**33**	**99**	1	1	845,583	6	85	13	15
Bicleaner AI (0.67)	1,096,288	25	97	3	3	985,066	8	92	8	8

Filter	ParIce filters									
	accepted (%)					rejected (%)				
	No. pairs	CC	3C	X	3X	No. pairs	CC	3C	X	3X
GPT-2	1,444,956	81	96	4	4	331,093	68	91	9	9
Logistic Regression	1,560,346	85	100	0	0	215,703	49	77	23	23
Random Forest	1,667,847	86	99	1	1	108,202	20	51	49	49
Support Vector Machine	1,646,183	91	100	0	0	129,866	28	58	42	42
Bicleaner baseline (0.50)	1,546,216	85	99	1	1	229,833	35	79	21	21
Bicleaner baseline (0.67)	1,242,258	86	100	0	0	533,791	48	86	14	14
Bicleaner retrained (0.50)	1,499,610	85	99	1	1	276,439	42	90	10	10
Bicleaner retrained (0.67)	1,244,412	94	100	0	0	531,637	55	95	5	5
Bicleaner lemmatized (0.50)	1,463,780	89	100	0	0	312,267	50	90	10	10
Bicleaner lemmatized (0.67)	1,117,814	88	100	0	0	604,235	69	98	2	2
Bicleaner AI (0.50)	1,262,313	**95**	**100**	0	0	513,736	60	86	13	14
Bicleaner AI (0.67)	1,152,319	91	100	0	0	623,730	77	95	5	5

Table 4.4: Results of the manual evaluation of samples of 100 randomly selected sentence pairs from datasets generated by different filtering approaches. We both evaluate sentence pairs accepted by each filtering approach, and rejected by it.

at 0% in the evaluated data for 8 out of 12 filtering approaches. However, as it is important to keep as many of the correct sentence pairs as possible, the most useful approaches may be the ones that remove the fewest mutual translations. We see that while the model having the highest proportion of CC, mutual translations, is Bicleaner AI, it has the drawback of filtering out the highest proportion of sentences compared to almost all other approaches. With almost half of the ParaCrawl data rejected when the threshold score is set to 0.67, and only 1,096,288 sentence pairs left out of 2,081,354 sentence pairs, 92% of the rejected sentence pairs are annotated in one of the C categories. In order to investigate further what is best for MT training, we next train multiple models, using all the different data sets we have compiled, in order to see how the translations generated by these models compare to the results of our manual evaluation.

4.2.4 Automatic Evaluation

In order to evaluate the effect of different filtering approaches for training data on MT output, we train different MT models for each of the compiled datasets and evaluate them using BLEU (Papineni et al., 2002). We use fairseq to train Transformer$_{\mathrm{BASE}}$ models as described in Section 3.5.4. We use the development and test sets provided for English–Icelandic news translation task at WMT 2021 (Akhbardeh et al., 2021). The development sets are used for validation, with validation loss being the criteria for early stopping if it does not improve for 10 checkpoints. The test sets are used for evaluation, using SacreBLEU.[7]

[7] SacreBLEU Signature: BLEU+numrefs.1+case.mixed+tok.13a+smooth.exp +version.2.2.0

ParaCrawl training experiments					
Dataset	no. pairs	en→is	time	is→en	time
Baseline: ParaCrawl v9 clean	2,967,519	20.2	23h3m	30.7	11h14m
Shallow filter 3 - Similar pairs	4,666,464	19.1	18h9m	30.4	29h56m
Shallow filter 4 - Similar segments	2,081,354	20.0	13h3m	31.9	15h57m
IS-perplexity (GPT-2)	1,218,256	**21.1**	5h50m	**33.0**	14h11m
SVM	1,991,924	19.6	13h41	32.4	15h56m
Logistic Regression	1,940,385	20.1	11h48	32.1	12h01m
Random Forest	1,981,405	19.5	6h37m	31.8	15h32m
Bicleaner 1.5 (0.50)	1,973,885	19.5	11h25m	32.2	15h33m
Bicleaner 1.5 (0.67)	1,705,042	19.3	8h29m	31.4	8h53m
Bicleaner retrained (0.50)	1,898,209	18.9	8h17m	31.9	15h41m
Bicleaner retrained (0.67)	1,615,913	19.5	7h36m	30.5	12h59m
Bicleaner lemmatized (0.50)	1,850,884	19.6	10h29m	31.6	17h19m
Bicleaner lemmatized (0.67)	1,512,437	19.3	6h27m	30.9	8h32m
Bicleaner AI (0.50)	1,235,771	20.5	8h26m	31.7	7h15m
Bicleaner AI (0.67)	1,096,288	*21.0*	4h50m	30.8	3h45m
ParIce training experiments					
Dataset	no. pairs	en→is	time	is→en	time
Baseline: ParIce shallow filter	1,776,049	*19.7*	23h29m	25.5	14h31m
IS-perplexity (GPT-2)	1,444,956	18.5	22h33m	24.7	10h18m
SVM	1,646,183	**19.8**	17h38m	*26.0*	13h04m
Logistic Regression	1,560,346	19.2	16h51m	*26.1*	13h30m
Random Forest	1,667,847	18.6	20h07m	25.2	12h22m
Bicleaner 1.5 (0.50)	1,546,216	*19.5*	21h52m	**26.2**	12h5m
Bicleaner 1.5 (0.67)	1,242,258	*19.5*	12h06m	25.6	9h01m
Bicleaner retrained (0.50)	1,499,610	*19.7*	7h13m	25.6	12h22m
Bicleaner retrained (0.67)	1,244,412	**19.8**	10h16m	25.5	6h13m
Bicleaner lemmatized (0.50)	1,463,780	**19.8**	15h12m	25.9	11h56m
Bicleaner lemmatized (0.67)	1,171,814	**19.8**	7h29m	25.6	8h56m
Bicleaner AI (0.50)	1,262,313	19.1	7h07m	*26.1*	7h44m
Bicleaner AI (0.67)	1,152,319	18.9	7h11m	25.1	7h28m

Table 4.5: BLEU scores and training time for different filtering approaches. Scores in bold are the highest for the dataset and translation direction. Scores in italics are lower but not significantly lower than the highest ones ($p > 0.05$).

Our translation models commonly generate translations using the same quotation marks as in the source sentences. Following Koszowski et al. (2021) we apply regular expressions to fix quoting, making sure Icelandic quotation marks („ “) are used in the Icelandic translations and English quotation marks (“ ”) in the English translations.

We train baseline models using only the datasets compiled by the shallow filters, as well as using the readily available cleaned ParaCrawl dataset. We then train models using the filtering approaches previously evaluated manually, calculate BLEU scores and record the time it took to train each model until convergence. Table 4.5 shows the results. The scores for the separate filtering approaches do not always differ significantly from the baselines. The baseline we set for ParIce, only applying the shallow filters, is very close to the highest BLEU score for en→is. On the other hand, in most cases filtering the data allows the models to converge faster, reducing the training time to reach scores in line with or above the baseline scores. We know from our manual evaluation that most of these training sets contain some erroneous pairs, and in order to try to reduce the number of these we select the dataset resulting in the highest BLEU score out of the datasets compiled by a Bicleaner model and the best resulting dataset compiled by a classifier. We do an ablation study to investigate whether by combining these filters, as well as the filter looking at perplexity in Icelandic sentences, can help us create a better training set.

When we combine multiple filters we only retain the sentence pairs all the filters accept. As well as combining the different filtering methods, we try adding all sentence pairs being scored high enough to have 95% possibility or more by all scoring metrics of being mutual translations, according to the manual evaluations shown in Table 4.3. We then remove all

ParaCrawl en→is filters			
Dataset	no. pairs	BLEU	time
Bicleaner AI (0.67) + LogReg	1,071,802	20.4	3h51m
Bicleaner AI (0.67) + GPT-2	776,984	**21.5**	4h17m
Bicleaner AI (0.67) + LogReg + GPT-2	756,503	20.7	3h40m
Bicleaner AI (0.67) + LogReg + GPT-2 + Add95 - ScoreMaj	851,059	*21.2*	4h10m
ParaCrawl is→en filters			
Dataset	no. pairs	BLEU	time
Bicleaner 1.5 (0.50) + SVM	1,930,998	32.3	20h24m
Bicleaner 1.5 (0.50) + GPT-2	1,147,961	31.9	9h02m
Bicleaner 1.5 (0.50) + SVM + GPT-2	1,119,400	32.1	7h32m
Bicleaner 1.5 (0.50) + SVM + GPT-2 + Add95 - ScoreMaj	1,920,186	31.6	7h05m
ParIce en→is filters			
Dataset	no. pairs	BLEU	time
Bicleaner Lemmatized (0.50) + SVM	1,405,446	**20.2**	17h59m
Bicleaner Lemmatized (0.50) + GPT-2	1,205,070	19.6	14h04m
Bicleaner Lemmatized (0.50) + SVM + GPT-2	1,161,337	18.9	13h24m
Bicleaner Lemmatized (0.50) + SVM + GPT-2 + Add95 - ScoreMaj	1,417,565	19.5	14h04m
ParIce is→en filters			
Dataset	no. pairs	BLEU	time
Bicleaner 1.5 (0.50) + LogReg	1,430,015	*26.1*	13h22m
Bicleaner 1.5 (0.50) + GPT-2	1,269,808	*25.7*	9h30m
Bicleaner 1.5 (0.50) + LogReg + GPT-2	1,179,158	*25.7*	10h46m
Bicleaner 1.5 (0.50) + LogReg + GPT-2 + Add95 - ScoreMaj	1,544,980	**26.2**	10h17m
Best datasets from both corpora combined.			
Dataset	no. pairs	BLEU	time
is→en: ParaCrawl – GPT-2 + ParIce Bicleaner 1.5 (0.50)	2,764,472	***33.2***	15h55m
en→is: ParaCrawl – Bicleaner AI (0.67) + GPT-2 + ParIce – Bicleaner Lemmatized (0.50) + SVM	2,182,430	***22.6***	18h57m

Table 4.6: BLEU scores and training time for combinations of different filtering approaches. The *Add95* and *ScoreMaj* filters, described in Section 4.2.4, are also added to the models combining three filters. While datasets compiled with combined filters were used to train MT systems delivering the highest BLEU scores for the English→Icelandic translation direction, for Icelandic→English the highest scoring systems were trained on data compiled with only one filter. Scores in bold are the highest scores for the dataset and translation direction they represent. Scores in italics are lower but not significantly lower ($p > 0.05$). Scores in bold and italics are the highest scores obtained for the translation direction.

sentence pairs where the scores indicate a more than 50% chance of the sentence pair being erroneous. These datasets are indicated with *Add95 - ScoreMaj* in Table 4.6, which shows the results of the different combinations. For the English→Icelandic translation direction combined filters compile datasets obtaining higher scores for both corpora. On the other hand, the BLEU scores never exceed the best standalone filters for the Icelandic→English translation direction. We hypothesize that English as a target language may be more robust to noise in the training data than Icelandic as a target language, cancelling out beneficial effects of training on somewhat cleaner data. We discuss these ideas further in Chapter 8.

Finally, we combine the highest-scoring datasets for ParaCrawl and ParIce to create a final dataset. The models trained from these datasets obtain the highest BLEU scores, 22.6 for English→Icelandic and 33.2 for Icelandic→English. Comparing these scores to the results of the WMT21 shared translation task for the same language pair and directions, we see that they are competitive with the Transformer$_{\text{BIG}}$ models (Vaswani et al., 2017) submitted by Koszowski et al. (2021) (en→is: 22.7; is→en: 33.3), and with the mBART-25 models submitted by Símonarson et al. (2021) (en→is: 24.3; is→en: 33.5), which are trained on more data, using more computational resources and for a longer time than we do. We will come back to these two models and compare them to our final models in Chapter 7.

4.3 Conclusions

In this chapter, we addressed our first two research questions: RQ1: How we can filter parallel corpora to minimize noise, while still losing little or no useful data from the original texts, and RQ2: To what degree we should filter parallel corpora without regard to the dataset or translation direction in question. Our results indicate that different filtering approaches suit different datasets and translation directions, even though we are working within the same language pair. For the language pair we focus on, English–Icelandic, we find that single filters work well for is→en, while a combination of filters work better for en→is, for both the datasets we work with. As the morphology is more complex in Icelandic, we speculate that models translating into a more morphological complex language may be more sensitive to ungrammatical and noisy target language data. A more thorough filtering thus works better, even though a higher proportion of beneficial sentence pairs are lost in the process. When translating into English the models may be more forgiving, as there are fewer word forms proportionally to lemmas, and so fewer sentence pairs need to be filtered out. While we see some general tendencies in our experiments, they do not show us what data exactly are detrimental and which are beneficial.

On another level, our manual evaluation shows that the scores generated by the automatic scoring systems have different interpretations depending on the dataset. If the scores are used for filtering or mining parallel data, the optimal score for the dataset should thus be found to result in a dataset that produces the best MT model. Feng et al. (2022) suggest a threshold of 0.6 for LaBSE when mining parallel text from CommonCrawl, stating that the threshold was selected by manually inspecting sampled data, but they do not specify the language pair used when inspecting the data. In order for the scoring mechanism to be most effective, the user should thus inspect the results for their dataset before setting a threshold. While all the scoring mechanisms seem to be useful for filtering out useless or detrimental data, and likewise to find possibly useful data (that is sentence pairs that are at least partial translations), none of the methods are very good at finding mutual translations, labelled as CC. In Chapter 6 we try to close in on that goal by mining for mutual translations on the sub-sentential level.

We trained two Bicleaner models for our experiments. They worked reasonably well and for filtering ParIce for the en→is translation direction, the lemmatized Bicleaner model gave the best results. These models could perhaps be made even better. Bicleaner uses n-gram models and we only used our parallel corpora to train these. By using If we would use larger corpora the n-gram models would likely give us more accurate scores thus resulting in a more accurate model.

In the next chapter, we will compare different approaches to sentence alignment, searching for the most effective way to align two documents on the sentence level.

Chapter 5

Sentence Alignment

In the previous chapter, we looked into the effectiveness of various filtering approaches and how different filters may suit different translation directions. In this chapter, we will turn our attention to sentence alignment and examine whether better sentence alignment approaches are likely to yield improved results on downstream MT tasks. Sentence alignment is the task of finding as many matching sentences as possible from two documents, one of which is the translation of the other. It can be considered to be a path-finding problem, with a list of sentences in one language to be the x-axis in a two-dimensional graph and the sentences in the other language to be the y-axis. Each potential sentence pair is a node in the graph and the objective of the sentence alignment algorithm is to find the best path through the graph. The path is most often continuous, with gaps when either one of the documents has sentences that do not have corresponding sentences in the other. The alignments can also be non-monotonous, with sentences crossing so the order of sentences differs between languages. This problem can usually be solved by chunking multiple sentences. We can thus describe automatic sentence alignment in terms of two different problems:

- Scoring the sentence pairs to give an indication as to how likely they are to be a translation of one another.
- Comparing two documents in different languages and returning the best set of aligned sentence pairs via an alignment algorithm.

In this chapter, we describe our own sentence alignment system, *SentAlign*, and evaluate it. Additionally, we describe five other available sentence aligners and how their scoring and alignment algorithm work. We use all six systems in our experiments, as well as experimenting with using a combination of systems do decide on alignments. We evaluate in three different ways. First, we compare how the alignment approaches fare on two different evaluation sets. Second, we manually evaluate samples of the aligned sentence pairs. Third, we evaluate the approaches in downstream NMT tasks in terms BLEU score.

We use the sentence alignment evaluation sets distributed with Bleualign[1] (see Section 5.2.4) as well as new English–Icelandic evaluation sets we compiled.[2] The parallel documents used are the EEA documents,[3] mostly regulations and directives, used to compile the EEA part of the ParIce corpus (see Section 3.5).

[1] https://github.com/rsennrich/Bleualign

[2] Available at https://repository.clarin.is/repository/xmlui/handle/20.500.12537/150

[3] Documents downloadable from: https://github.com/steinst/filter-align-datasets

The aim of this chapter is to gain insights into how important sentence alignment is in the process of compiling a parallel corpus and answer **RQ3: Is sentence alignment accuracy important for the results of a downstream MT task, or is effective filtering of the training data enough?** In answering that we will compare different approaches, also asking which methods we can use to help us find the best alignment approaches for a given task.

The remainder of the chapter is organized as follows. In Section 5.1, we describe previous and related work. Section 5.2 describes all the approaches we use for sentence alignment, including five alignment methods previously published by others, as well as our own system. In Section 5.3, we describe our experiments and results, and, in Section 5.4, we conclude the discussion on sentence alignment.

5.1 Related Work

Recently, the state-of-the-art in sentence alignment has been improved for some language pairs by employing sentence representations obtained from pre-trained multilingual language models (MLMs). Vecalign (Thompson and Koehn, 2019), which we describe in more detail in Section 5.2.5, uses LASER (Artetxe and Schwenk, 2019b) to score sentence pair candidates. Feng et al. (2022) compare LaBSE (Feng et al., 2022) and LASER for sentence alignment and Rajitha et al. (2020) compare LASER and XLM-R (Conneau et al., 2020) for the related task of document alignment. Fernando et al. (2023) compare LASER, LaBSE and XLM-R for both document and sentence alignment, incorporating bilingual lexicons to generate weights between sentences in a candidate pair, applying them to the MLM-based similarity scores to calculate a final score. In their evaluation, LaBSE scored highest on an evaluation dataset, while the difference was largely insignificant when evaluated on BLEU in a downstream NMT task. While these large MLMs can give good results for language pairs represented in the training data for these models, it has been noted that the results are not as good for languages that are not well represented in the models. In their review of using LASER and LaBSE for parallel corpora mining from bilingual texts, Chimoto and Bassett (2022) found that there is a stark difference in alignment quality depending on whether the language is represented in the MLM training data or not. They do note, however, that this problem can be somewhat alleviated by fine-tuning the language models on texts in the unrepresented language.

Previously, various different methods have been developed for sentence alignment. The first approaches to automatic sentence alignment were length-based. Gale and Church (1991) found that “the correlation between the length of a paragraph in characters and the length of its translation was extremely high”. Motivated by that, they describe a method for aligning sentences based on a simple statistical model of character lengths. Brown et al. (1991) also describe a length-based method, but use tokens instead of characters. When finding the optimal alignments, a sentence in one language is compared to all sentences in the other language. As most parallel documents have a similar amount of lines in each language, this is a quadratic time complexity algorithm, $O(n^2)$, where n is the approximate number of sentences in each language. In order to reduce the search space, dividing the corpora to be aligned into ‘beads’ (windows where it is most likely to find the correct alignments) is a common approach. Brown et al. (1991) assume prior alignment of paragraphs and use the paragraph boundaries as anchors, while Gale and Church (1991) rely on previously aligned sentences as anchor points and use these to divide the text into smaller chunks. Martinez et al. (1998) used markup in the texts for anchoring. That can be very useful when available, but is unfortunately not common in bilingual texts.

Lexical information has been shown to improve on pure length-based methods (Chen, 1993; Wu, 1994). The lexical features can be acquired by various means. Simard et al. (1992) suggest using cognates to help with aligning bilingual corpora. They informally define cognates for these purposes, as pairs of tokens in different language which share phonological or orthographic and semantic properties, usually having the same origin. Based on that they compute the "cognateness", comparing the number of matching tokens in a string to the length of the string. They observe that cognates are more likely to occur in mutual translations than in random pairs, although unrelated pairs of sentences frequently share cognates, especially if they appear in the same context. In their experiment, they use bitexts that can be correctly aligned, and show that, for aligning such texts, cognates can be useful in combination with other methods when alignment is performed in two passes, even though cognates are not very reliable for alignment on their own. Lamraoui and Langlais (2013) come to the same conclusion using a slightly different two-pass method, and Simard (1999) shows that accuracy can be improved when more than two languages are being aligned at once. Using bilingual lexicons is also common, though they are harder to obtain than cognates. Some systems infer a bilingual lexicon from the texts being aligned (Chen, 1993; Kay and Röscheisen, 1993; Moore, 2002), while others require an external lexicon to be provided (Wu, 1994; Li et al., 2010). Haruno and Yamazaki (1996) show that combining an induced lexicon with an external dictionary yields better results. Bilingual lexicon induction was discussed in more detail in Section 3.6.

Papageorgiou et al. (1994) use part-of-speech (PoS), commonly preserved in translation, by computing the optimum alignment based on the PoS tags. Tschorn and Lüdeling (2003) use a morphological analyzer to improve a dictionary-based distance measure, and Ma (2006) increases the robustness of a lexicon-based aligner by assigning greater weights to less frequent translated words, while using a dynamic programming algorithm that allows from 0–1 up to 4–1 alignments. Braune and Fraser (2010) use a multi-pass procedure using length-based statistics and a modified version of the model by Moore (2002).

As SMT improved in the early 2000s, MT-based alignment methods became viable. Following Adafre and de Rijke (2006), who use MT to mine parallel sentences from Wikipedia pages, Sennrich and Volk (2010) use MT and BLEU as a similarity score for their alignment system, Bleualign, described in more detail in Section 5.2.4. These methods are still being applied in various scenarios today.

In the WMT 2020 shared task on parallel corpus alignment and filtering (Koehn et al., 2020), three teams described their sentence alignment approaches. Xu et al. (2020) build their alignment approach upon statistical lexicon translation scores, based on word alignments obtained by fast_align (Dyer et al., 2013). Lu et al. (2020) used Champollion Ma (2006), a lexicon-based sentence aligner, and word alignment to extract the bilingual lexicon from a parallel corpus. Lo and Joanis (2020) use an iterative statistical method, first aligning paragraphs and then sentences within the paragraphs, employing ssal, a reimplementation and extension of Moore (2002), which uses the IBM-HMM model (Och and Ney, 2003).

Hasan et al. (2020) use aligner ensembles, taking a union of sentence pairs and thus collecting a large part of the aligned sentences. Zhang (2022) uses a divide and conquer approach, using 1–1 alignments, which are surrounded by other 1–1 alignments, as anchors, and then does a second pass.

5.2 Our Approaches to Sentence Alignment

In our experiments, we employ six different approaches to aligning parallel documents at the sentence level. We describe the scoring mechanism and the alignment selection algorithm in five previously available tools and try to identify their weak and strong points. Furthermore, we describe in detail our own tool, SentAlign.

5.2.1 Gale–Church Algorithm

Gale and Church (1991) note that DP is often used to align two sequences of symbols in a variety of settings. It could thus be expected that such a matching technique would be useful for aligning sequences of text in two languages. Details of the techniques differ from one application to the other, but all use a distance measure to compare individual elements within the sequences and a DP algorithm to minimize the total distance between aligned elements. The algorithm by Gale and Church (1991) is a DP method that compares the length (in characters) of two sentences to be aligned. With limited data and computing resources, sentence length is an obvious choice and had been used previously in work described in Brown et al. (1990) and Brown et al. (1991) to extract sentence pairs from parallel documents using statistical approaches. While the previous work uses words for length-based measures, Gale and Church use characters, arguing that there are more of them in the texts and thus there is less variation in length and so less uncertainty, which produces better sentence alignments.

Scoring
The distance measure in the Gale–Church algorithm is based on the assumption that longer sentences in one language tend to be translated into longer sentences in the other language, and shorter sentences tend to be translated into shorter sentences. They assert that each character in one language, L_1, gives rise to a random number of characters in the other language, L_2, which are independent and identically distributed with a normal distribution. The mean of these variables, c, is the expected number of characters in L_2 per character in L_1 and the variance, s^2, is the variance in the number of characters in L_2 per character in L_1. These parameters are determined empirically from a small corpus of English–German–French data, 15 economic reports issued by the Union Bank of Switzerland, containing 725 English sentences and corresponding sentences in the other languages. By observing two language pairs from this dataset, the authors assume language–independent values, which they expect to be useful for most pairs of European languages. Furthermore, they count the frequency of 1–1, 1–0, 0–1, 2–1, 1–2 and 2–2 sentence alignments in the same dataset and give the different alignment types probabilities based on the empirical evidence. These probabilities, as well as the original values for c and s^2, are the default settings in implementations of the Gale–Church algorithm used to calculate alignment cost for different types of alignments, along with a function of source and target sentence lengths. For further details, we refer the interested reader to the original paper.

Alignment Algorithm
The Gale–Church alignment algorithm finds the minimum distance between source sentences, s_1, ..., s_i, and their translations, t_1, ..., t_j, in a corpus containing i sentences in the source language and j sentences in the target language. It does so by recursively calculating the minimum cost for each pair of sentences, allowing for six types of sentence alignments: substitution (1–1), deleting a source sentence (1–0), inserting a target sentence (0–1), contracting two source sentences to one target sentence (2–1), expanding one source sentence

to two target sentences (1–2), and merging two source sentences to match with two merged target sentences (2–2). In the original implementation, the algorithm does not allow merging more than two sentences in one alignment.

After calculating the minimum cost for a pair, the algorithm adds it to the previously calculated costs to reach the pair, finding the least costly path after comparing the cost for all alignment types. This is repeated until the final sentence pair, (s_i, t_j), is reached and the least costly alignment path found.

The Gale–Church algorithm is simple and runs fast for relatively short parallel texts. For such a simple approach, using no semantic information to help with alignment, it is surprisingly accurate, as the results in Section 5.3 show. A large drawback is, however, that the algorithm calculates costs for all possible sentence pairs without having any inbuilt mechanism for dividing the input files into smaller beads or dealing with large files in any other way. As it is of $O(n^2)$ time complexity, it will slow down as the input files get very large and finally it will not be able to process them. For very large documents, Gale–Church needs some hard delimiters to split them up, as it can not deal with them as single units.

The language-independent values are derived from a small manually annotated dataset of two language pairs. Alignments of more than two merged sentences in each language are not considered and no probabilities given for such alignments as their manual annotation does not contain any alignments larger than 2–2. Furthermore, almost 90% of the alignments are 1–1. The assumptions derived from their dataset may fall short for some language pairs and some sets of parallel texts. This can be amended by evaluating the language pair and parallel texts to be aligned to recalculate the parameter values.

Experimental Settings

For our experiments, we reimplemented the Gale–Church algorithm in Python 3, with minor adjustments to the original implementation.[4] As working with very large files tends to yield high costs for very unlikely alignments, we add a cutoff score in order to reduce the number of calculations. When a cost to reach a node exceeds the cutoff score, no paths will be calculated from that node. We apply the cutoff mechanism when the number of sentences in either language exceeds 1,000. A visual inspection of the results indicates that applying the cutoff score does not seem to have an effect on the final alignment outcome in the vast majority of cases when aligning very large files. Our implementation allows the user to set the usually language-independent values of the distance measure in the input parameters. It is accompanied with a script for calculating these measures from an aligned corpus.

While our implementation is accompanied with tools to define language-dependent values for the scoring mechanism, we use the language-independent values given by Gale and Church in order for our results to be better comparable to others.

5.2.2 Hunalign

Varga et al. (2005) argue that the choice of appropriate language technology for a given task is greatly impacted by the availability of digital resources. They define languages spoken by less than 100 million and more than 500 thousand speakers as *medium density* languages, for which some useful data exists but not enough for many tasks. They describe a hybrid sentence alignment algorithm combining dictionary- and length-based methods for sentence alignment to be used for rapidly building parallel resources for medium density languages. While the language technology landscape has changed significantly since their paper was

[4] Our implementation is available at `https://github.com/steinst/galechurch`.

published, it has been argued that progress in NLP has been “restricted to a minuscule subset of the world’s ≈6,500 languages” (Blasi et al., 2022) and the work of the European Language Equality project shows that even the majority of the EU languages are at a risk of digital extinction (Gaspari et al., 2022). This hybrid approach may thus still be a valid choice for some languages where little parallel data and/or digital dictionaries are available.

Scoring

The similarity score has two main components, token-based and length-based. The token-based component searches for shared words in the two sentences. It starts by producing a translation of the source text by automatically finding translations for each source word token, looking for the highest frequency token in the target corpus. This pseudo-translation is then compared against the actual target text on a sentence-by-sentence basis to identify shared words in the sentences. The length-based component is based on the ratio of longer to shorter sentences with the relative weight set to maximize precision on the Hungarian–English training corpus, which the authors assume is a sensible choice for other languages as well. Finally, paragraph boundaries are treated as sentences with special scoring, with the similarity of two paragraph boundaries being a high constant and the similarity of a paragraph boundary to a real sentence being minus infinity, to find paragraph boundaries to pair up.

Alignment algorithm

The similarity score described above is calculated for every sentence pair around the diagonal of the alignment matrix, with at least a 500-sentence neighbourhood calculated and up to 10% of the number of sentences in the longer text, if it is longer than 5,000 sentences. The authors find this to be high enough to produce good recall on their evaluation dataset. They justify this approach by the assumption that the beginning and end of the texts are aligned and that the sentence ratio in the two texts represents the average one-to-many assignment ratio of alignment segments, expecting no significant deviation from that. Scores are also assigned to deleting/inserting and merging sentences. The score of deleting/inserting is calculated using the training corpus and the score of merging by summing up the minimum of the token-based scores for both sentences and the length-based score of the concatenation of the two sentences.

Once a similarity matrix has been obtained, an optimal path is selected. Initially, the algorithm does not take into account the possibility of more than two sentences matching one sentence, but, after an optimal path is found, a postprocessing step iteratively merges a neighbouring pair of 1–n ($n > 1$) and 0–1 segments wherever the resulting new segment has a better character-length ratio than the starting one. With this method, any 1–n alignments can be found.

Adding an external dictionary

While the base algorithm can give meaningful results, an external dictionary can be utilized by the system for more accurate lexical scoring. As dictionaries usually only show lemmas and not all the different possible word forms, the authors suggest stemming the texts to be aligned as well as the external bilingual dictionary. In our case we lemmatize the texts.

Experimental Settings

We perform two experiments with Hunalign: running the tool with and without an external dictionary. When we use a dictionary, we lemmatize both the Icelandic and English texts using ABLTagger first for PoS tagging and then Nefnir to lemmatize the Icelandic texts and spaCy for the English texts. We then provide the system with the dictionary we created and described in Section 3.6.

5.2.3 Gargantua

Gargantua (Braune and Fraser, 2010) is an unsupervised and language-independent sentence alignment tool which uses a two-step clustering approach to sentence alignment. It aims to find 1–n and n–1 alignments where $n \geq 0$, but does not search for many-to-many alignments. The authors argue that while the accuracy for unsupervised and language-independent approaches like Brown et al. (1991) and Gale and Church (1991) decreases drastically when aligning texts containing deletions or free translations, their approach augments a sentence length-based model with lexical statistics, making more informed high-quality alignments.

Scoring
To score the sentences, Gargantua uses sentence length-based statistics considering relative lengths in comparison to the mean length of source and target sentences, and translation likelihoods of each target word with all source words, according to IBM Model-1 (Brown et al., 1990).

Alignment
Gargantua uses a multi-pass approach. The first step looks for optimal alignments through the alignment matrix consisting only of 0–1, 1–0 and 1–1 correspondences. An approximate alignment is computed using the sentence length and the resulting 1–1 alignments then selected for creating translation tables using IBM Model-1.

In a second step, the previously acquired alignments are merged into clusters containing up to R sentences on either the source or target size, where R is an upper bound to the number of allowed sentences. If a 1–1 alignment is found next to a 0–1 or 1–0 alignment, that is a candidate for composing a cluster, and if the merge produces a better score it is accepted. As this step only clusters 1–1 alignments to 1–0 or 0–1 alignments, all alignments will have 1 sentence on at least one of the two sides. By default, the R value, the upper bound to the number of allowed sentences on either side of the alignment, is set to 4. The final alignments are found when an optimal score has been obtained for the whole matrix.

Experimental Settings
In our experiments, we use Gargantua using the default settings.

5.2.4 Bleualign

Bleualign (Sennrich and Volk, 2010, 2011) uses MT and BLEU to align sentences. It needs an MT system for at least one of the translation directions to generate translations of the sentences in at least one of the languages. It then compares the resulting translations to sentences in the target language to determine whether they are likely to be a translation of the original sentence. For the comparison, it uses BLEU to generate a similarity score. Bleualign is evaluated on a set of manually aligned texts consisting of parallel German/French data from the Text+Berg corpus[5] (Volk et al., 2010). This evaluation set has commonly been used for evaluating subsequent sentence alignment systems and we use it for our evaluation.

Scoring
Even though BLEU has been criticised as a measure of translation quality and is not considered reliable on a sentence level (Callison-Burch et al., 2006), the authors of Bleualign point out that judging the quality of a translation is harder than deciding whether two sentences

[5] Available at `https://github.com/rsennrich/Bleualign/tree/master/eval`

are possible translations of each other. Furthermore, they find that BLEU is very sensitive to misalignments, indicating that it should be capable of discriminating between aligned and unaligned sentence pairs.

BLEU is usually measured on up to 4-grams. This yields a score of 0 for sentence pairs too often for the purposes of sentence alignment, especially when using low performance MT systems. Thus Bleualign uses 2-grams, which yield better results. Furthermore, when comparing two sentences, s and t, the BLEU scores are different depending on which of the sentences is the hypothesis, due to the brevity penalty in BLEU. Therefore, Bleualign goes in both directions and uses the mean as the final score.

Algorithm

The alignment algorithm consists of two passes. In the first pass, anchor points are identified after translating the source language sentences and comparing the translations to the target text sentences. These anchor points are a set of 1–1 beads considered reliable based on BLEU as a similarity measure as well as on sentence order. Given a set of target sentences and another set of source sentence translations, the BLEU score is calculated for all members of the Cartesian product of the two. The best-scoring alignments are considered if they fall into order according to the position of the source and target sentences in their respective texts, with crossing alignments not allowed. To generate a final ordered list of beads, a shortest-path algorithm is used to find the path that maximizes the BLEU score.

In the second pass, all unaligned sentences, gap-sentences, that fall between the anchors, are extracted and aligned using a number of heuristics. At first Bleualign tries to find if any of the 1–1 beads are in fact part of a 1–n or n–1 alignment ($n > 1$). It does so by creating a list of all possible 1-, 2- or 3-sentence sequences that are composed of the gap-sentences and the sentences on either side of the gap. Then BLEU scores are calculated for the Cartesian product of the two lists, and if any 1–n bead scores higher than the original 1–1 bead it is replaced in the graph and the step is repeated. If not, analogous checks are done for n–1 alignments. When no new beads are found, a new search for 1–1 alignments is done in the gap. If a bead is found, the previous heuristic is repeated. If the gap size is down to 0 on either language side or if its size is asymmetrical by a factor of more than two, the remaining sentences are left unaligned. Otherwise Gale–Church can be used to find alignments between all remaining sentences in the gap. In this case, Gale–Church compares source translations to target sentences in order to be more robust for unrelated language pairs which could have very different lengths.

Finally, the algorithm can be run in both directions and an intersection of the results selected for high-precision results. Sennrich and Volk (2011) suggest in a follow-up to the original Bleualign paper to use an iterative approach for better alignment quality. In that paper, they bootstrap an MT system by first using Gale–Church to create alignments and then use these alignments to train an MT system that translates for Bleualign. They then iteratively improve the MT system by repeating the process multiple times, resulting in slightly better alignments.

Experimental Settings

In our experiments, we use OPUS-MT[6] (Tiedemann and Thottingal, 2020), both to translate the evaluation sets in Section 5.3.1 and the parallel documents in Section 5.3.3. In all cases, we supply Bleualign with translations generated in both translation directions.

[6] https://opus.nlpl.eu/Opus-MT/

5.2.5 Vecalign

In their Vecalign paper, Thompson and Koehn (2019) presented novel approaches both to scoring sentence-pair candidates and to the alignment algorithm. Vecalign uses multilingual sentence embeddings to calculate a normalized cosine distance between source and target sentences together with an approximation algorithm (Salvador and Chan, 2007), which make the algorithm linear in time and space complexity with respect to the number of sentences being aligned.

The authors reported a considerable gain over previous state-of-the-art methods both in terms of runtime and sentence alignment quality as measured by the evaluation sets released with Bleualign and by BLEU scores on downstream MT tasks.

Scoring

The authors of Vecalign propose using the similarity between sentence embeddings as the scoring function for sentence alignment. They use the LASER multilingual sentence embedding method and model (Artetxe and Schwenk, 2019b). An advantage of sentence embeddings is that blocks of sentences can be represented as the average of their sentence embeddings and the size of the resulting vector is independent of the number of sentence embeddings being averaged. Cosine similarity can then be used to compare the embeddings. It has been noted that embeddings can be globally inconsistent due to the “hubness” problem, described in Section 3.6.6 To tackle that problem, the embeddings can be normalized using nearest neighbours, as described in Section 3.6.6. In Vecalign on the other hand, the embeddings are normalized with embeddings randomly sampled from the given document instead of nearest neighbours, as that has linear complexity. The authors of Vecalign find that DP with cosine similarity favours many-to-many alignments over 1–1 alignments, even though the many-to-many alignments could be split into multiple 1–1 alignments. As sentence alignment seeks for minimal parallel units, they scale the cost by the number of source and target sentences being considered in a given alignment. Furthermore, they model insertions and deletions using a skip cost, which can only be meaningful in comparison to other costs, and is not expected to generalize across languages or normalizations. They thus define it in terms of the distribution of 1–1 alignment costs at alignment time.

Alignment Algorithm

Vecalign uses a DP algorithm to find the best path through the alignment matrix, which uses recursive approximation to reduce the search space. Salvador and Chan (2007) propose this approach for dynamic time warping but the authors of Vecalign are the first to apply it to sentence alignment. The approach works by first averaging adjacent pairs of sentence embeddings in both search and target documents, thus halving the number of embeddings for each document and producing approximate sentence alignments. This can be applied recursively. In this phase, only insertions, deletions and 1–1 alignments are considered. Finally, the approximate alignment can be refined using the original sentence vectors, constraining the model to a small window around the approximate alignment. In that phase, 1–n, n–1 and n–m alignments are also considered.

Experimental Settings

In our experiments, we use Vecalign with the LASER2 (Heffernan et al., 2022) embeddings model and allow up to 5 adjacent sentences to be merged for each alignment (n-m alignments, where $0 \leq n \leq 5$ and $0 \leq m \leq 5$).

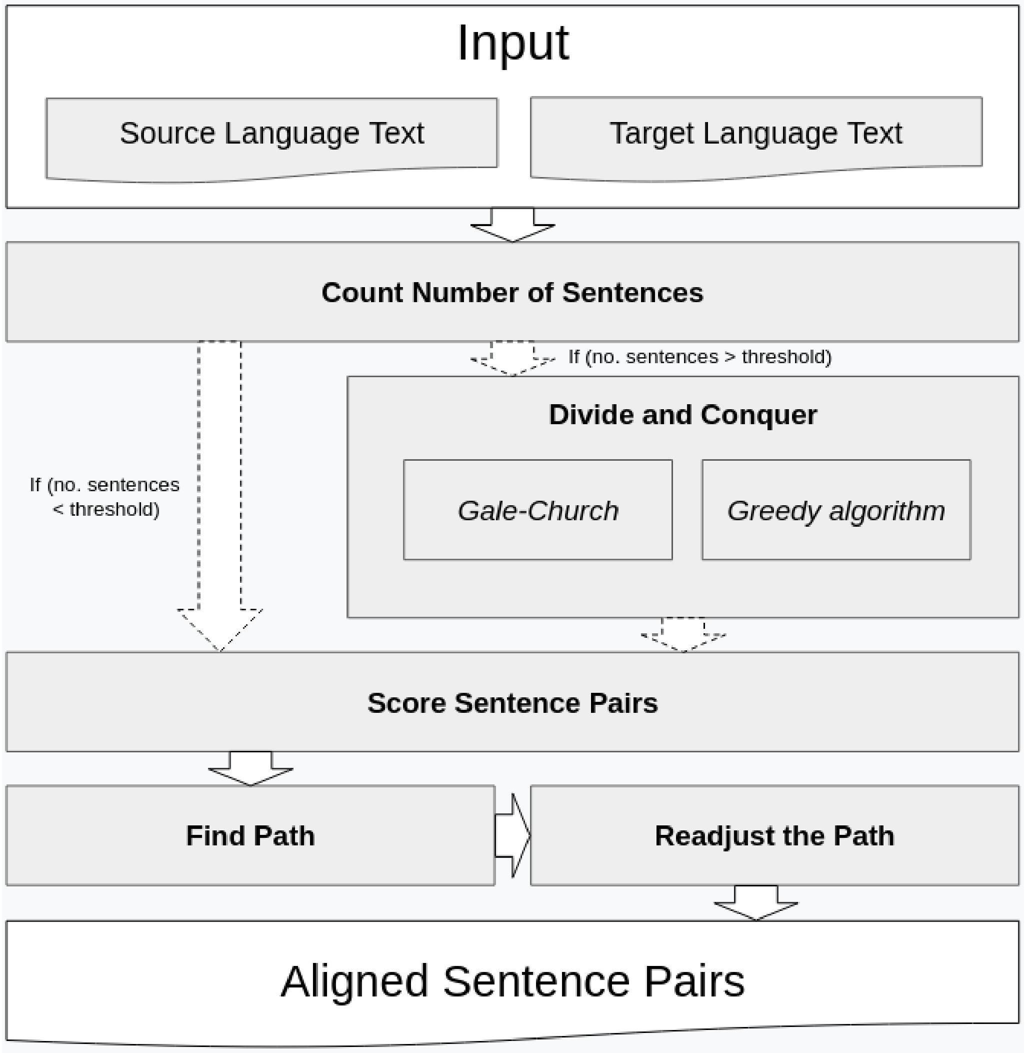


Figure 5.1: SentAlign module architecture.

5.2.6 SentAlign

We now present our own tool for sentence alignment, SentAlign.[7] It is capable of evaluating all possible alignment paths through fairly large documents, while using a LaBSE-based scoring mechanism resulting in highly accurate alignments, as shown by our experiments. While this is done to some extent at the cost of speed, the need for cutting corners in terms of computation and memory is not as acute as it was when many of the previous sentence aligners were developed. Furthermore, it can be argued that quality is much more important than speed, as alignment has only to be done once for a given parallel corpus. Having said that, our approach is of quadratic complexity, $O(n^2)$. In order to handle very large files, we therefore apply a divide-and-conquer (DaC) approach. When parallel documents have over $2,000$ sentences on each side, or equivalently when total nodes are lower than $4,000,000$, the DaC mechanism is activated in order to reduce the time complexity for these files, potentially to $\mathcal{O}(n \log n)$.

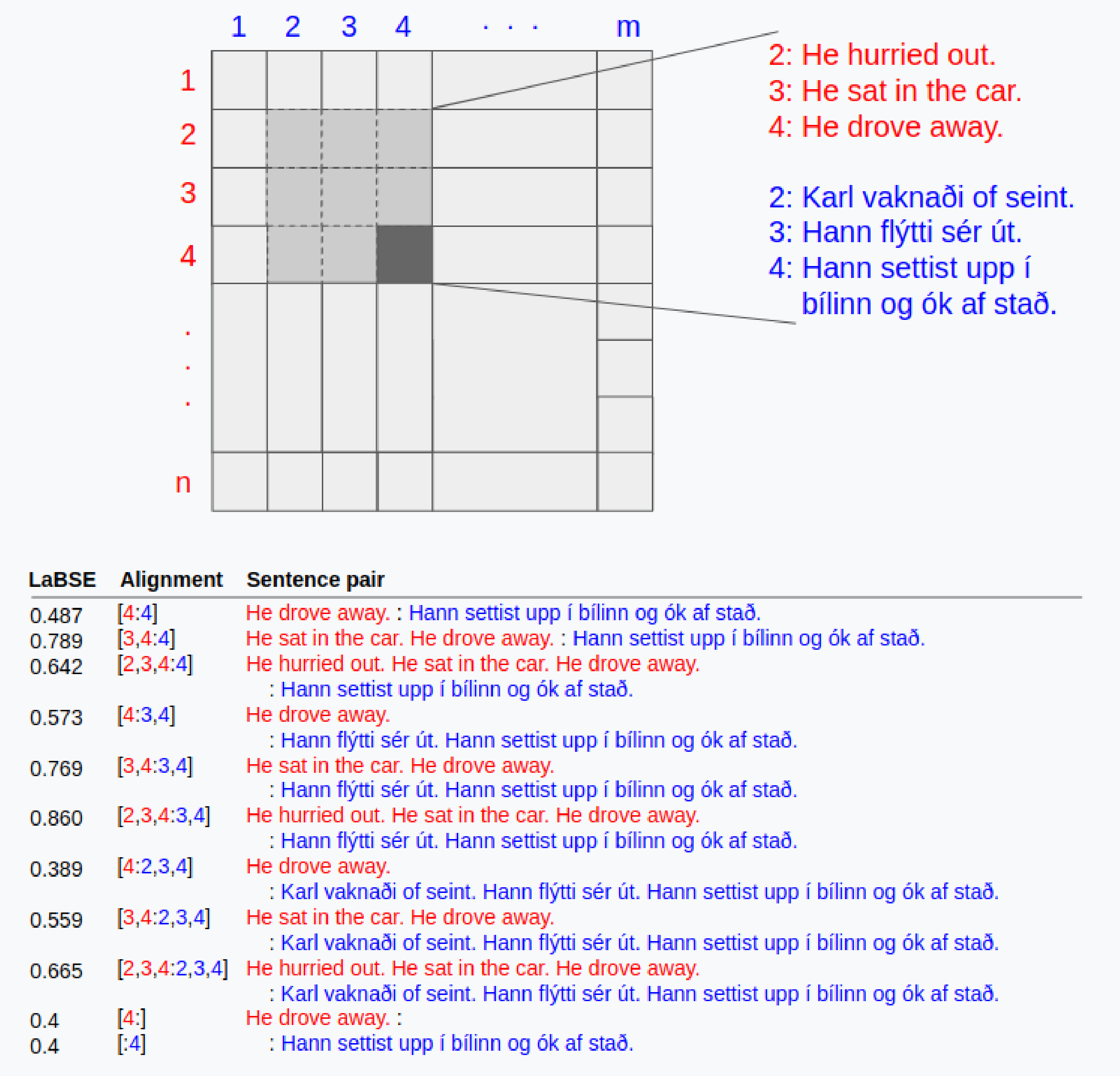
On a desktop computer running an i5-12600K processor, with 64 GB of memory and a GeForce RTX 3090 graphics card to calculate the LaBSE scores, it took approximately 13 hours to align 16501 file pairs containing a total of over 3,500,000 sentences in each

[7] `https://github.com/steinst/SentAlign`

language. This is similar to the time it takes to run many of the sentence aligners previously described. We describe SentAlign, illustrated in Figure 5.1, in more detail below.

Scoring

We use LaBSE to score sentence-pair candidates. A minimum threshold score, defined by the user, is required for a sentence pair to be accepted. For each node $[i : j]$ (where i is a sentence in the source language and j is a sentence in the target language) in the alignment graph, scores for all possible alignment combinations ending in that node are calculated. If merging up to three sentences on either side is allowed, both in source data and target data, this means comparing scores for $[[i], [j]]$, $[[i], [j-1, j]]$, $[[i], [j-2, j-1, j]]$, $[[i-1, i], [j]]$, $[[i-1, i], [j-1, j]]$, $[[i-1, i], [j-2, j-1, j]]$, $[[i-2, i-1, i], [j]]$, $[[i-2, i-1, i], [j-1, j]]$, $[[i-$



2: He hurried out.
3: He sat in the car.
4: He drove away.

2: Karl vaknaði of seint.
3: Hann flýtti sér út.
4: Hann settist upp í bílinn og ók af stað.

LaBSE	Alignment	Sentence pair
0.487	[4:4]	He drove away. : Hann settist upp í bílinn og ók af stað.
0.789	[3,4:4]	He sat in the car. He drove away. : Hann settist upp í bílinn og ók af stað.
0.642	[2,3,4:4]	He hurried out. He sat in the car. He drove away. : Hann settist upp í bílinn og ók af stað.
0.573	[4:3,4]	He drove away. : Hann flýtti sér út. Hann settist upp í bílinn og ók af stað.
0.769	[3,4:3,4]	He sat in the car. He drove away. : Hann flýtti sér út. Hann settist upp í bílinn og ók af stað.
0.860	[2,3,4:3,4]	He hurried out. He sat in the car. He drove away. : Hann flýtti sér út. Hann settist upp í bílinn og ók af stað.
0.389	[4:2,3,4]	He drove away. : Karl vaknaði of seint. Hann flýtti sér út. Hann settist upp í bílinn og ók af stað.
0.559	[3,4:2,3,4]	He sat in the car. He drove away. : Karl vaknaði of seint. Hann flýtti sér út. Hann settist upp í bílinn og ók af stað.
0.665	[2,3,4:2,3,4]	He hurried out. He sat in the car. He drove away. : Karl vaknaði of seint. Hann flýtti sér út. Hann settist upp í bílinn og ók af stað.
0.4	[4:]	He drove away. :
0.4	[:4]	: Hann settist upp í bílinn og ók af stað.

Figure 5.2: An example of how SentAlign finds the maximum score for a node in the alignment matrix. In this example, we are searching for the best alignment that ends in node [4:4]. We merge up to three sentences ending in that node and calculate the LaBSE score for each sentence pair. Note that for the null alignments, where either one of the sentences is discarded, we assign a minimum threshold score. In our experiments it is set to 0.4, found by searching for the best settings on the development set from the Text-Berg corpus. The scoring information is used in the pathfinding step to choose the optimal alignment.

$2, i-1, i], [j-2, j-1, j]]$, a total of $3 \times 3 = 9$ scores for each node. If no sentence pair that can be represented in a given node in the alignment graph reaches the threshold, an insertion or deletion is selected and allotted the minimum threshold score. Sentence pair scoring is illustrated with an example in Figure 5.2. If the number of words in either language exceeds a user-defined maximum, a penalty is applied to the alignment score. To find the maximum score for a node, this alignment score is then added to the score of the node it connects from, e.g. $[i-1 : j-1]$ for $[i : j]$, as that alignment has one sentence on each side.

Divide and Conquer

If the search space is larger, in terms of a number of nodes to consider, than a pre-defined threshold allows (the default is $4,000,000$), we search for high-confidence alignments (i.e. hard delimiters) to divide the search space into multiple smaller chunks to align separately, $k+1$ chunks for k hard delimiters. Zhang (2022) shows that for a quadratic time complexity sentence-alignment algorithm, “clumping” the parallel texts to be aligned using hard delimiters can reduce the time complexity to $O(n \log n)$. Our aim is to find the minimum amount of alignments to use as hard delimiters to split our parallel texts into chunks of manageable size.

There are two stages involved in finding the high-confidence alignments. Our first approach, used for large files above a user-defined divide-and-conquer threshold, first employs the Gale–Church algorithm to align the parallel text/chunk under consideration and then scores all resulting alignments using LaBSE. The highest-scoring alignment, which meets the required conditions (which include limits to how close the alignment can be to previous hard delimiters), is chosen as a high-confidence alignment.

If the parallel files are very large, running Gale–Church will take an excessive amount of time. Therefore, we have a fallback approach to be used when file size surpasses another threshold. In that case, we do not use Gale–Church but rather resort to a greedy algorithm that only looks for the highest scoring 1–1 alignments. This threshold can be user-defined, but by default (as well as in our experiments) it is set to $10,000$ sentences for either language.

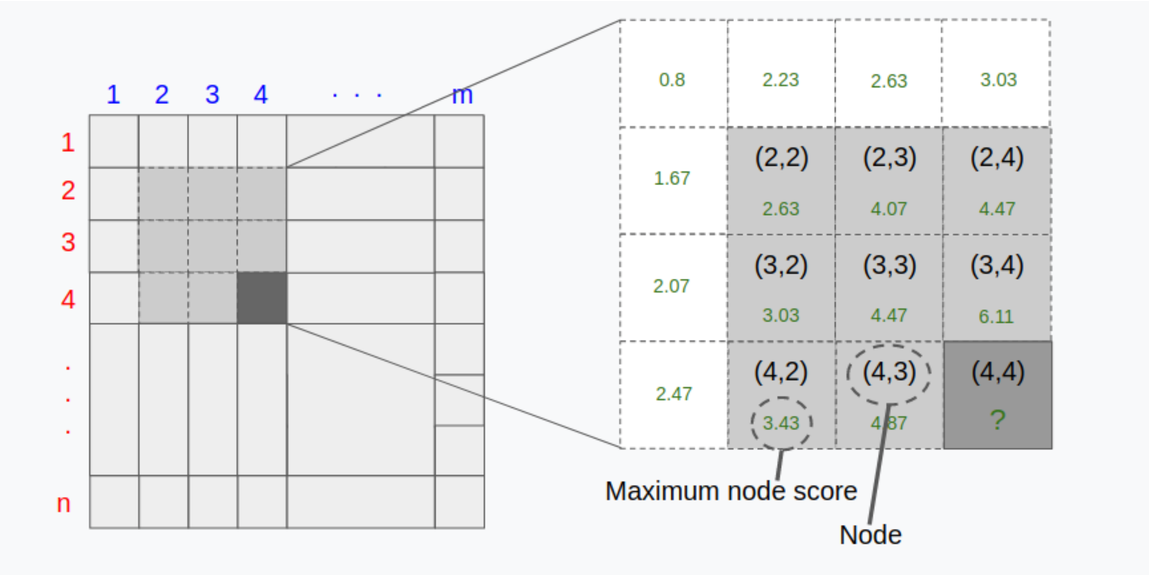
Both approaches only consider 1–1 alignments and only select alignment from the middle half of the parallel texts, if possible, with the middle half defined as the sentences in between the first and last 25% of the sentences in the texts. From there we select the highest-scoring alignment, split the parallel text into two chunks, and if the chunks are still too large repeat the process until the desired search space size is achieved.

Pathfinding

If we look at the alignment problem as a way of finding the optimal path through an $N \times M$ matrix, where N is the number of source sentences and M is the number of target sentences, we need to search from the initial node $[0, 0]$ to the final node $[n, m]$. As we have to check for insertions and deletions for all nodes, this means we have to calculate the cost for each node to find the optimal path. Dijkstra’s algorithm (Dijkstra, 1959) finds the shortest path from a node to all other nodes within a graph, by visiting all the nodes and calculating the minimum cost of arriving there. In our case, we calculate a maximum score for the path to each node. Edges between nodes in the graph are labelled with positive numbers, which the algorithm uses to calculate the score when travelling to that node.

When the aligner is configured, we set a maximum number of sentences that can be merged in each language for each alignment. We have to account for that during alignment when we calculate the cost to reach each node. For example, if the number is 3 and we are calculating the cost to reach node $(4, 4)$, we need to know the cost to reach there from $(1, 1)$, $(1, 2)$, $(1, 3)$, $(1, 4)$, $(2, 2)$ and all up to $(4, 3)$, with the alignment starting in $(2, 2)$

being a 3–3 merge, the alignment from $(3, 4)$ being a 1–0 deletion and the alignment from $(4, 3)$ being a 0–1 insertion. In the case of deletions and insertions, we use the minimum score threshold and assign that value to the edge, which we set as the cost of including one non-aligned sentence in the path. If one or more of the possible n–m ($n \geq 1$) alignments has a LaBSE score above the threshold, after applying penalties we select the highest-scoring one as described above. The edge label is assigned a value equaling the LaBSE score multiplied by the total number of sentences merged in both languages. That way each sentence is assigned the penalty-adjusted LaBSE score of the alignment of which it is a member. The



Alignment	Score	Penalty	From node	Calculated max node score
[4:4]	2 x 0.49	0	(3,3)	5.45
[3,4:4]	3 x 0.79	0	(2,3)	6.44
[2,3,4:4]	4 x 0.64	0	(1,3)	5.19
[4:3,4]	3 x 0.57	0	(3,2)	4.74
[3,4:3,4]	4 x 0.77	0	(2,2)	5.71
[2,3,4:3,4]	5 x 0.86	0	(1,2)	6.53 Optimal alignment
[4:2,3,4]	4 x 0.39	0.02	(3,1)	3.61
[3,4:2,3,4]	5 x 0.56	0.02	(2,1)	4.45
[2,3,4:2,3,4]	6 x 0.67	0.02	(1,1)	4.80
[4:]	1 x 0.40	0	(3,4)	6.51
[:4]	1 x 0.40	0	(4,3)	5.27

Figure 5.3: An example of how SentAlign finds the optimal alignment to a given node. When scores for all sentence pairs representing the alignments that ends in a given node have been calculated (see Figure 5.2), the score to reach the node is calculated by adding the alignment score to the maximum score of the node the alignment leads from. The score is assigned to each of the sentences that are merged and the sum for all merged sentences added to a previous score. To find the optimal alignment to node (4,4) in the figure, we calculate scores for each of the possible alignments and then add the scores to the maximum node score in the node the alignment connects to. Alignment [2,3,4:3,4] has five sentences, three in the source language and two in the target language, and thus the LaBSE score is multiplied by five. The last node in the previous alignment is (1,2), as the first sentences in this one are sentences 2 and 3. When the maximum score in node (1,2) is added to the alignment score, we obtain a score of 6.53, which is higher than the scores for all other alignments. Thus, this alignment is selected for this node. When maximum scores have been found for all nodes the algorithm works its way back to collect all alignments for the highest scoring path. Three alignments are assigned a penalty, which is subtracted from the final score. The penalty is assigned to an alignment when the length of the target sentence exceeds the maximum recommended sentence length.

value of the selected edge is then added to the value of the node it leads from, the total set as the current node's value and the edge leading to that node recorded as the optimal path there. Figure 5.3 illustrates this example of finding the optimum alignment to a node in the alignment graph. This process is repeated for each node until node (n, m) is reached. We then have the optimal score from $(0, 0)$ to (n, m) and mark the path by tracing backwards through the recorded edges.

Readjusting the Path
As Thompson and Koehn (2019), among others, have argued, sentence alignment should seek a minimal parallel pair, the pair having the fewest mergers while still being truly parallel. An effect of the scoring and alignment mechanism described above is that sometimes large mergers are preferred over smaller ones that provide better alignments, thus counteracting the desired results. This can happen when a LaBSE score for large mergers (say $3{\times}4$) is high enough to attain a higher average score from the source node to the end node, than if a better alignment with fewer sentences was selected (say $2{\times}2$) along with a lower-scoring substitution and an insertion. To counter that, we finish our process by reevaluating each alignment in the selected path by taking another look at mergers, insertions and deletions.

1. First, we investigate all $n{\times}m$ alignments, where ($n > 1$) and ($m > 1$), and search for the highest-scoring alignment which is a subset of the one we are investigating. If one is found that has a higher score than the original alignment, we look through the remaining sentences for another sentence pair scoring above the LaBSE threshold. We add the discarded sentences to the list of null alignments, consisting of previous insertions and deletions.

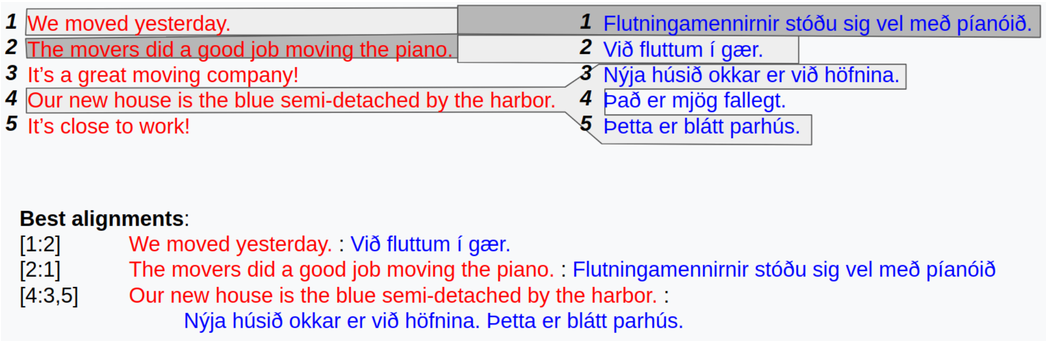
2. We look at the list of non-aligned source and target sentences and reevaluate whether the non-aligned sentences should remain discarded or be merged with the surrounding alignments. If a non-aligned sentence is adjacent to a sentence which has been aligned, we try merging it to that alignment and calculate the LaBSE score. If the score rises, we keep them merged and amend the path. This is repeated until all possible amendments have been made.

When this reevaluation is done, we return the set of alignments generated by the selected path through the alignment matrix.

Experimental Settings
For our experiments, we searched for the best settings for the alignment tool parameters using the development set from the Text-Berg corpus. We found the best LaBSE threshold to be 0.4, maximum number of words per language before applying a length penalty to be 80, and the penalty for each word exceeding that maximum to be 0.01. We performed a complete search through the alignment matrix, without chunking the search space by finding anchors, if the search space had less than $4,000,000$ nodes. If a file had more sentences than $10,000$ in either one of the languages, we did not run Gale–Church but instead searched for anchors only using the greedy algorithm.

5.3 Evaluation and Results

We compare the six sentence alignment systems in three different ways: using sentence alignment evaluation data sets, by manually inspecting the resulting alignments, and by calculating BLEU scores on NMT systems trained on alignments obtained from the corpus of



1 We moved yesterday.
2 The movers did a good job moving the piano.
3 It's a great moving company!
4 Our new house is the blue semi-detached by the harbor.
5 It's close to work!

1 Flutningamennirnir stóðu sig vel með píanóið.
2 Við fluttum í gær.
3 Nýja húsið okkar er við höfnina.
4 Það er mjög fallegt.
5 Þetta er blátt parhús.

Best alignments:
[1:2] We moved yesterday. : Við fluttum í gær.
[2:1] The movers did a good job moving the piano. : Flutningamennirnir stóðu sig vel með píanóið
[4:3,5] Our new house is the blue semi-detached by the harbor. :
Nýja húsið okkar er við höfnina. Þetta er blátt parhús.

Figure 5.4: Examples of non-monotonic alignments that most alignment algorithms do not allow. The first two are examples of crossed alignments. Source sentence 1 aligns with target sentence 2, while source sentence 2 aligns with target sentence 1. In the last example, a target sentence is skipped from in between two target sentences that are merged for the alignment. Most alignment algorithms do not allow this. Aligned sentences on both sides must be a continuation from previous alignments and only adjoining sentences are allowed to merge.

$16,501$ aligned documents which make up the EEA portion of the ParIce corpus. Furthermore, we use an ensemble of these six alignment tools to extract anchor points in all document pairs, and then use the best aligners to realign the text between the anchor points. The hypothesis is that if the majority of the sentence aligners agree on an alignment, it is highly likely that it is correct. We only measure the results of the ensemble approach in terms of effect on the downstream NMT task.

5.3.1 Measuring Against Evaluation Sets

The manually aligned German–French evaluation set created from the Text+Berg corpus (Volk et al., 2010), first used to evaluate Bleualign, is commonly used for sentence alignment evaluation. While most alignment tools, with the exception of Bleualign, do not allow reordering of sentences (see examples illustrated in Figure 5.4: crossing alignments $[2,1]$ and $[1,2]$ where source language sentence s_2 aligns with target language sentence t_1, s_1 aligns with t_2, and sentences are skipped when merging as in $[4:3,5]$ where t_4 is deleted while t_3 and t_5 are merged). There are examples of such alignments in this evaluation set, which makes it impossible for the aligners to attain a perfect score. Furthermore, a few entries of null alignments are missing from the Bleualign files. To try to be consistent with previous reported scores, we do not make any changes to the evaluation set, but we do use a slightly amended version of the evaluation script provided with Vecalign. There, precision is calculated using equation (5.1):

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{5.1}$$

As only some null alignments are included in the evaluation set and some are not, the results can be different based on whether a given sentence aligner returns null alignments or only useful alignments. We thus only calculate precision on non-null alignments, i.e. alignments that are true sentence pairs. Furthermore, Vecalign calculates *recall* as precision with no insertion/deletion or swaps. That is very similar to how we calculate *precision*. We change the recall calculation and, while we still only consider non-null alignments, we use equation (5.2) to calculate recall:

Alignment results on Text+Berg						
	Strict			Lax		
Algorithm	P	R	F_1	P	R	F_1
Gargantua	0.76	0.75	0.76	0.89	0.78	0.83
Hunalign	0.66	0.69	0.67	0.86	0.74	0.80
Gale–Church	0.68	0.69	0.69	0.80	0.73	0.76
Vecalign	0.90	0.90	0.90	0.99	0.91	0.95
Bleualign	0.93	0.66	0.77	**1.00**	0.68	0.81
SentAlign	**0.94**	**0.92**	**0.93**	**1.00**	**0.93**	**0.96**

Table 5.1: Results of the evaluation of different sentence alignment systems using the German–French Text+Berg evaluation set. Note that Hunalign does not use any external dictionary here. The highest scores are in bold. Our SentAlign algorithm outperforms all systems both for the strict and lax conditions, although Bleualign has a perfect score for precision, just like SentAlign. Vecalign scores under the lax condition are very close to SentAlign, while other systems are far behind.

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{5.2}$$

Following the original Bleualign paper, in Table 5.1 we report results both under the strict condition where exact matches between the gold alignment and the hypothesis are demanded, and under the lax condition where a hypothesis is true if there is an overlap with a gold alignment on both language sides. Under the lax condition, a 2–2 alignment, which is recognized as two 1–1 alignments, will yield two true positives, while it would have yielded two false positives under the strict condition.

We compiled an evaluation set for English–Icelandic sentence alignment from five subcorpora of the ParIce corpus. The evaluation set (Steingrímsson, 2021) is available under an open license and contains a total of 549 sentence alignments from 10 aligned documents from these subcorpora.[8] These documents are arguably easier to align than the Text+Berg documents as none of them contain long stretches of non-alignments and there are few n–n ($n > 1$) merge alignments.

We repeat the same experiment as before for this alignment set, using all the same settings as before for all aligners. In addition, we run Hunalign employing the English–Icelandic lexicon described in Section 3.6. As shown in Table 5.2, the scores are higher for all aligners except for SentAlign. The reason why the scores are not higher for SentAlign may be due to the fact that we are missing a development set for the English–Icelandic language pair and are using the parameters set for the Text+Berg de–fr development set. The acceptance threshold for the scoring mechanism we use, LaBSE, may be different for different language pairs. While we found that 0.4 was the optimum threshold score for the Text+Berg corpus, Feng et al. (2022) set their threshold when mining sentences from CommonCrawl to 0.6. This suggests that analysis of the data to be processed could be useful on a case-by-case basis. Another possibility is that our SentAlign is more accurate than other alignment systems when dealing with multiple insertions/deletions in a row or with multiple merges. As this data set does not contain many such alignments this advantage is not useful here, but it is worth noting that it may play a role in other experiments.

As with the evaluation set from Text+Berg, the sentence embeddings-based alignment systems SentAlign and Vecalign are the most accurate. Bleualign is the third best system,

[8] http://hdl.handle.net/20.500.12537/150.

Alignment results on English–Icelandic evaluation set						
	Strict			Lax		
Algorithm	P	R	F_1	P	R	F_1
Gargantua	0.82	0.76	0.79	0.89	0.78	0.83
Hunalign	0.72	0.75	0.73	0.87	0.78	0.82
Hunalign+dict	0.74	0.76	0.75	0.89	0.79	0.84
Gale–Church	0.78	0.79	0.79	0.87	0.81	0.84
Vecalign	0.92	**0.94**	**0.93**	0.97	**0.95**	**0.96**
Bleualign	**0.93**	0.78	0.85	0.98	0.79	0.88
SentAlign	**0.93**	0.92	**0.93**	**0.99**	0.92	0.95

Table 5.2: Results of the evaluation of different sentence alignment systems using an English–Icelandic evaluation set. The highest scores are in bold. While SentAlign outperforms other systems in terms of precision, Vecalign has a slightly higher F_1 score under lax conditions. These two systems still fare considerably better than the other systems.

but in order to achieve good accuracy it needs a good MT system to base its scores on. By replacing the OPUS-MT models with higher quality models, Bleualign could possibly be further improved.

5.3.2 Manual Evaluation of Aligned Pairs

We created multiple MT training datasets from the EEA subcorpus of ParIce. We used the different sentence alignments systems separately, as well as combining the best-performing ones with anchor point alignments selected with a majority vote by an ensemble of the aligners. The best-performing systems were defined as those obtaining the highest BLEU scores in the downstream MT task described in Section 5.3.3. After running sentence alignment, we filtered the data. First, we applied the shallow filters described in Section 4.2.2. As our intention here is to measure the effect of different sentence aligners, we opted for keeping other variables as constant as possible, and selected one filtering approach to be used for all the data, regardless of the translation direction (see Table 5.4). In Chapter 4, we found that three of our approaches scored highest for Icelandic→English, with none being significantly better than the others. As two of the sentence aligners we use have scoring mechanisms based on sentence embeddings, we chose a filter that does not primarily use such embeddings, but performs similarly well to a filter that does so. This is our logistic regression classifier, which classifies sentence pairs based on four different scores: LaBSE, NMTScore, LASER and WAScore (see Section 4.2.2). By selecting a filter not solely based on sentence embeddings, we hope to reduce homogeneity in the selection process and search for detrimental alignments from different points of view.

In order to investigate whether a manual inspection of the aligned data could be an indicator of quality in a downstream NMT task, we randomly selected 250 sentence pairs from each training dataset and performed a manual evaluation using the same taxonomy as in Section 4.2.3.[9] Table 5.3 gives the results of the manual evaluation. The results suggest that

[9] The evaluation was carried out by two people, me and Finnur Ágúst Ingimundarson. Finnur received a list of all the sentences in random order and annotated each sentence. I then checked the list and if I strongly disagreed with the annotation of a sentence pair I changed it. This sort of data is rather straightforward to label and we should not expect high disagreement. For the 3000 alignments I made 88 changes, so the two annotators agreed 97.1% of the time. We calculated Cohen's Kappa (see Section 3.6.8) and obtained κ=0.815, indicating

Manual Evaluation of Aligned Sentence Pairs

Aligner	CC	CB	CS	X
Gale–Church	225	22	0	3
Bleualign	**242**	8	0	0
Gargantua	230	18	1	1
Hunalign	230	19	0	1
Hunalign+dict	226	23	0	1
Vecalign	221	28	0	1
SentAlign	**242**	8	0	0
Anchors+Bleualign	239	10	0	1
Anchors+Gale–Church	239	11	0	0
Anchors+Hunalign+dict	**243**	5	0	2
Anchors+Vecalign	237	13	0	0
Anchors+SentAlign	241	9	0	0

Table 5.3: Results of the manual evaluation of samples of 250 randomly selected sentence pairs from the EEA subcorpus in ParIce, generated by different alignment approaches. The taxonomy is the same as described in Table 4.1 in Chapter 4. CC is a correct translation of good quality; CB is a correct translation, but quality is lacking; CS is a correct translation, but either one of the sentences is very short; X is a wrong translation.

all training sets mostly contain valid alignments. The alignments that deviate from that are usually annotated as CB, meaning that it is a correct translation, but of low quality or contains a partial misalignment, as defined in the taxonomy shown in Table 4.1. We categorized further all the alignments in the CB class, and found that out of a total of 213 CB annotated alignments, 19 were classified in such a way due to spelling errors, usually derived from inadequate OCR, and all other CB annotated alignments were partial alignments, ranging from one or two extraneous words on either side of one of the aligned sentences to added text that made one side of the alignment more than double the length of the other. Inspecting the faulty alignments, Hunalign seems to be the most likely to have lengthy misalignments, while the sentence embeddings-based aligners seem least likely to produce such pairs.

5.3.3 Downstream MT Task

We used fairseq to train Transformer$_{\text{BASE}}$ models using the settings described in Section 3.5. The results are reported in Table 5.4. As the main task was to align the EEA subcorpus from ParIce, we used the EEA development and evaluation sets described in Section 3.5.3. We calculate statistical significance using the pairwise bootstrap test as before and found that our SentAlign aligner achieved the best results, 42.8 and 53.6 as measured by BLEU, for en→is and is→en, respectively. This is significantly better than all other aligners with the exception of Hunalign when using our English–Icelandic lexicon, which obtained a BLEU score of 42.4. Using anchors also seems to be useful, with the anchored models overall obtaining stronger results than the non-anchored ones, and the best one, *Anchors+SentAlign*, reaching the highest score in our alignment experiments, 43.4 and 54.0 BLEU for en→is and is→en, respectively. While the majority of the anchor-using models are stronger than the ones not using anchors, Bleualign and Vecalign are exceptions to that. They obtain slightly but not

almost perfect agreement. Having reached agreement for the annotations, I alone did the further categorization of all alignments classified as CB.

Training with data produced by different alignment tools			
Sentence Aligner	no. pairs	en→is	is→en
Gale–Church	683,813	41.8	51.4
Hunalign	717,879	41.4	52.1
Hunalign+dict	798,558	**42.4**	53.0
Gargantua	606,768	39.1	48.9
Bleualign	627,019	42.0	53.0
Vecalign	670,595	41.8	51.7
SentAlign	877,485	***42.8***	***53.6***
Aligned after anchoring datasets with ensemble vote			
Sentence Aligner	no. pairs	en→is	is→en
Anchors+Hunalign+dict	800,564	**43.2**	**53.7**
Anchors+Gale–Church	837,446	42.2	52.3
Anchors+Bleualign	778,587	42.1	52.7
Anchors+Vecalign	883,693	42.7	51.6
Anchors+SentAlign	903,692	***43.4***	***54.0***

Table 5.4: Number of pairs and BLEU scores for different alignment approaches. Bold and italic scores are the highest scores for each category. Bold and non-italic scores are insignificantly lower than the highest score. Other scores are significantly lower. For the is→en translation direction, the results from using the ensemble anchors and SentAlign are insignificantly higher than only using SentAlign.

significantly lower BLEU scores for is→en when using the anchor sentences. There may be various reasons for that. Possibly this is related to the training being stopped when the model has not improved on the validation set for 10 epochs. An indicator of that being a factor is the fact that the *Anchors+Vecalign* setup scored surprisingly low in comparison to the results on the evaluation data in Section 5.3.1, stopped training earlier than all other models, after 53 and 47 epochs for en→is and is→en, respectively. Most other models stopped training after 70 to 90 epochs and the *Hunalign+dict* model trained for the longest amount of time and did not stop until after 88 and 103 epochs for en→is, is→en, respectively. We use a development set of just over 2,000 sentences from a rather homogeneous data set. Perhaps a larger or more extensive validation set would result in more optimal stopping points at training time.

5.4 Conclusion

In this chapter, we introduced a new sentence aligner and showed by experiments and comparison to previously published systems that it is very competitive. We also showed that by using an ensemble of sentence aligners to find hard delimiters in a parallel corpus, the quality can be improved somewhat, as measured by BLEU in a downstream MT task. While using sentence embeddings seems to be very useful for this task, our experiments show that using the lexical-based Hunalign system aided by an external dictionary can also give good results.

At the beginning of this chapter, we set out to answer our third research question: Is sentence alignment accuracy important for the results of a downstream MT task, or is effective filtering of the training data enough? In order to answer that question, we consider the results of aligning the evaluation sets to be a test of aligner accuracy and the BLEU scores in the downstream task a quality measure of the MT system. Tables 5.1 and 5.2 give accuracy scores for Vecalign and SentAlign that outperform the other systems by far. The lowest-performing systems on the evaluation sets are also the lowest-performing ones in terms of BLEU scores in the downstream MT task. That said, Vecalign does not do particularly well in the downstream MT task, while SentAlign does. The difference between the systems as measured by BLEU also does not seem to be as clear as when evaluated on the evaluation sets. Overall, sentence alignment accuracy seems to be a clear indicator of downstream MT quality.

We also wanted to answer the added question of which methods could be used to help find the best alignment approaches for a given task. If evaluation sets are available for the language pair being aligned, the results provided can be a good indicator for which sentence alignment system to use. As well as using evaluation sets, we tried manually annotating a small subset of the data produced by the alignment systems after filtering. While the ranking of systems in the manual evaluation was a bit different from the ranking produced by BLEU-evaluation, we calculated correlation between the number of sentence pairs classified as CC (correct and of good quality) and BLEU scores, both for en$\rightarrow$is and is$\rightarrow$en. We found the Kendall's τ coefficient (Kendall, 1938) for each correlation, 0.45 for the correlation between en$\rightarrow$is and 0.57 for is$\rightarrow$en, both statistically significant with $p < 0.05$. This tells us that manual evaluation using the taxonomy we employ can also be helpful in determining which sentence aligner to use, if other resources are not available. By looking at the results, we see that the top-3 systems in the manual evaluation results rank number 1, 2 and 5 in the BLEU-evaluation.

In the next chapter, we will investigate methods to mine parallel sentences from comparable corpora. We will also consider if we can mine useful data from what is discarded in the parallel document sentence alignment process, by using similar methods to those used when mining comparable corpora.

Chapter 6

Comparable Corpora and Utilizing Discarded Data

In Chapter 5, we aligned parallel documents at the sentence level and extracted parallel sentence pairs from the alignments. Previously, in Section 4.2.1, we filtered Paracrawl, a corpus consisting of parallel sentence pairs mined from web-crawled corpora, to extract from it parallel sentence pairs beneficial for MT training. In this chapter, we experiment with mining our own set of parallel sentences from comparable corpora, using cross-lingual information retrieval (CLIR) and a classifier.

We experiment with extracting data from Wikipedia articles,[1] which are a very accessible source of comparable documents for a wide variety of languages. We compare the results of our mining experiment to the publicly available Wikimatrix (Schwenk et al., 2021), an extraction of parallel sentences across multiple languages. We then proceed to investigate if we can use similar methods to further process parallel sentence pairs in a parallel corpus. We test the feasibility of the idea on the Samanantar corpus of Indian languages (Ramesh et al., 2022), looking specifically at English–Bengali which we find to be somewhat noisy. Finally, we seek to answer the research question we set out for this chapter: **RQ4: Are text segments, discarded during sentence alignment and filtering, suitable as a source for mining useful sentence pairs for MT training?** We investigate whether we can apply comparable corpora mining approaches to this often overlooked potential source of comparable data. For this, we work with the ParIce corpus, particularly sentences that were excluded from the training set created using our highest-scoring setup in the downstream MT task in Section 5.3.3.

6.1 Related Work

Comparable corpora have been shown to be a useful source for mining parallel segments that can help improve MT quality (Wolk et al., 2016; Hangya and Fraser, 2019). Afli et al. (2015) extract parallel data from a multimodal comparable corpus from the Euronews[2] and TED[3] web sites. Chu et al. (2015) extract parallel texts from the Chinese and Japanese Wikipedia, and Ling et al. (2014) employ a crowdsourcing approach to extract parallel text from Twitter data in order to find the translations in tweets. Karimi et al. (2018) describe the approach for extracting parallel sentences from English–Persian document-aligned Wikipedia

[1] https://www.wikipedia.org/

[2] https://www.euronews.com/

[3] https://www.ted.com/

entries. They use two MT systems to translate from Persian to English and the reverse. The information retrieval (IR) system produces two scores that indicates the relevancy of any given translation for the original sentences, one for each original English sentence and Persian→English translations, and another for each original Persian sentence English→Persian translations. These scores are used to calculate similarity scores for each sentence pair. Multilingual sentence embeddings have also been applied to the problem, obtaining state-of-the-art performance (Schwenk, 2018; Artetxe and Schwenk, 2019b). Ramesh et al. (2022) describe the collection of parallel corpora for 11 Indic languages from diverse comparable corpora using LaBSE embeddings (Feng et al., 2022), a language-agnostic BERT sentence embedding model trained and optimized to produce similar representations for bilingual sentence pairs that are translations of each other.

Word alignments have previously been used for parallel sentence extraction. Munteanu and Marcu (2006) experiment with extracting parallel sub-sentences from comparable corpora using word alignments to link words in the source and target language and use log-likelihood-ratios to estimate probability of all word-to-word links, which they use to determine if two strings of words are parallel. Zariņa et al. (2015) identify parallel sentences using word alignments, experimenting with five different alignment-based scores. They presume that if a pair of sentences are equivalent in two languages, there should be many word alignments between the sentences, and non-parallel sentences should have few or no word alignments. Stymne et al. (2013) use alignment-based heuristics to filter out sentence pairs, and Lu et al. (2020) use a word alignment-based translation score as a part of their scoring ensemble for filtering a noisy parallel corpus. Their translation score is a simplified version of the translation score introduced by Khadivi and Ney (2005). Azpeitia et al. (2017) and Azpeitia et al. (2018) describe a method using CLIR and lexical translations obtained using word alignments, with a simple overlap metric. They obtained the highest results for the BUCC 2017[4] and BUCC 2018[5] shared tasks.

Work on sub-sentential fragment extraction includes that of Hangya and Fraser (2019), who use bilingual word embeddings to greedily align words in partially parallel sentences, and then average the word alignment scores and weight them using segment length to decide if a given segment pair is parallel. However, we are not aware of any work to date attempting to utilize discarded parallel training data.

6.2 Mining Comparable Corpora for Parallel Sentences

When parallel sentences are extracted from parallel corpora, it can usually be assumed that the sentence order in the texts is the same. In that case, as described in Chapter 5, extracting parallel sentences becomes a pathfinding, scoring and filtering problem. When dealing with comparable corpora in two languages, in contrast, we work on the assumption that two documents in two languages are not mutual translations, but that they share similar content, domain or theme. Thus, the documents potentially contain semantically equivalent sentences in the two languages. Parallel sentence candidates can usually come from anywhere in two comparable documents, i.e. a potential parallel counterpart of one sentence in the source-language document can be any sentence in the target-language document. If the average number of sentences in comparable documents is n, the number of potential sentence pairs that have to be evaluated can be up to n^2. This quickly becomes overwhelming as n increases and so it is imperative to reduce the evaluation load, ideally to a maximum of $k \times n$

[4] `https://comparable.limsi.fr/bucc2017/`

[5] `https://comparable.limsi.fr/bucc2018/`

candidates, where k is a constant number of allowed candidates for each sentence in the comparable documents.

In this section, we describe our work on extracting parallel data from comparable corpora, described in Steingrímsson et al. (2021b).[6] Our approach divides the problem into two main steps. We started by extracting parallel sentence candidates using an inverted index-based CLIR tool called *FaDA* (Lohar et al., 2016), that can be applied to documents in any two languages, provided that a bilingual dictionary for the languages is available. Using the tool for our experiments we consider each sentence to be one document. In the second step, we scored the sentence candidates using two different scores: LaBSE, based on contextualized embeddings, and WAScore, based on high-precision word alignments. Then, a logistic regression classifier selected sentence pairs based on these scores.

We performed three different tests to evaluate our approach. First, we used a BUCC-style evaluation set and, second, a manually curated set created from all sentences in fifteen Wikipedia articles. For these sets, we measured precision, recall and $F1$-scores, based on the extracted parallel sentences. Third, we measured accuracy in terms of BLEU-score on a downstream MT task, where the extracted parallel sentences were used as supplemental data for training NMT systems. The systems were compared to a baseline in order to give an indication of the usefulness of the supplemental data for NMT training.

6.2.1 Data

We worked with the English–Icelandic language pair, for which no evaluation sets had previously been made available for parallel sentence extraction from comparable corpora. Therefore, we built test sets in order to be able to evaluate our approach.[7] We prepared the following data sets for our experiments:

- *CompNews*: We generated development and test sets for identifying parallel sentences in news corpora, in the style of the test sets compiled for the BUCC 2017 shared task on parallel sentence identification (Zweigenbaum et al., 2016). The sets consist of a small set of known parallel sentences, as well as a larger list of randomly sampled sentences from monolingual corpora in the same domain, but with no known parallel pairs. The parallel sentences used are the $2,000$ English–Icelandic/Icelandic–English sentence pairs made available as development data for the news translation task in WMT 2021 (Akhbardeh et al., 2021).[8] The non-parallel sentences were randomly selected from News crawl 2018,[9] and 2018 news texts in Icelandic sampled from the IGC. This resulted in two lists of 100,000 sentences each: one list of English sentences and another list of Icelandic sentences. 2% of the sentences in each list were known to have a corresponding sentence in the other language. We made a $40/60$ split, with the true parallel sentence pairs equally distributed between the splits. The smaller part was used as a development set and the larger part as a test set.

[6] The work described in this section was carried out in cooperation with Pintu Lohar. Pintu Lohar, a researcher at Dublin City University, created the list of parallel sentence pair candidates, using the cross-lingual information retrieval tool *FaDa*, while I carried out the second step of scoring and selecting the final sentences, as well as training and evaluating the MT models.

[7] All available at: `https://github.com/steinst/bucc2021-en-is`

[8] The development set for WMT 2021 contains 1000 sentences in each translation direction. It is available at: `http://statmt.org/wmt21/translation-task.html`

[9] Available at: `https://data.statmt.org/news-crawl/en/`

- *CompWiki*: We randomly selected 15 Wikipedia articles available in both Icelandic and English. The texts were split into sentences and the CLIR tool, FaDA, was used to obtain translation candidates for each sentence. These sentence pairs were manually evaluated and marked as parallel, partially parallel, or non-parallel. Out of a total of 10,098 sentences, 86 were marked as parallel and 421 as partially parallel.

- *CompTrain*: In order to gain some information on the kind of scores the two scoring methods, LaBSE and WAScore, give to non-parallel data, on the one hand, and parallel data, on the other hand, we compiled a dataset with 50,000 randomly sampled pairs from the two monolingual corpora used for CompNews and added parallel sentences from version 1 of the ParIce corpus. We selected 2,500 random sentence pairs from a development set published with the corpus and removed all pairs containing sentences with five tokens or less. This resulted in 1,743 sentence pairs, marked as positive data for a classifier. The resulting 51,743 sentence pairs are scored in the same way we score the parallel sentence candidates (i.e. with LaBSE and WAScore), and used to train the logistic regression classifier.

6.2.2 Mining Approach

FaDA, the CLIR-based bilingual document alignment tool that we used in the first step of the mining process, considers each sentence as a separate document and starts by indexing both the source-language and target-language documents. It then constructs a query that selects important words in the document (which is a sentence in our case) based on occurrence count as well as frequency relative to frequency in all documents. The query terms are then translated, in our case using our English–Icelandic lexicon (see Section 3.6), and the system searches the translated query terms in the target-language index, returning the top-10 target-language candidates for each source-language sentence. This process is then repeated in the other direction. For the CLIR tool we used to obtain parallel pair candidates, a bilingual lexicon with lexical translation probabilities is needed.

When mining comparable corpora, it can be hard to distinguish between true alignments and partial alignments. If we try to assign similarity scores between two sentences that are semantically close, most scoring systems will not detect nuances due to a few extraneous words on either side of the aligned sentences that do not have an equivalent in the other sentence. In order to try to help with this problem, we used two different types of scores, LaBSE and WAScore. As shown in Section 4.2.3, LaBSE can score an English–Icelandic sentence pair with a good correlation between the score and the likelihood of the pair being semantically equivalent. By using word alignments and WAScore, we have a mechanism which lowers the confidence for misaligned pairs, which contain extraneous words that can not be aligned (see Section 3.2). These scores were used as an input for a logistic regression

CompNews				
Set	Size	Precision	Recall	F_1
Intersection	$135k$	0.95	0.80	0.87
Union	$1860k$	0.92	0.86	0.86

Table 6.1: Precision, Recall, F_1-measure and number of extracted sentences for a union and intersection of the *FaDA* output.

CompWiki			
Set	Precision	Recall	F_1
Parallel	0.39	0.90	0.54
+partially	0.84	0.33	0.47

Table 6.2: Precision, Recall and F_1-measure as measured when only looking at the sentence pairs marked as parallel in the test data, and when the partially parallel have been added to the desired output.

classifier which determines whether a sentence is parallel or not. The classifier was trained on our CompTrain dataset, described in Section 6.2.1.

6.2.3 News Data

We started by experimenting on the CompNews dataset, with the simple goal of extracting as many parallel sentence pairs as could be found from the two lists of 100,000 sentences in English and Icelandic. After running *FaDA*, we obtained 10 candidates for each of the 100,000 sentences in each language. We created two different candidate sets, one by taking an intersection of both directions, en→is and is→en, and the other by taking a union of the two directions.

As shown in Table 6.1, the intersection returned 135k candidate pairs while the union set contained a total of 1,860k pairs. We calculated LaBSE scores and WAScore for each of the candidate pairs and applied a logistic regression classifier trained on CompTrain to the scores. After applying our classifier, we ended up with 2,034 sentence pairs, of which 1,871 were part of the 2,000 valid sentence pairs in the evaluation dataset. Using the intersection data, we ran our classifier on 93% fewer sentence pairs, while still obtaining almost as many of the valid sentence pairs (1,693) from the 1,782 pairs accepted by our classifier. While the F_1-scores for both approaches were similar, using the union data set we obtained higher recall but using the intersection data set gave better precision.

6.2.4 Extracting Sentence Pairs from Wikipedia Articles

In the same fashion as before, we evaluated our method on Wikipedia data using our CompWiki evaluation set, this time only working with an intersection of the two translation directions. The set contains 10,098 sentences, of which our classifier deemed 200 sentence pairs to be parallel. 77 of these are annotated as parallel in the manually curated test set and 90 as partially parallel. This means we correctly identified all but 9 of the parallel sentence pairs while also extracting 90 out of 421 partially parallel ones. Table 6.2 shows precision, recall and F_1 scores for the experiment, both for valid parallel sentence pairs only, and for sentence pairs either parallel or partially parallel.

6.2.5 Downstream MT Task

After evaluating the accuracy of our approach for extracting parallel sentence pairs from Wikipedia, we proceeded to collect all parallel sentence pairs we could identify in 35,690 article pairs on the English and Icelandic Wikipedia. The collection contained 412,442 Icelandic sentences and 4,259,150 English sentences. Our setup, using *FaDA* and our classifier,

Wikipedia Training					
Training Data	Supplemental Sentences	TestEEA	TestEMA	TestOS	Combined
ParIce50k	0	9.0	9.0	1.6	8.1
ParIce50k+WikiMatrix	313,875	5.6	5.2	2.3	5.1
ParIce50k+Our approach	55,744	13.9	15.9	7.0	13.7

Table 6.3: BLEU scores for MT systems trained on parallel data and sentences extracted from comparable corpora.

yielded 55,744 sentence pairs that were classified as parallel sentences using an intersection of both translation directions.

There have been previous efforts in extracting parallel sentences from Wikipedia. One of the largest such efforts is the WikiMatrix project that mined parallel sentences in 1,620 language pairs. We compared the en–is language pair in WikiMatrix to the output of our system. The first obvious difference is that the WikiMatrix dataset has a lot more data, almost 314,000 sentence pairs compared to our 55,744. To investigate the usefulness of the datasets, we trained a baseline NMT system, a Transformer$_{\mathrm{BASE}}$ model, on 50,000 sentence pairs randomly sampled from the ParIce corpus. We compared it to systems where WikiMatrix was added as supplemental data, and to a system where the results of our approach was used to supplement the ParIce data, using the same hyperparameters.

We compare BLEU scores for the different setups on a combination of three test sets (Barkarson and Steingrímsson, 2020), as well as on each of the test sets individually: TestEEA – containing sentence pairs from EEA regulatory documents; TestEMA – containing sentence pairs from medicine descriptions distributed by EMA; and TestOS – containing sentence pairs from OpenSubtitles. TestEEA and TestEMA, extracted from rather specialized texts, generally have long sentences, while TestOS, from a rather open domain, tends to have shorter sentences. The test sets are used as filtered by Jónsson et al. (2020). All the sentence pairs in the test sets have been manually checked for correctness.

In Steingrímsson et al. (2021b), our paper on mining comparable corpora, we note that our classifier accepts some sentence pairs even though they have a very low WAScore. In order to investigate the effect of using WAScore as a threshold, we train a number of NMT models where we remove sentence pairs under the threshold score. We find that for this data, setting a low threshold for WAScore helps us remove sentence pairs detrimental for training, without losing too many beneficial sentence pairs. In the experiment, this raises our combined BLEU score by approximately one point, while using only 34k supplemental parallel pairs for training instead of 56k, as shown in Figure 6.1.

6.3 Re-Evaluating Data That Would Potentially Be Discarded

When parallel corpora are preprocessed for MT training, a part of the data is commonly discarded. This can be due to sentence length that is incompatible with the model training settings, bad alignments, sub-standard translations, or some other cause of faulty data, as outlined in Chapter 4. For language pairs with limited resources, this can be costly, as in such cases modest amounts of acceptable data may be useful to increase output quality. We carried out two experiments where we extract useful parallel sentences from discarded data. In the

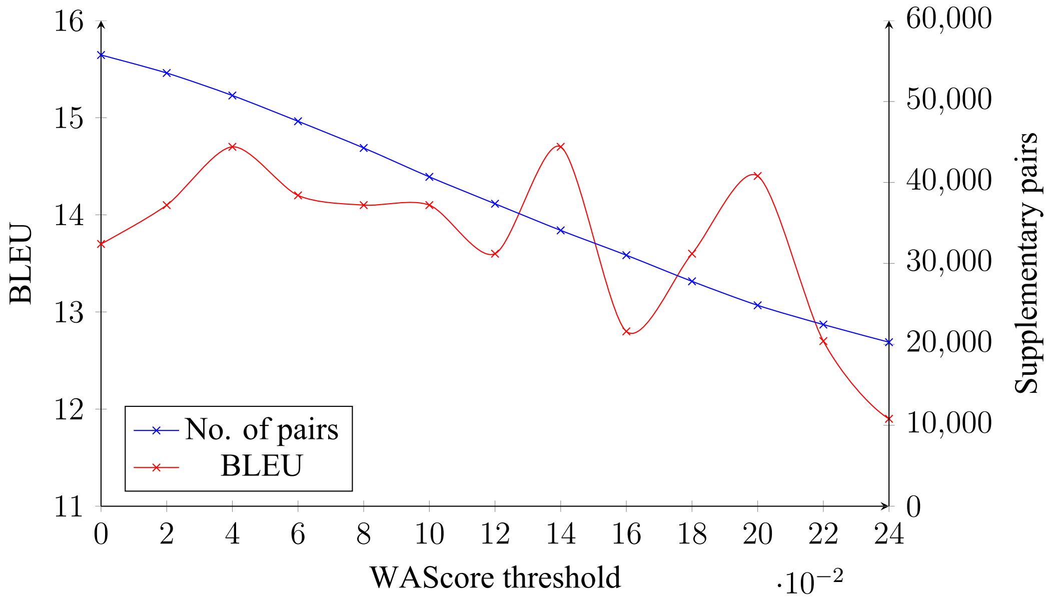


Figure 6.1: BLEU score of the combined evaluation sets for NMT models trained on 50,000 sentence pairs from ParIce as well as supplementary sentence pairs mined from Wikipedia, with different WAScore thresholds.

first experiment, we work with an English–Bengali parallel corpus, split up the sentences in discarded pairs into a few segments and try to find parallel sub-sentences.[10] Our second experiment with discarded data will be described in Section 6.4.

6.3.1 Extracting Sub-sentences from Discarded Parallel Pairs

First, we work with the English–Bengali parallel corpus from the recently released Samanantar data set (Ramesh et al., 2022). This is a publicly available parallel corpora collection for 11 Indic languages. The English–Bengali training data contains 8.52 million sentence pairs. When inspecting random samples from the dataset, we found that not all the sentences pairs are mutual translations, although many contain parallel sub-sentences that can be useful to acquire translation knowledge.

We begin by calculating LASER, LaBSE and WAScore for each sentence in the corpus, and, in the same fashion as we carried out in Section 6.2, use a logistic regression classifier that considers the scores to decide which sentence pairs to filter out. We then order the remaining sentence pairs in descending order based on LaBSE similarity score and create differently sized sets of parallel sentence pairs, with one set containing the 500,000 highest-scoring pairs, another containing the 1,000,000 highest-scoring pairs, and so on as shown in Table 6.4. Note that the S_1 data set represents all the 5.6 million sentence pairs that our classifier deemed acceptable. The other sets contain a subset of the sentence pairs from S_1, based on the order of similarity score.

6.3.2 Model Training and Evaluation

We use all of the differently sized data sets in Table 6.4 to train Transformer$_{\text{BASE}}$ models, as described in Chapters 4 and 5, and evaluate them separately. We train each model on a single

[10] The work in this section was joint work with Pintu Lohar. Pintu inspected the English–Bengali corpus and found that it contains some problematic sentence pairs, as well as splitting up the data.

Dataset	**Size**	**en→bn**		**bn→en**	
	(#sentence pairs $\times 10^6$**)**	**BLEU**	time	**BLEU**	time
Samanantar	8.52	18.1	29h27m	27.9	20h2m
S_1	5.6	19.0	14h33m	27.8	19h5m
S_2	5	19.1	15h43m	***28.5***	11h22m
S_3	4	18.9	16h32m	27.2	9h8m
S_4	3	19.5	7h32m	26.6	6h38m
S_5	2	18.7	5h57m	25.6	5h37m
S_6	1	17.3	1h29m	23.3	1h43m
S_7	0.5	14.9	1h6m	19.9	37m
Final	**3.84 (2+fragments)**	***19.7***	10h52m	26.8	10h32m

Table 6.4: BLEU scores, evaluated on the FLORES evaluation set, for models trained on different sets of selected data until convergence. Scores in bold and italics are highest and significantly higher than other scores.

A100 GPU and use early stopping with the patience set to 5 epochs, the same approach as Ramesh et al. (2022) when they train Transformer$_{\mathrm{BASE}}$ models to compare against their large model. We evaluate the models using BLEU scores calculated on the FLORES evaluation set (Goyal et al., 2022). We use SacreBLEU (Post, 2018) following the process carried out by Ramesh et al. (2022). For Bengali–English, we use the default mteval-v12a tokenizer, but since the SacreBLEU tokenizer does not support Bengali we first tokenize using the IndicNLP[11] tokenizer before running SacreBLEU. SacreBLEU signatures for en→bn[12] and bn→en[13] are provided in footnotes.

6.3.3 Baseline System

We trained models for both translation directions on the full Samanantar dataset of 8.52 million sentence pairs and set that as a baseline for our experiment. The models achieved 18.1 and 27.9 BLEU for en→bn and bn→en, respectively (see Table 6.4), which is somewhat below the scores of 20.3 and 32.2 reported for IndicTrans (Ramesh et al., 2022), trained on the same data. This is most likely due to the model size. We train Transformer$_{\mathrm{BASE}}$ models with $\approx 60,000,000$ parameters, while IndicTrans is a very large transformer model with $\approx 400,000,000$ parameters.

When we evaluate and compare the models trained on different amounts of data, where the smallest datasets have only the highest-scoring sentence pairs in terms of the similarity score used, we find that the BLEU score rises when sentence pairs are added, but only up to a point, when it starts going down again. We speculate that this results from the data becoming more and more noisy, eventually hindering performance. These turning points are different for each language direction, which could be for a number of reasons. For example, the noise might be more prevalent in one language than the other or generating text in one language may need more data than in the other due to complex morphology or other systemic factors.

In our experiment, the turning point is lower for the en→bn dataset, with the highest BLEU for a subset of 3,000,000 sentence pairs. As we do not know whether a more fine-grained turning point would be below or above the 3,000,000 sentence pair mark, we use the 2,000,000 highest-scoring sentence pairs for our final system and process further the other

[11] `https://github.com/AI4Bharat/indicnlp_catalog`

[12] BLEU+numrefs.1+case.mixed+tok.none+smooth.exp +version.2.2.0

[13] BLEU+numrefs.1+case.mixed+tok.13a+smooth.exp +version.2.2.0

Type of selection/discarding	**#sentence/segment pairs**
Whole pairs selected	$1.25M$
Whole Bengali and Partial English	$79K$
Whole English and Partial Bengali	$88K$
Both partial	$456K$
Discarded	$1.74M$

Table 6.5: Result of sub-sentential selection

3.6 million sentence pairs, deemed acceptable by the initial classifier. In order to extract from them data likely to be useful, we split up the sentences in both languages using commas and conjunctions as delimiters. In English, we use 'and' and 'or', and 'ও' and 'এবং' in Bengali.

This results in pairs of sentence parts, with more than half the sentences in each language containing only one or two parts, but approximately $200,000$ Bengali sentences and $400,000$ English sentences with five parts or more. From these parts we create new segments by creating all possible combinations of up to six adjoining sentence parts for each language, as well as the original full sentence, on the condition that the combination contains five or more words. We then pair each new segment against all segments in the other language for any given pair. This results in a total of ≈ 115 million segment pairs.

As before, we use LaBSE to estimate semantic similarity for all segment pairs. Feng et al. (2022) use the threshold 0.6 for selecting sentence pairs mined from CommonCrawl, as they find pairs scoring higher than or equal to this threshold likely to be at least partial translations of each other. Partial translations are often an effect of misalignment and, according to Koehn et al. (2018), including them in a training set can be detrimental to the resulting MT quality. Our aim is to reduce the number of partial translations in our training set and extract from them better mutual translations. Thus, we decide to set our threshold even higher, to 0.75. Furthermore, we proceed to find the best segment pair created from each sentence pair, and only include that one in our training set, so that a given segment pair cannot produce more than one new segment pair. The resulting pairs can be the original sentence pair or a sub-sentence from one or both sides. Out of the 3.6 million sentence pairs, more than 1.7 million were discarded, over 1.2 million sentence pairs were accepted unchanged, and the remaining sentences were accepted when a part had been removed from either one or both languages, as listed in Table 6.5. Using this approach, we produce 1.84 million segment pairs which we add to our foundation training set of $2,000,000$ sentence pairs. We then use this combined data to train a new translation model to investigate whether this processing approach affects the quality of translations, as measured by BLEU.

6.3.4 Evaluation

In order to evaluate if our methodology works to increase translation quality of an NMT system, we train a new model using the same hyperparameters as before and calculate BLEU scores. Table 6.4 shows how using our method gives us the highest BLEU score for en→bn, which is the translation direction we used to decide what data we should process for sub-sentence selection. This indicates that the added segment pairs add more value than if the same number of unchanged sentence pairs would have been added to the training data. By processing the dataset using our methodology, we reduce the training time by 65% while raising the BLEU score by 1.6, from 18.1 to 19.7. A statistical significance test shows that our improved system, trained on less data, is significantly better than the baseline, training

	English	**Icelandic**
Without alignments	$482,975$	$563,381$
Discarded in filtering	$350,964$	$364,267$
1. Total discarded	$833,939$	$927,648$
2. Min. three words + Deduplication	$234,835$	$242,456$
3. After sentence splits	$2,793,254$	$2,279,111$

Table 6.6: Number of discarded sentences used in the experiment and the resulting number of sentence segments, which are candidates for new alignments.

on all of Samanantar, with $p < 0.01$. It is also noteworthy that our system is only 0.6 BLEU below that of IndicTrans, reported in Section 6.3.3, which is almost seven times larger in terms of parameters and trained on the whole Samanantar dataset. We also tested for statistical significance between our system and IndicTrans and found that there is no statistically significant difference between the systems for this translation direction.

Our experimental result shows that the extracted segment pairs can contribute to improving the BLEU score when added as additional data set for training. This has the added benefit of faster convergence, in our case reducing training time by 65%. While our segmentation approach is simple, we show that parallel sub-sentences are useful to acquire translation knowledge and extracting them can lead to significant improvement in performance.

6.4 Discarded Data

In our final experiment using comparable corpora mining methods, we further process the English–Icelandic ParIce data, aligned in Section 5.3.3. We use the highest-scoring approach in terms of BLEU score, where SentAlign was used to align after anchoring the dataset using an ensemble of aligners. The training data resulting from that experiment will be used as a baseline in this experiment. We then use a combination of the approaches in the two previous sections to extract parallel sentences from the data that was discarded in the alignment and filtering process. We add these (discarded) sentence pairs to the training data sets and train new models to investigate if they improve the accuracy of the models in terms of BLEU score.

6.4.1 Data Selection and Pre-processing

Using the highest-scoring alignment approach, we obtain 903,692 sentence pairs from ParIce to train our MT system. We collect all unique sentences that were discarded somewhere in the process, either by not obtaining an alignment by the sentence alignment algorithm or if it was a part of a pair not accepted by our filters. In total, we have over 833,000 discarded sentences in English and over 927,000 in Icelandic, as shown in Table 6.6. After deduplication, and removing all sentences that have less than three tokens that only contain alphabetical characters, we are left with approximately 235,000 and 242,000 sentences for English and Icelandic, respectively.

Next, we split all sentences into segments as we did with the English–Bengali data in Section 6.3. As before, we used conjunctions, ‘and’ and ‘or’ for English and ‘og’ and ‘eða’ for Icelandic. Additionally, we used punctuation for splitting, the same symbols for both languages: .,;:?!()-”’|. We combined the segments into larger sentence parts and created all possible combinations of adjoining segments, from single segments up to recreating the orig-

Processing Step	**No. Pairs left**
FaDA	$2,777,429$
Acceptable Overlap	$1,878,202$
LaBSE minimum	$542,344$
Remove identical	$542,240$
Logistic regression filter	$342,066$
Multiple translations removed	$91,249$
Subsentence removal	$55,371$
Language filter	$36,200$

Table 6.7: Sentence pairs remaining after each step of processing pairs mined from the discarded data.

inal sentence, provided the combinations had a minimum length of three words, maximum length of 120 words, and that 70% of the tokens only contained alphabetical letters. This resulted in 2,793,254 unique Icelandic sentences and sentence parts and 2,279,111 English ones.

6.4.2 Mining for Sentence Pairs in the Discarded Data

As in Section 6.2 we run FaDA to create sentence pair candidates. We use our English–Icelandic lexicon (see Section 3.6) and generate 10 candidates for each Icelandic and English sentence. We then take an intersection of the two generated sets and work further only with sentence pairs suggested for both directions. This results in 2,777,429 pairs to be inspected further, applying the following cleaning steps:

- We remove all sentence pairs with major overlap, in which more than 60% of the tokens in either language are also present in the other.
- We calculate LaBSE score for all pairs and discard pairs that have a lower score than 0.3, because they have a very low chance of being correct (as we found out in Section 4.2.3).
- If two sentence pairs are identical, apart from symbols and numbers, we select the one having the higher LaBSE score.
- We calculate LASER, NMTScore and WAScore for the sentences and classify them using our logistic regression classifier.
- We check if there is more than one pair for each English or Icelandic sentence. If so, only the highest-scoring pair in terms of LaBSE is selected.
- For each sentence pair A, we check for other sentence pairs where the sentences are subsentences of A, such that the subsentence is between 67% and 100% of the length of the original one. If we find another sentence pair, B, having an Icelandic sentence B_{is} that is a substring of A_{is} and an English sentence B_{en} which is a substring of A_{en}, we select the pair that has a higher LaBSE score and discard the other one. This way, we remove nearly identical sentence pairs originating from the same sentences.
- Finally, we run our pairs through a *fasttext* language filter, the same one we used in Chapter 4.

Dataset	**en→is** **BLEU**	**is→en** **BLEU**
903,692 pairs (no discarded data)	43.4	54.0
939,892 pairs (including discarded data)	***43.9***	**54.3**

Table 6.8: Best BLEU scores for models trained with and without the sentence pairs mined from discarded data. The score in bold, representing the training set for is→en including the discarded data, are higher but not significantly higher (p>0.05) than the score obtained without the discarded data in the training set. The score in bold and italic, representing the training set for en→is including the discarded data, are significantly higher than the score obtained without the discarded data in the training set.

In Table 6.7, we show the number of remaining sentence pairs after each processing step. After the final step, only 36,200 sentence pairs remain. We add these pairs to the training data previously acquired by sentence alignment and filtering, resulting in a total of 939,892 sentence pairs. We train Transformer$_{\text{BASE}}$ models using the same settings as before, with patience set to 10 epochs, and calculate BLEU scores for the system with best loss on the in-domain EEA development set from the ParIce 21.10 dev/test splits (Barkarson et al., 2021), compiled from held-out documents from the same source as the ParIce corpus. We compare the results to the systems trained without the supplemental sentence pairs mined from discarded data.

Table 6.8 shows the results of our experiment. We obtain higher scores for both translation directions, but only the en→is translation is significantly higher (p<0.05) when the training data is supplemented with the sentence pairs mined from the discarded data. With regards to the low number of sentence pairs retrieved using this approach the results should not be surprising, but it is still an indicator of this approach being able to give some additional benefits to a training set for MT.

6.5 Conclusions

Our experiments have shown that our method, combining cross-lingual information extraction, contextualized embeddings-based scoring, and a classifier based on multiple different scoring mechanism, is efficient at finding parallel segments in comparable corpora. Our experiments reveal that we can expect a part of the mined pairs to be partially parallel, and that by splitting the sentences up and investigating which parts of the sentences are most appropriate for pairing with other sentences or sentence parts, we can improve the quality of our parallel corpora, leading to better quality MT models trained on the data.

Regarding our research question, our experiments indicate that there is a potential in taking a second look at data that would usually be discarded. Such data can be considered to be comparable corpora and treated as such for mining parallel sentence pairs. While it does not result in a very large number of sentence pairs in our experiments, it does have a positive effect on a downstream MT task.

In the next chapter, we will conclude by training MT models on the combined data produced by our experiments, and comparing them to models trained on only the ParIce corpus. We will compare the models using automatic metrics on the WMT evaluation sets, as well as having professional translators and linguists carry out manual evaluation using two evaluation methodologies.

Chapter 7

Putting It All Together: Evaluation

In Chapter 6, we mined comparable corpora for parallel sentence pairs, refined a parallel corpus by removing extraneous data from partially parallel sentence pairs, and took a second look at data discarded during the compilation phase of a parallel corpus. In previous chapters, we have developed approaches to improve alignment and filtering of parallel data. We evaluated MT output by computing BLEU scores using different evaluation sets. For general comparison, we have used the evaluation data set for English–Icelandic from the WMT21 news translation task. We have also used our own evaluation sets, which we created by sampling and manually revising sentence pairs from ParIce subcorpora: EEA texts, Open Subtitles, ESO, EMA and news from the Nordic Council of Minsters. We trained our models using the Transformer$_{\text{BASE}}$ architecture.

In this chapter, we will put all of our approaches together, combining training data compiled using the highest-scoring methods, and recruit translation professionals and linguists to manually evaluate the output of MT systems trained on that data. The purpose of the manual evaluation is to investigate whether our methods for compiling training data have a measurable effect on the output of MT systems, as perceived by humans.

While BLEU is an easy to use indicator of translation quality and can be useful for diagnosing whether MT systems improve or deteriorate when hyperparameters are changed in training, or when training data changes, the evaluation metric has been criticized for having low correlation with human judgements for most of the time it has been in use, see e.g. Callison-Burch et al. (2006) and Freitag et al. (2022). Nonetheless, it is still used in most MT research and is thus a convenient, although somewhat flawed, way of comparing different MT systems (using the same tokenization). Indeed, a wide range of automatic metrics have been developed. We experiment with a recent one, COMET-22, which was shown to have a high correlation with human judgment in the WMT22 metrics shared task (Freitag et al., 2022).

We evaluate output from MT systems, trained on differently compiled datasets for different translation directions using both a classic fluency evaluation approach and Multidimensional Quality Metrics (MQM), a framework created as part of the EU QTLaunchPad and QT21[1] projects.

We aim to investigate whether the approaches we have developed for preparing MT training data are useful, and that they produce data that leads to better MT output as perceived by human users. We will compare systems trained on ParIce version 21.10 to a version of ParIce aligned and filtered using our best approaches, as well as a dataset composed of all

[1] `www.qt21.eu`

the datasets we have created in the preceding chapters. We use two model architectures: Transformer$_{\text{BASE}}$ as before, and mBART (Liu et al., 2020b), fine-tuned on our datasets.

In Section 7.1, we give an overview of relevant related work and Section 7.2 provides a description of our final models. In Section 7.3, we evaluate the systems' output automatically using the two previously mentioned metrics. Section 7.4 describes the human evaluation and its results, and Section 7.5 concludes the chapter with discussions about the results and their significance.

7.1 Related Work

Ever since the first MT systems were developed, there have been efforts to evaluating their quality in terms of how well they convey the intent of the original text, fluency in the target language, style, tone, consistency and other factors. Efforts were made to standardize the measurement procedure in the ALPAC report (ALPAC, 1966), assessing "intelligibility" (how natural the text reads) and "fidelity" (how precisely the translation comprehends the meaning intended) on nine-point scales. In the early 1990s, methodologies were developed within an ARPA-sponsored MT research program (White et al., 1994), suggesting three measures for evaluating MT: Adequacy (can the information in a professional translation be found in the MT output), fluency (does the translation read like good English) and comprehension (the degree to which a translated text can be understood).

For the first two WMT shared translation tasks, MT output was evaluated in terms of adequacy and fluency, rated on five-point scales (Koehn and Monz, 2006). Vilar et al. (2007) pointed out drawbacks to these measures, arguing that they were subjective, evaluators could be biased, and the results were not reproducible. They suggested ranking MT systems instead. Ranking-based approaches became the official WMT metrics in 2008 (Callison-Burch et al., 2008) and remained so until 2016. Graham et al. (2017) argue that as consistency levels are low for these methods and researchers who assess the systems have been shown to slightly favour their own, other methods of human evaluations are needed. Graham et al. (2013) had previously suggested using a continuous scale in the range of 0–100, allowing for scores to be standardized to eliminate individual judge preferences, resulting in higher inter-annotator agreement. Callison-Burch et al. (2007) found that adequacy and fluency evaluations are highly correlated. In light of this, Bojar et al. (2016) explored evaluating MT by asking annotators to provide an assessment of the direct quality of a system's output relative to a reference translation. This is called direct assessment (DA) and has been employed for evaluations at WMT since 2017.

Some of the shortcomings identified for DA include that scores have been found to correlate poorly with more fine-grained MT evaluation and to exhibit weaker preference for human translations compared to machine output (Freitag et al., 2021b). Moreover, non-professional crowd-sourced workers are typically used for DA, requiring robust vetting. Freitag et al. (2022) point out that 63% of annotations for WMT22 were removed due to failing quality control, and Bentivogli et al. (2018) find that they exhibit a reference-bias, with evaluators scoring correct translations lower if they deviate from the reference text.

Way (2018) argues that there is no single 'gold standard' measure of quality for MT, and that it needs to be evaluated in the context of the use-case for which it is intended. He emphasises that human evaluation of MT output is crucial if system developers are to improve their systems. Läubli et al. (2020) and Toral et al. (2018) make identical recommendations for best practices when evaluating MT, based on empirical evidence from their research. These include using original source texts and not source texts that are translations themselves, evaluating whole documents and not just sentences, evaluating both fluency and adequacy, and choosing professional translators as raters.

A wide range of other different approaches have been suggested. Scarton and Specia (2016) propose using a corpus of reading comprehension tests, evaluating machine-translated documents based on answers to questions by fluent speaker of the target language. Forcada et al. (2018) suggest gap-filling as a cheaper alternative to reading comprehension approaches when evaluating the usefulness of MT for gisting.

The MQM framework was developed to be flexible and suitable for evaluating any sort of translated text, human or machine translated (Lommel et al., 2014). It is not a one-size-fits-all metric, but rather a model for declaring multiple metrics, a fine-grained approach to quality evaluation allowing the results for error types to be compared. It is language neutral and therefore applicable to any language pair.

7.2 The Final MT Models

In order to test the feasibility of our approaches, we train models using different datasets, described in Table 7.1. They are: 1) ParIce 21.10; 2) a refined version of ParIce, realigned and filtered using our highest-scoring methods for each translation direction, as well as parallel sentence pairs mined from the discarded data; and 3) a training set containing all the data we processed, ParIce using the alignment approaches from Chapter 5, filtering from Chapter 4 and discarded data refined in Chapter 6, ParaCrawl using the filtering approaches from Chapter 4 and sentence pairs from Wikipedia collected using the comparable corpora mining approaches described in Chapter 6.

We train our final models using two different architectures: Transformer$_{\text{BASE}}$ models (as in Section 3.5), and mBART25 fine-tuned on our datasets. mBART25 is an LLM ($\approx 610,000,000$ parameters) pre-trained on 25 languages, including English but not Icelandic. The model has been successfully adapted to translating between English and Icelandic (Símonarson et al., 2021). We use fairseq (Ott et al., 2019) to train the models. The models are fine-tuned on a desktop computer running an i5-12600K processor, with 64 GB of memory and a GeForce RTX 3090 GPU. We train for 500,000 updates, taking approximately 50 hours for each model. We use the same hyperparameters as Liu et al. (2020b), except that we do 10,000 warm-up steps. We validate the models every 25,000 steps in terms of BLEU score

Dataset	**Description**	**en→is #pairs**	**is→en #pairs**
ParIce 21.10	ParIce parallel corpus, version 21.10	1,864,679	1,864,679
ParIce refined	Realigned and refiltered ParIce corpus + sentence pairs extracted from the discarded data.	1,495,524	1,505,706
All data	ParIce refined + ParaCrawl + Wikipedia sentence pairs	2,277,023	2,754,596

Table 7.1: Description of data sets used for training the final models. We realigned and refiltered the whole ParIce corpus using the highest-scoring filtering approaches for each translation direction, as described in Chapter 4, and the highest-scoring alignment approaches, as described in Chapter 5. We then mined for more parallel pairs in the discarded data, as described in Section 6.3. For the largest training set, we added ParaCrawl as filtered for each translation pair in Chapter 4 and the parallel Wikipedia sentence pairs mined in Section 6.2.4.

		en→is	is→en
mBART	ParIce 21.10	20.0	25.9
	ParIce refined	20.8	27.9
	All data	**23.0**	**34.1**
Transformer$_{\text{BASE}}$	ParIce 21.10	19.2	25.7
	ParIce refined	19.6	26.3
	All data	**23.0**	**33.5**

Table 7.2: BLEU scores for the final models. Scores in bold are highest and significantly higher (p<0.05) than other scores in the same group.

on the development set from the en–is shared task at WMT21, and finally select the best checkpoint based on validation BLEU. We do this for each dataset and translation direction and evaluate the best models on the WMT21 en–is evaluation set. For one of the models, the one fine-tuned on the largest dataset containing all of our processed data and translating is→en, the best checkpoint was the last one, indicating that the model could possibly improve even further with more updates. For other models, the highest-scoring checkpoints were from 100,000 (ParIce 21.10: is→en) to 400,000 (All data: en→is).

7.3 Automatic Evaluation

In the preceding chapters, we have evaluated MT output by calculating BLEU scores on different evaluation data sets. For general comparison, we have used the evaluation data set for English-–Icelandic from the WMT21 news translation shared task. We also used that for evaluating our final systems in this chapter. Furthermore, we also evaluated the output using COMET-22 (Rei et al., 2022), a COMET model that has been shown to have a high correlation with human evaluation (Freitag et al., 2022). COMET (Rei et al., 2020) is a neural framework for MT evaluation. The framework builds on pre-trained cross-lingual language models such as multilingual BERT (Devlin et al., 2019) or XLM-RoBERTa (Conneau et al., 2020). The framework is based on an estimator model and a translation ranking model. The estimator model is trained on manually evaluated sentence pairs to minimize the mean square error between predicted scores and quality assessments (such as DA or MQM). The translation ranking model receives the source sentence, hypothesis and a reference, obtains sentence embeddings from a pre-trained LM and uses the harmonic mean between the distance from the hypothesis to the source and from the hypothesis to the reference to calculate a similarity score.

We compared our systems and checked if there was statistical significance between our model scores, using paired bootstrap resampling for BLEU and paired T-tests for COMET. Table 7.2 gives the BLEU scores for all our models. For both types of models and both translation directions, the models trained on all the data are significantly better than the other models. For all but Transformer$_{\text{BASE}}$ en→is, the models trained on ParIce Refined are also significantly better than the ones trained on ParIce 21.10.

We compared our models to similar models submitted to the WMT21 shared task for translating the English–Icelandic language pair. Allegro.eu (Koszowski et al., 2021) based their models on the Transformer$_{\text{BIG}}$ architecture. They trained their models using 3,900,000 parallel sentence pairs from ParIce, ParaCrawl and WikiMatrix, and 2,900,000 synthetic pairs (back-translations). They obtained a BLEU score of 22.7 for en→is and 33.3 for is→en.

		en→is	is→en
mBART	ParIce 21.10	0.7953	0.7768
	ParIce refined	**0.8000**	0.7858
	All data	*0.7961*	**0.8245**
Transformer$_{\text{BASE}}$	ParIce 21.10	0.7431	0.7540
	ParIce refined	0.7563	0.7638
	All data	**0.7740**	**0.8130**

Table 7.3: COMET-22 scores for our final models. Scores in bold are the highest, but not significantly higher than scores in italic, according to the results of the t-test presented in Table 7.4 (p<0.05).

This is slightly less than even our best Transformer$_{\text{BASE}}$ models, which are trained on considerably fewer sentence pairs (see Table 7.1) and do not take advantage of any back-translated data. Icelandic NLP company Miðeind fine-tune mBART25 for their submission (Símonarson et al., 2021). They use a filtered version of ParIce, as well as additional data from the JW300 corpus (Agić and Vulić, 2019) and a small corpus of theses abstracts (Símonarson and Snæbjarnarson, 2021). Furthermore. they use back-translations, over 30,000,000 sentence pairs for each translation direction. They trained on sixteen 32GB nVidia V100 GPUs for 4 days, reaching BLEU scores of 22.7 for en→is and 32.9 for is→en. This is lower than the scores of our mBART models, trained on thoroughly aligned and filtered data, obtained in 2 days on one GeForce RTX 3090 GPU, which is less than 50% more powerful than one V100 GPU, according to common benchmarks. Símonarson et al. (2021) then used these models to generate new back-translations and use these to continue training, reaching 24.3 en→is and 33.5 for is→en. While their en→is model outperforms ours in terms of BLEU, their is→en model does not. We conjecture that this is due to our data processing methods, making the data better suitable for NMT training.

We have mentioned some of the criticism of BLEU. Recently, neural metrics have been said to be better and more robust and the title of the overview paper for the WMT22 metrics shared task calls for researchers to stop using BLEU (Freitag et al., 2022). COMET-22 has been shown to be one of the metrics having the highest correlation with human evalua-

		Model X / Model Y	Tied	X wins	Y wins	p-value
mBART	en→is	PI 21.10 / PI refined	12.7%	6.3%	81%	**0.0350**
		PI 21.10 / All data	15.7%	34.7%	49.7%	0.7157
		PI refined / All data	14%	73.7%	12.3%	0.1551
	is→en	PI 21.10 / PI refined	1.7%	0%	98.3%	**0.0001**
		PI 21.10 / All data	0%	0%	100%	**0.0000**
		PI refined / All data	0%	0%	100%	**0.0000**
Transformer$_{\text{BASE}}$	en→is	PI 21.10 / PI refined	0.7%	0%	99.3%	**0.0000**
		PI 21.10 / All data	0%	0%	100%	**0.0000**
		PI refined / All data	0%	0%	100%	**0.0000**
	is→en	PI 21.10 / PI refined	2%	0%	98%	**0.0003**
		PI 21.10 / All data	0%	0%	100%	**0.0000**
		PI refined / All data	0%	0%	100%	**0.0000**

Table 7.4: T-test results of COMET-22 scores for our final models. A p-value in bold means that there is significant difference between the systems, with $p < 0.05$.

tion. Before we proceed with the manual evaluation of our models, we compute COMET-22 scores for our models.

We chose to use the COMET-22 model (Rei et al., 2022) because it was one of two models having the highest correlation with human judgements in the WMT22 metrics shared task. The other high correlating metric, MetricX XXL, is not publicly available. Table 7.3 shows the COMET-22 scores for our models. The scores are mostly in line with the BLEU scores. The mBART models score slightly higher than the Transformer$_{\text{BASE}}$ models and the is→en models score higher than the en→is models. The greatest deviation from the BLEU results are for the mBART models trained for en→is, where the score for all models are very close, and the ParIce refined model scores the highest.

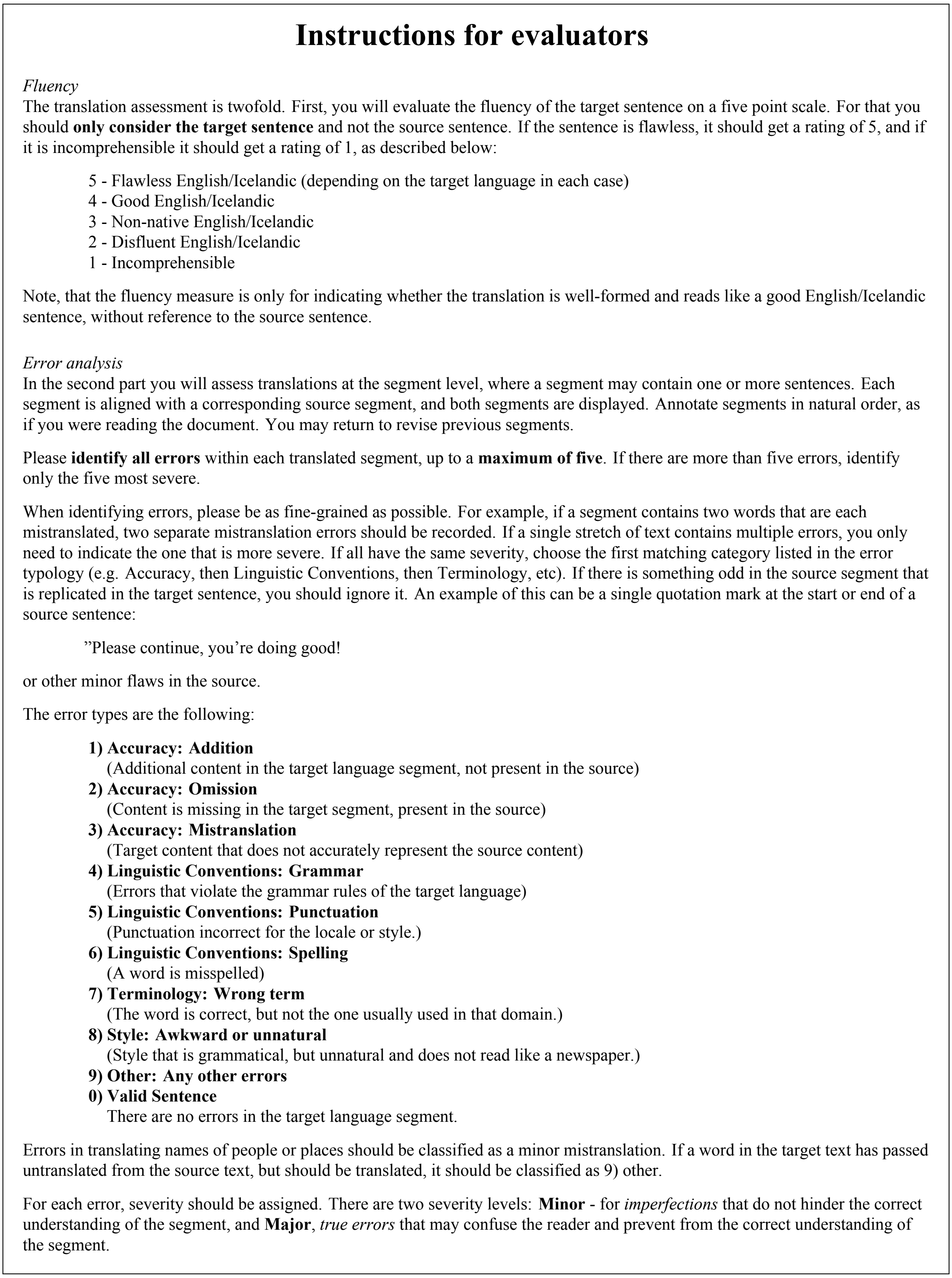
We carried out paired t-tests to test for statistical significance. Table 7.4 gives the results of the paired t-tests. For all model pairs in groups of models/translation directions, there is a statistically significant difference between all model outputs, except for mBART/en→is. There, the ParIce refined mBART model is significantly better than the ParIce 21.10 mBART model, but there is no significant difference between the other two model pairs.

7.4 Manual Evaluation

While there is no consensus on the best approaches to human evaluation of MT output, the best approach may often be task-based and depend on the purpose of the translations (Way, 2018). In our case, we want to investigate whether training MT models on the different datasets, compiled using different approaches, affects the output of the models in such a way that human evaluators perceive the difference. Firstly, we want to see if the translations become more natural when we process the training data with methods we expect to be more appropriate for the dataset and translation direction. Secondly, we want to know if the translations are more accurate, with fewer mistranslation or other kinds of effects detrimental to correctly representing the meaning of the source sentence in the target language.

Our evaluation was twofold. First, we collected subjective reports of the fluency of the target sentences using the approach employed for the first shared translation tasks at WMT, and described by Koehn and Monz (2006). They evaluate fluency on a five-point scale, shown in the evaluator instructions in Figure 7.1. To evaluate translation accuracy, we opted for using MQM which is very flexible and we can use it to gain an insight into what kind of errors are prevalent in each model. The errors are grouped in four main categories, with each of the main categories having a number of subcategories. For a given evaluation task a subset of the categories can be chosen. Freitag et al. (2021a) argue that a fine-grained evaluation schema like MQM is needed when the difference in quality of the MT systems is narrowing. Although we expected there to be quality differences in our models depending on how the training data was processed, some of the models (ParIce 21.10 and ParIce refined) use training sets extracted from the same parallel corpus. We thus wanted our evaluation approach to be able to discern small differences and opted for using MQM.

We wanted the output of each model to be evaluated by multiple evaluators. We also wanted the coverage to be high enough to be meaningful. However, manual evaluation is time-consuming and we had to recruit volunteers to carry out the task. We had four groups of models that we wanted to compare, two types of architecture and two translation directions, with three models in each group – a total of 12 models to evaluate. As we wanted the evaluations to be carried out in a professional manner, we set a requirement for our evaluators to be either professional translators or linguists educated in an English-speaking country. We



Instructions for evaluators

Fluency

The translation assessment is twofold. First, you will evaluate the fluency of the target sentence on a five point scale. For that you should **only consider the target sentence** and not the source sentence. If the sentence is flawless, it should get a rating of 5, and if it is incomprehensible it should get a rating of 1, as described below:

5 - Flawless English/Icelandic (depending on the target language in each case)
4 - Good English/Icelandic
3 - Non-native English/Icelandic
2 - Disfluent English/Icelandic
1 - Incomprehensible

Note, that the fluency measure is only for indicating whether the translation is well-formed and reads like a good English/Icelandic sentence, without reference to the source sentence.

Error analysis

In the second part you will assess translations at the segment level, where a segment may contain one or more sentences. Each segment is aligned with a corresponding source segment, and both segments are displayed. Annotate segments in natural order, as if you were reading the document. You may return to revise previous segments.

Please **identify all errors** within each translated segment, up to a **maximum of five**. If there are more than five errors, identify only the five most severe.

When identifying errors, please be as fine-grained as possible. For example, if a segment contains two words that are each mistranslated, two separate mistranslation errors should be recorded. If a single stretch of text contains multiple errors, you only need to indicate the one that is more severe. If all have the same severity, choose the first matching category listed in the error typology (e.g. Accuracy, then Linguistic Conventions, then Terminology, etc). If there is something odd in the source segment that is replicated in the target sentence, you should ignore it. An example of this can be a single quotation mark at the start or end of a source sentence:

> ”Please continue, you’re doing good!

or other minor flaws in the source.

The error types are the following:

1) Accuracy: Addition
(Additional content in the target language segment, not present in the source)
2) Accuracy: Omission
(Content is missing in the target segment, present in the source)
3) Accuracy: Mistranslation
(Target content that does not accurately represent the source content)
4) Linguistic Conventions: Grammar
(Errors that violate the grammar rules of the target language)
5) Linguistic Conventions: Punctuation
(Punctuation incorrect for the locale or style.)
6) Linguistic Conventions: Spelling
(A word is misspelled)
7) Terminology: Wrong term
(The word is correct, but not the one usually used in that domain.)
8) Style: Awkward or unnatural
(Style that is grammatical, but unnatural and does not read like a newspaper.)
9) Other: Any other errors
0) Valid Sentence
There are no errors in the target language segment.

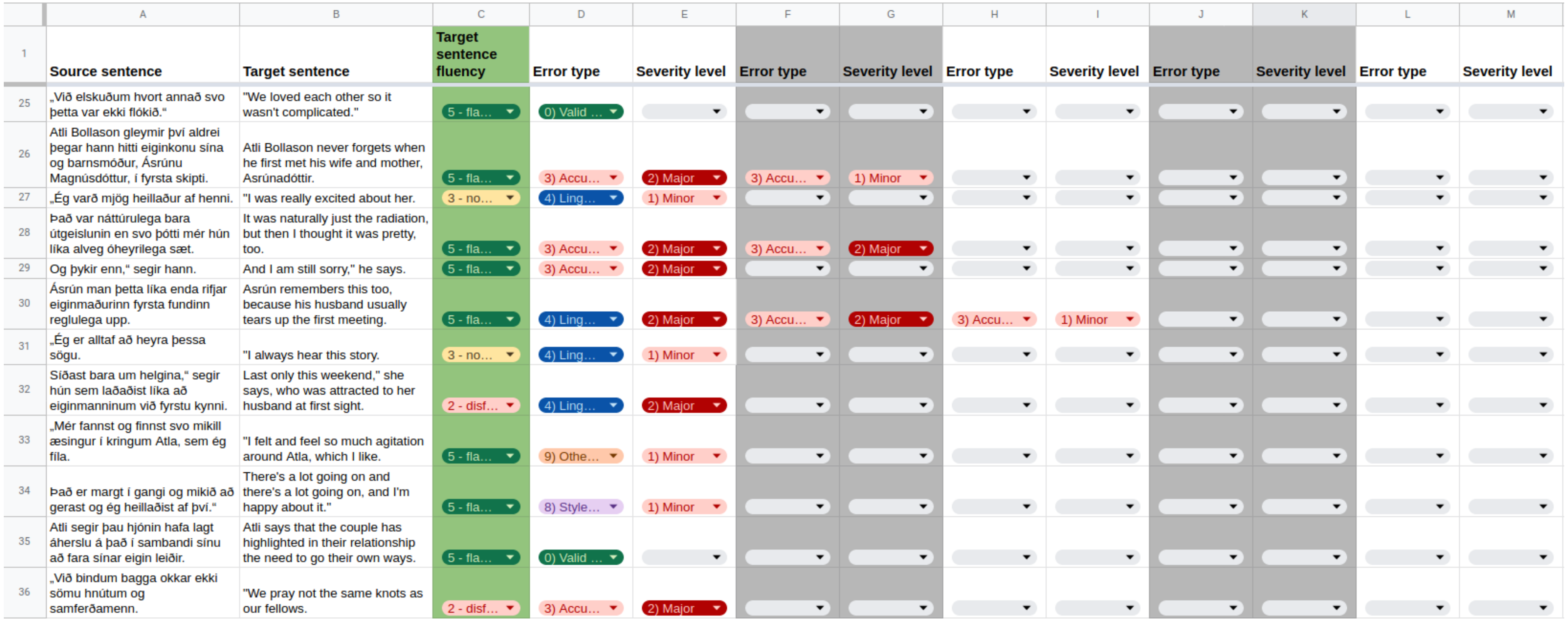
Errors in translating names of people or places should be classified as a minor mistranslation. If a word in the target text has passed untranslated from the source text, but should be translated, it should be classified as 9) other.

For each error, severity should be assigned. There are two severity levels: **Minor** - for *imperfections* that do not hinder the correct understanding of the segment, and **Major**, *true errors* that may confuse the reader and prevent from the correct understanding of the segment.

Figure 7.1: Instructions given to evaluators for the manual evaluation task. They were asked to read this thoroughly before starting and ask for clarification if something was not clear.

managed to attract seven evaluators that meet these requirements.[2] All of the evaluators are native Icelandic speakers, fluent in English.

[2] We would like to thank the evaluators for their hard work. They are: Björn Halldórsson, Einar Freyr Sigurðsson, Finnur Ágúst Ingimundarson, Hilmar Hilmarsson, Þór Tryggvason, Þórdís Úlfarsdóttir and Ölvir Gíslason.



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Source sentence	Target sentence	Target sentence fluency	Error type	Severity level	Error type	Severity level	Error type	Severity level	Error type	Severity level	Error type	Severity level
25	„Við elskuðum hvort annað svo þetta var ekki flókið.“	"We loved each other so it wasn't complicated."	5 - fla…	0) Valid …									
26	Atli Bollason gleymir því aldrei þegar hann hitti eiginkonu sína og barnsmóður, Ásrúnu Magnúsdóttur, í fyrsta skipti.	Atli Bollason never forgets when he first met his wife and mother, Asrúnadóttir.	5 - fla…	3) Accu…	2) Major	3) Accu…	1) Minor						
27	„Ég varð mjög heillaður af henni.	"I was really excited about her.	3 - no…	4) Ling…	1) Minor								
28	Það var náttúrulega bara útgeislunin en svo þótti mér hún líka alveg óheyrilega sæt.	It was naturally just the radiation, but then I thought it was pretty, too.	5 - fla…	3) Accu…	2) Major	3) Accu…	2) Major						
29	Og þykir enn,“ segir hann.	And I am still sorry," he says.	5 - fla…	3) Accu…	2) Major								
30	Ásrún man þetta líka enda rifjar eiginmaðurinn fyrsta fundinn reglulega upp.	Asrún remembers this too, because his husband usually tears up the first meeting.	5 - fla…	4) Ling…	2) Major	3) Accu…	2) Major	3) Accu…	1) Minor				
31	„Ég er alltaf að heyra þessa sögu.	"I always hear this story.	3 - no…	4) Ling…	1) Minor								
32	Síðast bara um helgina,“ segir hún sem laðaðist líka að eiginmanninum við fyrstu kynni.	Last only this weekend," she says, who was attracted to her husband at first sight.	2 - disf…	4) Ling…	2) Major								
33	„Mér fannst og finnst svo mikill æsingur í kringum Atla, sem ég fíla.	"I felt and feel so much agitation around Atla, which I like.	5 - fla…	9) Othe…	1) Minor								
34	Það er margt í gangi og mikið að gerast og ég heillaðist af því.“	There's a lot going on and there's a lot going on, and I'm happy about it."	5 - fla…	8) Style…	1) Minor								
35	Atli segir þau hjónin hafa lagt áherslu á það í sambandi sínu að fara sínar eigin leiðir.	Atli says that the couple has highlighted in their relationship the need to go their own ways.	5 - fla…	0) Valid …									
36	„Við bindum bagga okkar ekki sömu hnútum og samferðamenn.	"We pray not the same knots as our fellows.	2 - disf…	3) Accu…	2) Major								

Figure 7.2: The Google Sheet used by the human annotators for evaluation.

We decided to evaluate 300 sentence pairs for each model. The sentences were randomly chosen from the WMT21 English–Icelandic evaluation sets, with original English sentences as source for en→is translations and original Icelandic sentences as source for is→en translations. We split the evaluation sets into batches of 50 sentences in a row, and selected six random batches, but with the same source sentences chosen for each model. By selecting multiple sentences in a row, we obtained multiple translations in a row from the same news articles, giving the evaluators some context for the translated sentences.

We used the same 300 sentence pairs for each model and divided them between four evaluators. Each evaluator assessed sentence pairs for three model types, in both directions, thus evaluating a total of 225 sentence pairs for each direction, 450 sentence pairs in total. The evaluation time varied between the evaluators, from approximately 6 hours for the fastest one, to approximately 10 hours for the one who took the most time to finish the task of evaluating 450 sentence pairs. One of the evaluators evaluated twice as much as the others, 75 sentences for all the models, 900 sentence pairs in total.

We loaded the sentence pairs into a Google Sheets document, one pair in a row. The sentence pairs are followed by drop-down lists from which the evaluator selected one of the correct categories, both for fluency and for the MQM categories. Separate sheets, represented in Figure 7.2, were prepared for each evaluator. Each sentence pair was only evaluated by one evaluator.

When manually evaluating translation output, it is important that the evaluators follow the same procedure when evaluating and have the same understanding of the task. We prepared guidelines, shown in Figure 7.1. The guidelines for MQM were modeled after Freitag et al. (2021a) and following Mariana et al. (2015).

7.4.1 Fluency

Fluency results are given as the average of the raw judgement scores, rated on the scale of 1-5. We also normalize the scores for each evaluator, using Equation (7.1) to obtain standardized scores, and give averaged standardized scores for each model.

$$z = \frac{x - \mu}{\sigma} \tag{7.1}$$

		Model	Average	Average z
mBART	en→is	PI 21.10	3.11	0.026
		PI Refined	3.14	0.065
		All data	2.99	-0.092
	is→en	PI 21.10	3.64	-0.049
		PI Refined	3.60	-0.090
		All data	3.85	0.139
Transformer$_{\text{BASE}}$	en→is	PI 21.10	2.22	-0.174
		PI Refined	2.49	0.106
		All data	2.46	0.068
	is→en	PI 21.10	2.81	-0.221
		PI Refined	3.03	-0.047
		All data	3.43	0.268

Table 7.5: Results of fluency evaluation. The higher the scores, the more fluent the output.

Severity	Category	Weight
Major	All	5
Minor	Linguistic Conventions: Punctuation	0.1
	All others	1

Table 7.6: Weights given to errors for the error categories used in our MQM evaluation.

In the equation, μ is the mean of all fluency scores given by the evaluator and σ is the standard deviation.

The results of the fluency evaluation are given in Table 7.5. The Transformer$_{\text{BASE}}$ is→en model has the clearest results, with clear difference between models trained on ParIce 21.10, ParIce refined and all data, and fluency improving with more processed data and then more data. This is in line with both the BLEU and COMET-22 scores. The evaluations for the other model groups do not reveal such a clear trend. Transformer$_{\text{BASE}}$ en→is shows more fluency when trained on ParIce refined rather than ParIce 21.10, but the model trained on all the data scores slightly less than ParIce refined. The mBART model also shows somewhat different results with the ParIce 21.10 model scoring slightly higher than ParIce Refined for the mBART is→en model and the model trained on all the data scoring lowest for mBART en→is.

7.4.2 MQM

We adapted the MQM framework to our needs and selected the error categories we believed most relevant for our comparison. We asked the evaluators to classify the errors into eight error types, and to mark the error as "other" if it did not fit any of the selected categories. The list of error types is given in Figure 7.1. For each error, we asked the evaluators to assign a severity class, *Minor* for imperfections that do not hinder the correct understanding of the segment, and *Major* for errors that may confuse the reader and prevent the correct understanding of the segment. The error types and severity are used to calculate an MQM score for each segment. We exported our results from the Google Sheets documents and used MQM Viewer[3] to calculate the scores for each model. For calculating the scores, we used the default weights in MQM viewer, given in Table 7.6. These weights were found by Freitag et al. (2021a) to give good stability over two language pairs they evaluated, en–de and en–zh, while also matching system-level rankings from professional translators rating segments on a seven-point Likert-type scale.

Table 7.7 gives the results for the MQM evaluation. For all model groups, except one, the trend is the same as with the BLEU scores. The ParIce 21.10 models have the most errors and the models trained on all the data have the fewest errors. The only exceptions are the Transformer$_{\text{BASE}}$ en→is models, where the ParIce Refined model has a slightly worse score than ParIce 21.10. Upon further inspection, the evaluation of these models also differs in another respect. The evaluations of one of the four evaluators score almost twice as high (worse) as the average for the other three evaluators, with 11.063 per segment. The evaluator giving the lowest scores (best) has an average of 5.249 per segment. Furthermore, the high-score evaluator rates the models so that their rankings are different from the rankings produced by all other evaluators. In other cases, the difference is never that pronounced.

[3] `https://github.com/google/wmt-mqm-human-evaluation`

		Model	MQM	Major	Minor	Acc.	Ling. conv.	Other
mBART	en→is	PI 21.10	7.819	6.850	0.967	5.873	1.553	0.393
		PI Refined	5.991	4.983	1.003	4.437	1.378	0.177
		All data	5.675	4.600	1.070	3.633	1.782	0.260
	is→en	PI 21.10	4.938	4.200	0.737	4.143	0.638	0.157
		PI Refined	4.446	3.667	0.777	3.593	0.652	0.200
		All data	3.094	2.450	0.643	2.137	0.917	0.040
T_{BASE}	en→is	PI 21.10	7.648	6.650	0.997	6.590	0.871	0.187
		PI Refined	7.827	6.900	0.923	6.513	1.163	0.150
		All data	6.973	5.933	1.040	5.760	1.133	0.080
	is→en	PI 21.10	5.110	4.283	0.827	4.397	0.667	0.047
		PI Refined	4.084	3.333	0.750	3.360	0.678	0.047
		All data	2.357	1.757	0.590	2.023	0.314	0.020

Table 7.7: Average MQM score per segment. The lower the score, the fewer and less severe the errors are. The table also gives average scores for each severity class, and for supercategories of the error classes.

This may skew the results somewhat, which is one of the dangers of having few evaluators and few evaluated sentence pairs.

7.4.3 Limitations

We only evaluated 300 sentence pairs for each model and each sentence pair was only evaluated by one evaluator. This means that if one model has just a few more sentences than the next model, that are short, simple and translate well and a few less sentences that are long, complicated and receive poor translations, the score balance can change substantially. Although we have not noticed such issues in the evaluated data, such an effect may still be present. Another bias that a small evaluation can bring about, is that of one evaluator behaving markedly different from the others, as we may be seeing with the $\text{Transformer}_{\text{BASE}}$ en→is models discussed in the previous section. We do not check for consistency or standardize the MQM evaluation scores in any way, and so just one evaluator may skew the results substantially. Furthermore, as each sentence pair is only evaluated by one evaluator, we cannot measure inter-annotator agreement. Thus, we cannot say for sure that one of the evaluators is annotating in a markedly different way from the others. It is a possibility that this deviation is because of real issues in the translations.

One more limitation worth noting is that although all the evaluators are fluent English speakers, have lived in an English-speaking country and/or worked as professional translators translating to and from English, none of them are native English speakers. This may affect their evaluations, particularly the is→en ones.

7.5 Discussion

We have evaluated our final systems using four approaches, two automatic: BLEU and COMET-22, and two manual: fluency and MQM. The BLEU scores all rise when the data is processed using our methods and when we add more processed data to the training set. This was expected as we had already seen these effects in our previous experiments, where we based our decisions on the BLEU score. The other automatic metric we use, COMET-22, shows similar outcomes, although in one case they are not as significant, with no significant difference between using all our data to fine-tune an mBART model for en→is translation or just the refined dataset.

The manual MQM evaluation, on the other hand, shows more differences, especially for the error types in the *accuracy* category. The *linguistic conventions* category, which contains grammar, punctuation and spelling errors, does not show the same results and is more in line with the fluency evaluation, with little marked difference between the models in terms of fluency, but ParIce refined scoring a little better than the others. An inspection of the results shows that the majority of the translations into Icelandic have fluency issues, commonly caused by problems with inflection and other morphological problems. As the manual scores indicate, both for fluency and MQM, this is a slightly smaller problem for the mBART models than for the Transformer$_{\text{BASE}}$ models. Due to these problems, it may also be problematic to discern other errors. We speculate that it may be harder to make proper use of MQM as translation quality is reduced and that, when there are multiple errors in most sentences, the evaluators tend to focus on fewer error groups and select just the most prominent errors.

With the is→en models, the MQM results are more clear. There are larger differences between the MQM scores and both BLEU and COMET-22 agree with the results. The fluency results are not quite as clear, with little noticeable difference between the ParIce 21.10 and ParIce Refined mBART models. In that case, the fluency results are in line with the *linguistic conventions* results for mBART, indicating that there may not be much difference between the models in that regard. In contrast, translation accuracy is higher when our processing methods have been applied to the training data. Fluency is higher for the mBART models than for the Transformer$_{\text{BASE}}$ models, which seems to be the greatest benefit of using the mBART models. More training data is probably needed in order to improve fluency for these models, when translating into a morphologically rich language such as Icelandic. Tang et al. (2021) show that the largest improvements mBART models show in terms of BLEU, as compared to training MT models from scratch, are in low-resource scenarios. Detailed error analysis of these models would be interesting to see if the improvements are, as in our case, mostly in terms of fluency rather than accuracy.

In this chapter our aim was to investigate whether the approaches we have developed for preparing training data for MT is useful to produce better quality MT systems. While not all of the metrics and evaluation approaches applied agree all of the time, they generally agree that ParIce refined produces better MT models than ParIce 21.10, and using all our processed datasets, as described in Section 7.2, produces better MT models than ParIce Refined. Our evaluation indicates that this may be more true for translation accuracy than for fluency.

Chapter 8

Conclusions and Future work

This thesis aimed to explore approaches for making better use of available parallel data when training MT systems.

For the vast majority of the world's languages, parallel data is scarce. When building MT systems for these languages it is highly important to be able to extract all useful sentence pairs from the available data. It has been argued that unsupervised methods can be key for building MT systems when parallel data is scarce. However, recent work has shown poor results for low-resource languages, questioning the role of unsupervised NMT for dissimilar languages and open-domain MT (Kim et al., 2020). Recent efforts have been made to extend accessibility of MT to 200 languages (Costa-jussà et al., 2022), but this still leaves out the vast majority of the world's languages and over one billion native speakers of these languages (Joshi et al., 2020). Furthermore, even though more languages have access to MT, it may not be up to the standard required if it is trained on defective data. While the technology is sensitive to using noisy data, there is a risk of losing potential for better quality MT if proper measures are not taken to prepare the data using the best approaches possible.

In this thesis, we have studied this problem by analyzing how to improve individual steps in parallel corpora compilation and in preparing training sets for MT. After having developed the necessary tools and datasets for our research, we experimented with multiple approaches for filtering parallel corpora and aligning parallel documents, before we set out to mine data from comparable corpora. We then applied comparable corpora mining techniques to extract useful parallel sentence pairs from what are normally classified as defective sentence pairs, as well as from data discarded during the parallel corpora compilation phase. Finally, we manually evaluated our combined approaches on a downstream MT task and compared the results to a baseline.

At the start of this thesis, we put forward four research questions concerning the importance of improved selection of training data for MT, and how we can ensure that our training sets are as good as possible. In the following section we will revisit these questions and review how we addressed them.

8.1 Research Questions

The aim of our thesis was to address the following four main research questions, set forward in Chapter 1:

RQ1: How can we filter parallel corpora to minimize noise, and still lose little or no useful data from the original texts?

In Chapter 4 we addressed this question. We looked at different filtering mechanisms for scoring and classifying sentence pairs, as well as traditional shallow filtering approaches. We manually annotated samples of data at different stages of filtering as well as computing BLEU scores for MT systems trained on the data. Our results indicate that applying shallow filtering is not sufficient and that mechanisms calculating scores for semantic equivalence should be included in the filtering process. There are various approaches available for minimizing noise, but when they return filtered data with little or no noise, the majority of the rejected sentence pairs are usually also acceptable and potentially useful for MT training, as we found in our manual evaluation of filtered data. This means there is still potential for extracting more sentence pairs which are potentially useful for MT training. When working on RQ4 we revisit this problem, especially focusing on rejected and discarded data.

We compared our results to the results of two systems, participating in the WMT 2021 news translation task, trained in a very similar manner to our systems. The comparison indicates that even our smaller $\text{Transformer}_{\text{BASE}}$ achieves comparable translation quality, using fewer resources and only a fraction of the training time. We also see from our results in Chapter 4 that models that have been filtered more thoroughly seem to converge faster. We can deduce from this that training data that is better filtered, not only improves MT output quality, but is also in line with a call for greener and more sustainable models of AI which consume less electricity and output fewer emissions, see e.g. Yusuf et al. (2021) and Jooste et al. (2022).

RQ2: To what degree should we consider filtering parallel corpora for MT training to be independent of the dataset and languages being filtered, and the intended translation direction of the MT system being built?

This question was also addressed in Chapter 4. Our results clearly indicate that different filtering approaches suit different datasets. We studied two very different corpora, compiled in different ways. After using similar shallow filtering approaches, we compared the quality of the data using a number of scoring mechanisms and found that the scoring mechanisms should have different acceptability thresholds depending on the dataset. We did not try to investigate why this is the case, but speculate that it may in part be due to different levels of prevalence of domain-specific texts in the data. Texts in some domains may, for example, commonly have more rare words or loan words from other languages, distinct syntactic structures, or other factors that lead to lowering the confidence in scoring the sentence pairs. We suggest that the most suitable filtering approaches are chosen after a careful analysis of what is most viable for the given dataset.

In relation to the question on translation direction, our results clearly indicate that it is not necessarily best to train $L_1 \rightarrow L_2$ using the same data as when training $L_2 \rightarrow L_1$. Considering our experiments, there is a clear argument for applying special filtering approaches for each translation direction. In our case, when translating into Icelandic, we seem to need more thorough filtering, and hypothesise that the morphological complexity of the target language may play a role. MRLs have a large number of word forms that may create an OOV-problem when training data is scarce, and, in such cases, tend to be more sensitive to flawed or erroneous sentences. As we discuss in Section 8.2, more work is needed to understand this issue.

RQ3: Is sentence alignment accuracy important for the results of a downstream MT task, or is effective filtering of the training data enough?

In Chapter 5, we aligned a corpus of EEA regulatory documents using various sentence alignment tools, as well as building our own. We also tried using ensembles of aligners in a two-step approach. After alignment, we filtered our data using the best filtering approaches for the dataset and translation direction as determined by our experiments in Chapter 4. We obtained significant results as measured by BLEU in downstream MT tasks, confirming that some alignment approaches are better than others, with our own tool, SentAlign, showing the best results. However, the scores of all the best-performing systems are in a rather small range so the effect may not be very large. To answer our research question, we conclude that sentence alignment accuracy is important, but effective filtering is necessary.

RQ4: Are text segments discarded during sentence alignment and filtering suitable as a source for mining useful sentence pairs for MT training?

In Chapter 6, we experimented with two approaches to make better use of discarded or deficient data. In Section 6.3, we re-evaluated sentence pairs that were more likely than other sentence pairs to be only partially aligned and contain erroneous data. We looked for better alignments by segmenting the sentence pairs and scoring pairs on the sub-sentential level. Using this method, we replaced some of the sentence pairs in the training data with more accurate alignments, raising BLEU scores for translations into Bengali, an MRL, but not for translations into English. As with our work on RQ3, this raises the question about whether NMT is more sensitive to noise in the training data when the target language is a MRL, explaining why cleaning the training data has more positive effect on translating into the MRL than into English.

In Section 6.4, we mined parallel sentence pairs from data discarded during the compilation phase of a parallel corpus. We collected sentences in both languages that either did not obtain an alignment from the sentence aligner or were discarded by a filter. We treated the collection of discarded sentences in the two languages as comparable corpora and looked for potential parallel candidates for each sentence in all the data discarded in the other language. After selecting the best candidates and again filtering the pairs, we acquired a small set of additional sentence pairs. We added that set to the previous training data and measured BLEU scores on a downstream MT task, finding that the BLEU scores improved. While the gain was small, it was statistically significant.

In light of these two experiments, we conclude that there is potential in exploiting discarded data and re-evaluating low-confidence sentence pairs in a parallel corpus. These results can also serve as a partial answer to RQ1, as an approach to minimize the useful data lost from a parallel corpus when we use it to compile training data for MT.

8.2 Future Work

We have explored multiple aspects of compiling training data for MT and the development of tools and data to support that work. In this section, we will discuss possible paths for expanding our research. There are immense possibilities for further studies in this area and we will give a few concrete examples.

8.2.1 Filtering

While we see some general tendencies in our filtering experiments, they do not show us what data exactly are detrimental and which are beneficial. In our manual evaluation of the filtered data, we used the taxonomy by Kreutzer et al. (2022) developed for evaluating web-crawled corpora, but it may not necessarily make clear distinctions between sentence pairs that can be beneficial for MT training and those that are not. In future work, we want to delve deeper into this issue and investigate if the differences between the datasets used for training in our work can give us an idea of which sentence pairs are most important to filter out. We intend to do this by investigating the pairs discarded by our filters, to compare what is being thrown away, both when it leads to higher quality models and when it leads to lower quality models. This could lead to insights that help with constructing filters that work on a more fine-grained level when that is needed. We would also like to carry out similar experiments for other language pairs, with both morphologically complex languages and more simple ones, to see if our results hold. In doing so, we might also explore whether more filtering benefits morphologically complex target languages more than morphologically simpler target languages.

We experimented with a GPT-2 model to identify sentences that do not look like the typical Icelandic sentences the model is trained on. The training set we used is quite limited, only 10,000 sentences, and in the training set there is a clear difference between admissible and inadmissible sentences. Enlarging the training dataset and adding a category for sentences that have minor flaws, such as spelling or grammatical errors, may be useful if these errors are only in the intended source language of the training data set and not in the target language. This could make the filter more flexible, depending on the quality we need for the task, or translation direction, at hand.

Finally, studying more filtering approaches may help us produce even better training sets, especially if we can learn more about what kind of sentence pairs we want to keep in our training data and which we want to leave out. In Chapter 7, we experimented with automatic evaluation models based on cross-lingual LLMs. Advances have been made in building such models that do not need any references (Agrawal et al., 2021; Rei et al., 2021). Studying whether they could help with filtering training data could be worthwhile.

8.2.2 Alignment

Our results indicate that SentAlign is the best-performing system out of the ones we evaluated. While the results are statistically significant, all our scores are in a rather small range. This may be an effect of the homogeneity of our data. We are selecting from millions of lines of EEA regulatory texts and the advantages of gaining a marginal amount of quality sentence pairs may not be great. To better confirm the edge SentAlign seems to have on the other systems, aligning more datasets and comparing the results would be useful.

While using a contextualized sentence-embedding-based scoring mechanism such as LaBSE seems to be very useful for the task of sentence alignment, it would also be interesting to experiment with replacing it with another one, e.g. lexical or translation-based, to further study our alignment algorithm and compare it against approximation algorithms in different scenarios. Our experiments also show that using the lexical-based Hunalign system aided by an external dictionary can give good results. To improve the accuracy of our sentence alignment approaches even further, we would like to investigate whether combining these approaches in some way could improve the results.

Finally, we would like to have a better look at the effect of partial alignments on downstream MT. If we can measure the effects of various misalignments, it could help us construct more effective methods to align and filter parallel corpora for MT. We thus want to investigate how different kinds and levels of misalignments in a parallel corpus affect quality, the extent to which they are useful, and when they become detrimental.

8.2.3 Re-evaluating Discarded Data

In our work on re-evaluating discarded and deficient data, we noticed that when translating into English we did not see the same benefit as when translating into one of the two MRLs we experimented with, Bengali and Icelandic. This may indicate that most of the newly added data is already in the existing training set; as Bengali and Icelandic are MRLs with fewer examples of each word form, the coverage of the data (and the resulting MT systems) are extended by adding more data. Furthermore, this may imply that there is a difference in noise tolerance depending on language pairs or translation directions, with English being more noise tolerant and therefore better able to take advantage of noisy data than the MRLs. We want to test this hypothesis using more language pairs and more training datasets.

In refining the discarded data, we used a simple segmentation technique, segmenting on conjunctions and punctuation. We want to explore other methods of segmentation for these purposes, such as constituency parsing, that may produce more useful segmentation. Furthermore, it would be useful to investigate what types of sub-sentences help and which hinder performance.

8.2.4 Evaluation

Manual evaluation is an expensive and time-consuming effort. Therefore, doing a large and proper evaluation is hard. Recently, automatic metrics that have been shown to have a high correlation with human evaluation have been published. We used one of these metrics, COMET-22. Looking further into the correlation with the language pair we mostly work with, English–Icelandic, could be useful, as well as looking into fine-tuning the metric for use with that language pair.

Our results indicate that using mBART as the base system may improve translation quality more by improving fluency than improving translation accuracy. It would be interesting to perform comparisons on other language pairs, and perhaps do larger evaluation experiments, to see if the same effect will be exhibited. It would also be interesting to see whether further improving our translation models, e.g. by including the use of back-translations and checkpoint averaging, would improve some aspects of quality more than others by evaluating the results of such models using MQM.

8.3 Final Remarks

Our results show that, at least in some cases, more can be made of data that has been prepared for training MT systems. This is important because it means that better systems can be built using what we have available, sometimes even though we train smaller models, as indicated by our comparison with two submissions to the WMT21 shared translation task. This is also important because it means better systems can be built for languages with scarce resources, if proper care is taken to use the available resources well. In our experiments, we have explored a number of ways to do so, and, in this chapter, we have suggested paths for further research in this area.

With this thesis we wish to draw the attention to the importance of focusing on well-designed and purposeful ways of building resources for training MT systems. To build high-quality MT systems for the languages that have been left behind, we believe that comprehensive approaches, allowing for making the most of available data, are fundamental.

Bibliography

Adafre, Sisay Fissaha and Maarten de Rijke. 2006. Finding Similar Sentences across Multiple Languages in Wikipedia. In *Proceedings of the Workshop on NEW TEXT Wikis and blogs and other dynamic text sources*, pages 62–69, Trento, Italy. Association for Computational Linguistics.

Afli, Haithem, Loïc Barrault, and Holger Schwenk. 2015. Building and Using Multimodal Comparable Corpora for Machine Translation. *Natural Language Engineering*, 22(4):603 – 625.

Agić, Željko and Ivan Vulić. 2019. JW300: A Wide-Coverage Parallel Corpus for Low-Resource Languages. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3204–3210, Florence, Italy. Association for Computational Linguistics.

Agrawal, Sweta, George Foster, Markus Freitag, and Colin Cherry. 2021. Assessing Reference-Free Peer Evaluation for Machine Translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1158–1171, Online. Association for Computational Linguistics.

Akhbardeh, Farhad, Arkady Arkhangorodsky, Magdalena Biesialska, Ondřej Bojar, Rajen Chatterjee, Vishrav Chaudhary, Marta R. Costa-jussa, Cristina España-Bonet, Angela Fan, Christian Federmann, Markus Freitag, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Leonie Harter, Kenneth Heafield, Christopher Homan, Matthias Huck, Kwabena Amponsah-Kaakyire, Jungo Kasai, Daniel Khashabi, Kevin Knight, Tom Kocmi, Philipp Koehn, Nicholas Lourie, Christof Monz, Makoto Morishita, Masaaki Nagata, Ajay Nagesh, Toshiaki Nakazawa, Matteo Negri, Santanu Pal, Allahsera Auguste Tapo, Marco Turchi, Valentin Vydrin, and Marcos Zampieri. 2021. Findings of the 2021 Conference on Machine Translation (WMT21). In *Proceedings of the Sixth Conference on Machine Translation*, pages 1–88, Online. Association for Computational Linguistics.

Alkhouli, Tamer, Gabriel Bretschner, and Hermann Ney. 2018. On The Alignment Problem In Multi-Head Attention-Based Neural Machine Translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 177–185, Brussels, Belgium. Association for Computational Linguistics.

Allen, Robert B. 1987. Several Studies on Natural Language and Back-Propagation. In *Proceedings of the IEEE First International Conference on Neural Networks*, volume II, pages 335–341, Piscataway, NJ.

ALPAC. 1966. *Language and Machines: Computers in Translation and Linguistics : a Report*. National Academy of Sciences, National Research Council.

Arcan, Mihael, Daniel Torregrosa, Sina Ahmadi, and John P. McCrae. 2019. Inferring Translation Candidates for Multilingual Dictionary Generation with Multi-Way Neural Machine Translation. In *Proceedings of TIAD-2019 Shared Task - Translation Inference Across Dictionaries co-located with the 2nd Language, Data and Knowledge Conference (LDK 2019)*, pages 13–23, Leipzig, Germany. CEUR-WS.

Aroyo, Lora, Matthew Lease, Praveen Paritosh, and Mike Schaekermann. 2022. Data Excellence for AI: Why Should You Care? *Interactions*, 29(2):66–69.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2016. Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2289–2294, Austin, Texas. Association for Computational Linguistics.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2018. 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. Association for Computational Linguistics.

Artetxe, Mikel, Gorka Labaka, and Eneko Agirre. 2019. 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. Association for Computational Linguistics.

Artetxe, Mikel and Holger Schwenk. 2019a. Margin-based Parallel Corpus Mining with Multilingual Sentence Embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy.

Artetxe, Mikel and Holger Schwenk. 2019b. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Arthur, Philip, Graham Neubig, and Satoshi Nakamura. 2016. Incorporating Discrete Translation Lexicons into Neural Machine Translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567, Austin, Texas. Association for Computational Linguistics.

Azpeitia, Andoni, Thierry Etchegoyhen, and Eva Martínez Garcia. 2018. Extracting Parallel Sentences from Comparable Corpora with STACC Variants. In *Proceedings of the 11th Workshop on Building and Using Comparable Corpora*, pages 48–52, Miyazaki, Japan. European Language Resources Association.

Azpeitia, Andoni, Thierry Etchegoyhen, and Eva Martínez Garcia. 2017. Weighted Set-Theoretic Alignment of Comparable Sentences. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 41–45, Vancouver, Canada. Association for Computational Linguistics.

Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio. 2015. 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*.

Banerjee, Satanjeev and Alon Lavie. 2005. METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Bañón, Marta, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel L. Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, Sergio Ortiz Rojas, Leopoldo Pla Sempere, Gema Ramírez-Sánchez, Elsa Sarrías, Marek Strelec, Brian Thompson, William Waites, Dion Wiggins, and Jaume Zaragoza. 2020. ParaCrawl: Web-Scale Acquisition of Parallel Corpora. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4555–4567, Online. Association for Computational Linguistics.

Barkarson, Starkaður and Steinþór Steingrímsson. 2019. Compiling and Filtering ParIce: An English-Icelandic Parallel Corpus. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 140–145, Turku, Finland. Linköping University Electronic Press.

Barkarson, Starkaður and Steinþór Steingrímsson. 2020. ParIce dev/test/train splits 20.05. CLARIN-IS.

Barkarson, Starkaður, Steinþór Steingrímsson, Finnur Ágúst Ingimundarson, Hildur Hafsteinsdóttir, and Árni Davíð Magnússon. 2021. ParIce dev/test sets 21.10. CLARIN-IS.

Barrault, Loïc, Magdalena Biesialska, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Yvette Graham, Roman Grundkiewicz, Barry Haddow, Matthias Huck, Eric Joanis, Tom Kocmi, Philipp Koehn, Chi-kiu Lo, Nikola Ljubešić, Christof Monz, Makoto Morishita, Masaaki Nagata, Toshiaki Nakazawa, Santanu Pal, Matt Post, and Marcos Zampieri. 2020. Findings of the 2020 Conference on Machine Translation (WMT20). In *Proceedings of the Fifth Conference on Machine Translation*, pages 1–55, Online. Association for Computational Linguistics.

Belinkov, Yonatan, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. 2017. What do Neural Machine Translation Models Learn about Morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 861–872, Vancouver, Canada. Association for Computational Linguistics.

Bentivogli, Luisa, Mauro Cettolo, Marcello Federico, and Federmann Christian. 2018. Machine Translation Human Evaluation: an investigation of evaluation based on Post-Editing and its relation with Direct Assessment. In *Proceedings of the 15th International Conference on Spoken Language Translation*, pages 62–69, Brussels. International Conference on Spoken Language Translation.

Bjarnadóttir, Kristín, Kristín Ingibjörg Hlynsdóttir, and Steinþór Steingrímsson. 2019. DIM: The Database of Icelandic Morphology. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 146–154, Turku, Finland. Linköping University Electronic Press.

Bjarnadóttir, Kristín. 2012. The Database of Modern Icelandic Inflection (Beygingarlýsing íslensks nútímamáls). In *Proceedings of the Workshop on Language Technology for*

Normalisation of Less-Resourced Languages – SaLTMiL 8 – AfLaT2012, pages 13–18, Istanbul, Turkey.

Blasi, Damian, Antonios Anastasopoulos, and Graham Neubig. 2022. Systematic Inequalities in Language Technology Performance across the World's Languages. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5486–5505, Dublin, Ireland. Association for Computational Linguistics.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurélie Névéol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 131–198, Berlin, Germany. Association for Computational Linguistics.

Bouamor, Houda and Hassan Sajjad. 2018. H2@BUCC18: Parallel Sentence Extraction from Comparable Corpora Using Multilingual Sentence Embeddings. In *Proceedings of the 11th Workshop on Building and Using Comparable Corpora at LREC 2018*, pages 43–47, Miyazaki, Japan. European Language Resources Association (ELRA).

Brandt, Martha Dís, Hrafn Loftsson, Hlynur Sigurþórsson, and Francis M. Tyers. 2011. Apertium-IceNLP: A rule-based Icelandic to English machine translation system. In *Proceedings of the 15th Annual conference of the European Association for Machine Translation*, pages 217–224, Leuven, Belgium. European Association for Machine Translation.

Braune, Fabienne and Alexander Fraser. 2010. Improved Unsupervised Sentence Alignment for Symmetrical and Asymmetrical Parallel Corpora. In *Coling 2010: Posters*, pages 81–89, Beijing, China. Coling 2010 Organizing Committee.

Breiman, Leo. 2001. Random Forests. *Machine Learning*, 45(1):5–32.

Briakou, Eleftheria and Marine Carpuat. 2021. Beyond Noise: Mitigating the Impact of Fine-grained Semantic Divergences on Neural Machine Translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7236–7249, Online. Association for Computational Linguistics.

Brown, Peter F., John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A Statistical Approach to Machine Translation. *Computational Linguistics*, 16(2):79–85.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The Mathematics of Statistical Machine Translation: Parameter Estimation. *Computational Linguistics*, 19(2):263–311.

Brown, Peter F., Jennifer C. Lai, and Robert L. Mercer. 1991. Aligning Sentences in Parallel Corpora. In *29th Annual Meeting of the Association for Computational Linguistics*, pages 169–176, Berkeley, California, USA. Association for Computational Linguistics.

Brown, Tom, 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. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*, pages 1877–1901, Vancouver, Canada. Curran Associates, Inc.

Callison-Burch, Chris, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2007. (Meta-) Evaluation of Machine Translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 136–158, Prague, Czech Republic. Association for Computational Linguistics.

Callison-Burch, Chris, Philipp Koehn, Christof Monz, Josh Schroeder, and Cameron Shaw Fordyce, editors. 2008. *Proceedings of the Third Workshop on Statistical Machine Translation*. Association for Computational Linguistics, Columbus, Ohio.

Callison-Burch, Chris, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the Role of Bleu in Machine Translation Research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, pages 249–256, Trento, Italy. Association for Computational Linguistics.

Caseli, Helena M., Maria das Graças V. Nunes, and Mikel L. Forcada. 2006. Automatic induction of bilingual resources from aligned parallel corpora: application to shallow-transfer machine translation. *Machine Translation*, 20:227–245.

Chalmers, David J. 1992. Syntactic Transformations on Distributed Representations. In Noel Sharkey, editor, *Connectionist Natural Language Processing: Readings from Connection Science*, pages 46–55. Springer Netherlands.

Chaudhary, Vishrav, Yuqing Tang, Francisco Guzmán, Holger Schwenk, and Philipp Koehn. 2019. Low-Resource Corpus Filtering Using Multilingual Sentence Embeddings. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 261–266, Florence, Italy. Association for Computational Linguistics.

Chen, Stanley F. 1993. Aligning Sentences in Bilingual Corpora Using Lexical Information. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 9–16, Columbus, Ohio, USA. Association for Computational Linguistics.

Chimoto, Everlyn and Bruce Bassett. 2022. Very Low Resource Sentence Alignment: Luhya and Swahili. In *Proceedings of the Fifth Workshop on Technologies for Machine Translation of Low-Resource Languages (LoResMT 2022)*, pages 1–8, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Cho, Kyunghyun, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Chrisman, Lonnie. 1991. Learning recursive distributed representations for holistic computation. *Connection Science*, 3:345–366.

Chu, Chenhui, Toshiaki Nakazawa, and Sadao Kurohashi. 2015. Integrated Parallel Sentence and Fragment Extraction from Comparable Corpora: A Case Study on Chinese–Japanese Wikipedia. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 15(2).

Cohen, J. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Conneau, Alexis, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Conneau, Alexis, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. 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. Association for Computational Linguistics.

Conneau, Alexis and Guillaume Lample. 2019. Cross-lingual Language Model Pretraining. In *Advances in Neural Information Processing Systems 32 (NeurIPS 2019)*, volume 32, page 7059–7069, Vancouver, Canada. Curran Associates, Inc.

Cortes, Corinna and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning*, 20(3):273–297.

Costa-jussà, Marta Ruiz, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Alison Youngblood, Bapi Akula, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon L. Spruit, C. Tran, Pierre Yves Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzm'an, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation. *ArXiv*, abs/2207.04672.

Cox, David R. 1958. The Regression Analysis of Binary Sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2):215–232.

Defauw, Arne, Sara Szoc, Anna Bardadym, Joris Brabers, Frederic Everaert, Roko Mijic, Kim Scholte, Tom Vanallemeersch, Koen Van Winckel, and Joachim Van den Bogaert. 2019. Misalignment Detection for Web-Scraped Corpora: A Supervised Regression Approach. *Informatics*, 6(3):35.

Dehghani, Mostafa, Stephan Gouws, Oriol Vinyals, Jakob Uszkoreit, and Łukasz Kaiser. 2019. Universal Transformers. In *7th International Conference on Learning Representations, ICLR 2019, Conference Track Proceedings*, New Orleans, Louisiana.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings* *of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Dhar, Prajit, Arianna Bisazza, and Gertjan van Noord. 2022. Evaluating Pre-training Objectives for Low-Resource Translation into Morphologically Rich Languages. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4933–4943, Marseille, France. European Language Resources Association.

Dijkstra, Edsger W. 1959. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.

Dinu, Georgiana and Marco Baroni. 2015. Improving zero-shot learning by mitigating the hubness problem. In *3rd International Conference on Learning Representations, ICLR 2015, Workshop Track Proceedings*, San Diego, California.

Doddington, George. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the Second International Conference on Human Language Technology Research*, HLT '02, page 138–145, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Dostert, Léon. 1955. The Georgetown-I.B.M. experiment. In A. D. Booth W. N. Locke, editor, *Machine translation of languages*, pages 124–135. MIT Press, Cambridge, Mass.

Dou, Zi-Yi and Graham Neubig. 2021. Word Alignment by Fine-tuning Embeddings on Parallel Corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics.

Dyer, Chris, Victor Chahuneau, and Noah A. Smith. 2013. 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*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

El-Kishky, Ahmed, Vishrav Chaudhary, Francisco Guzmán, and Philipp Koehn. 2020. CCAligned: A Massive Collection of Cross-Lingual Web-Document Pairs. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5960–5969, Online. Association for Computational Linguistics.

Esplà-Gomis, Miquel. 2009. Bitextor: a Free/Open-source Software to Harvest Translation Memories from Multilingual Websites. In *Beyond Translation Memories: New Tools for Translators Workshop*, Ottawa, Canada.

Esplà-Gomis, Miquel, Víctor M. Sánchez-Cartagena, Jaume Zaragoza-Bernabeu, and Felipe Sánchez-Martínez. 2020. Bicleaner at WMT 2020: Universitat d'Alacant-Prompsit's submission to the parallel corpus filtering shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 952–958, Online. Association for Computational Linguistics.

Fan, Angela, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2021. Beyond English-Centric Multilingual Machine Translation. *The Journal of Machine Learning Research*, 22(1):4839–4886.

Feng, Fangxiaoyu, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT Sentence Embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Fernando, Aloka, Surangika Ranathunga, Dilan Sachintha, Lakmali Piyarathna, and Charith Rajitha. 2023. Exploiting bilingual lexicons to improve multilingual embedding-based document and sentence alignment for low-resource languages. *Knowledge and Information Systems*, 65(2):571–612.

Fleiss, J.L. 1973. *Statistical methods for rates and proportions*. Wiley.

Forcada, Mikel, Mireia Ginestí-Rosell, Jacob Nordfalk, Jim O'Regan, Sergio Ortiz-Rojas, Juan Pérez-Ortiz, Felipe Sánchez-Martínez, Gema Ramírez-Sánchez, and Francis Tyers. 2011. Apertium: A free/open-source platform for rule-based machine translation. *Machine Translation*, 25:127–144.

Forcada, Mikel L. and Ramón P. Ñeco. 1997. Recursive hetero-associative memories for translation. In *Proceedings of the International Work-Conference on Artificial and Natural Neural Networks: Biological and Artificial Computation: From Neuroscience to Technology*, IWANN '97, page 453–462, Berlin, Heidelberg. Springer-Verlag.

Forcada, Mikel L., Carolina Scarton, Lucia Specia, Barry Haddow, and Alexandra Birch. 2018. Exploring Gap Filling as a Cheaper Alternative to Reading Comprehension Questionnaires when Evaluating Machine Translation for Gisting. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 192–203, Brussels, Belgium. Association for Computational Linguistics.

Freitag, Markus, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021a. Experts, Errors, and Context: A Large-Scale Study of Human Evaluation for Machine Translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Freitag, Markus, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F. T. Martins. 2022. Results of WMT22 Metrics Shared Task: Stop Using BLEU – Neural Metrics Are Better and More Robust. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 46–68, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Freitag, Markus, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. 2021b. Results of the WMT21 Metrics Shared Task: Evaluating Metrics with Expert-based Human Evaluations on TED and News Domain. In *Proceedings of the Sixth Conference on Machine Translation*, pages 733–774, Online. Association for Computational Linguistics.

Gage, Philip. 1994. A New Algorithm for Data Compression. *The C Users Journal*, 12(2):23–38.

Gale, William A. and Kenneth W. Church. 1991. A Program for Aligning Sentences in Bilingual Corpora. In *29th Annual Meeting of the Association for Computational Linguistics*, pages 177–184, Berkeley, California. Association for Computational Linguistics.

Gaspari, Federico, Owen Gallagher, Georg Rehm, Maria Giagkou, Stelios Piperidis, Jane Dunne, and Andy Way. 2022. Introducing the Digital Language Equality Metric: Technological Factors. In *Proceedings of the Workshop Towards Digital Language Equality within the 13th Language Resources and Evaluation Conference*, pages 1–12, Marseille, France. European Language Resources Association.

Goyal, Naman, Cynthia Gao, Vishrav Chaudhary, Peng-Jen Chen, Guillaume Wenzek, Da Ju, Sanjana Krishnan, Marc'Aurelio Ranzato, Francisco Guzmán, and Angela Fan. 2022. The Flores-101 Evaluation Benchmark for Low-Resource and Multilingual Machine Translation. *Transactions of the Association for Computational Linguistics*, 10:522–538.

Gracia, Jorge, Christian Fäth, Matthias Hartung, Max Ionov, Julia Bosque-Gil, Susana Veríssimo, Christian Chiarcos, and Matthias Orlikowski. 2020. Leveraging Linguistic Linked Data for Cross-Lingual Model Transfer in the Pharmaceutical Domain. In *The Semantic Web – ISWC 2020*, pages 499–514, Athens, Greece. Springer International Publishing.

Gracia, Jorge, Besim Kabashi, and Ilan Kernerman. 2021. Results of the Translation Inference Across Dictionaries 2021 Shared Task. In *Proceedings of the Workshops and Tutorials held at LDK 2021*, pages 208–220, Zaragosa, Spain. CEUR-WS.

Graham, Yvette, Timothy Baldwin, Meghan Dowling, Maria Eskevich, Teresa Lynn, and Lamia Tounsi. 2016. Is all that Glitters in Machine Translation Quality Estimation really Gold? In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3124–3134, Osaka, Japan. The COLING 2016 Organizing Committee.

Graham, Yvette, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2013. Continuous Measurement Scales in Human Evaluation of Machine Translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 33–41, Sofia, Bulgaria. Association for Computational Linguistics.

Graham, Yvette, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2017. Can machine translation systems be evaluated by the crowd alone. *Natural Language Engineering*, 23(1):3–30.

Graves, Alex and Jürgen Schmidhuber. 2005. Framewise Phoneme Classification with Bidirectional LSTM and Other Neural Network Architectures. *Neural Networks*, 18(5–6):602–610.

Hajič, Jan. 2000. Morphological Tagging: Data vs. Dictionaries. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 94–101, Seattle, Washington. Association for Computational Linguistics.

Hangya, Viktor and Alexander Fraser. 2019. 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. Association for Computational Linguistics.

Haruno, Masahiko and Takefumi Yamazaki. 1996. High-Performance Bilingual Text Alignment Using Statistical and Dictionary Information. page 131, Santa Cruz, California.

Hasan, Tahmid, Abhik Bhattacharjee, Kazi Samin, Masum Hasan, Madhusudan Basak, M. Sohel Rahman, and Rifat Shahriyar. 2020. Not Low-Resource Anymore: Aligner Ensembling, Batch Filtering, and New Datasets for Bengali-English Machine Translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2612–2623, Online. Association for Computational Linguistics.

Heffernan, Kevin, Onur Çelebi, and Holger Schwenk. 2022. Bitext Mining Using Distilled Sentence Representations for Low-Resource Languages. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2101–2112, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hendy, Amr, Mohamed Gomaa Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How Good Are GPT Models at Machine Translation? A Comprehensive Evaluation. *ArXiv*, abs/2302.09210.

Herold, Christian, Jan Rosendahl, Joris Vanvinckenroye, and Hermann Ney. 2022. Detecting Various Types of Noise for Neural Machine Translation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2542–2551, Dublin, Ireland. Association for Computational Linguistics.

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Horsmann, Tobias and Torsten Zesch. 2017. Do LSTMs really work so well for PoS tagging? – A replication study. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 727–736, Copenhagen, Denmark. Association for Computational Linguistics.

Ingólfsdóttir, Svanhvít Lilja, Hrafn Loftsson, Jón Friðrik Daðason, and Kristín Bjarnadóttir. 2019. Nefnir: A high accuracy lemmatizer for Icelandic. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 310–315, Turku, Finland. Linköping University Electronic Press.

Jónsson, Haukur Páll, Haukur Barri Símonarson, Vésteinn Snæbjarnarson, Steinþór Steingrímsson, and Hrafn Loftsson. 2020. Experimenting with Different Machine Translation Models in Medium-Resource Settings. In *Text, Speech, and Dialogue - 23rd International Conference, TSD 2020, Brno, Czech Republic, September 8-11, 2020, Proceedings*, volume 12284 of *Lecture Notes in Computer Science*, pages 95–103. Springer.

Jooste, Wandri, Rejwanul Haque, and Andy Way. 2022. Knowledge Distillation: A Method for Making Neural Machine Translation More Efficient. *Information*, 13(2).

Joshi, Pratik, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. 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. Association for Computational Linguistics.

Joulin, Armand, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. 2017. Bag of Tricks for Efficient Text Classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431, Valencia, Spain. Association for Computational Linguistics.

Junczys-Dowmunt, Marcin. 2019. Microsoft Translator at WMT 2019: Towards Large-Scale Document-Level Neural Machine Translation. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 225–233, Florence, Italy. Association for Computational Linguistics.

Jónsson, Haukur and Hrafn Loftsson. 2022. DMS: A System for Delivering Dynamic Multitask NLP Tools. In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence - Volume 1: NLPinAI,*, pages 504–510. INSTICC, SciTePress.

Kaalep, Heiki-Jaan and Kaarel Veskis. 2007. Comparing Parallel Corpora and Evaluating their Quality. In *Proceedings of Machine Translation Summit XI: Papers*, Copenhagen, Denmark.

Kalchbrenner, Nal and Phil Blunsom. 2013. Recurrent Continuous Translation Models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Karimi, Akbar, Ebrahim Ansari, and Bahram Sadeghi Bigham. 2018. Extracting an English-Persian Parallel Corpus from Comparable Corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, pages 3477–3482, Miyazaki, Japan. European Language Resources Association (ELRA).

Kay, Martin and Martin Röscheisen. 1993. Text-Translation Alignment. *Computational Linguistics*, 19(1):121–142.

Kendall, Maurice. 1938. A New Measure of Rank Correlation. *Biometrika*, 30(1–2):81–93.

Khadivi, Shahram and Hermann Ney. 2005. Automatic Filtering of Bilingual Corpora for Statistical Machine Translation. In *Natural Language Processing and Information Systems*, pages 263–274, Berlin, Heidelberg. Springer.

Khanna, Tanmai, Jonathan Washington, Francis Tyers, Sevilay Bayatlı, Daniel Swanson, Tommi Pirinen, Irene Tang, and Hèctor Alòs i Font. 2021. Recent advances in Apertium, a free/open-source rule-based machine translation platform for low-resource languages. *Machine Translation*, 35:1–28.

Khayrallah, Huda and Philipp Koehn. 2018. On the Impact of Various Types of Noise on Neural Machine Translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 74–83, Melbourne, Australia. Association for Computational Linguistics.

Kim, Yunsu, Miguel Graça, and Hermann Ney. 2020. When and Why is Unsupervised Neural Machine Translation Useless? In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 35–44, Lisboa, Portugal. European Association for Machine Translation.

Klein, Guillaume, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

Koehn, Philipp. 2004. 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. Association for Computational Linguistics.

Koehn, Philipp. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Koehn, Philipp. 2009. *Statistical Machine Translation*. Cambridge University Press.

Koehn, Philipp. 2020. *Neural Machine Translation*. Cambridge University Press.

Koehn, Philipp, Vishrav Chaudhary, Ahmed El-Kishky, Naman Goyal, Peng-Jen Chen, and Francisco Guzmán. 2020. Findings of the WMT 2020 Shared Task on Parallel Corpus Filtering and Alignment. In *Proceedings of the Fifth Conference on Machine Translation*, pages 726–742, Online. Association for Computational Linguistics.

Koehn, Philipp, Francisco Guzmán, Vishrav Chaudhary, and Juan Pino. 2019. Findings of the WMT 2019 Shared Task on Parallel Corpus Filtering for Low-Resource Conditions. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 54–72, Florence, Italy. Association for Computational Linguistics.

Koehn, Philipp, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open Source Toolkit for Statistical Machine Translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Koehn, Philipp, Huda Khayrallah, Kenneth Heafield, and Mikel L. Forcada. 2018. Findings of the WMT 2018 Shared Task on Parallel Corpus Filtering. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 726–739, Belgium, Brussels. Association for Computational Linguistics.

Koehn, Philipp and Christof Monz. 2006. Manual and Automatic Evaluation of Machine Translation between European Languages. In *Proceedings on the Workshop on Statistical Machine Translation*, pages 102–121, New York City. Association for Computational Linguistics.

Koehn, Philipp, Franz J. Och, and Daniel Marcu. 2003. Statistical Phrase-Based Translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133, Edmonton, Canada. Association for Computational Linguistics.

Koszowski, Mikołaj, Karol Grzegorczyk, and Tsimur Hadeliya. 2021. Allegro.eu Submission to WMT21 News Translation Task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 140–143, Online. Association for Computational Linguistics.

Kreutzer, Julia, 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. 2022. Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets. *Transactions of the Association for Computational Linguistics*, 10:50–72.

Kudo, Taku and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Kurfalı, Murathan and Robert Östling. 2019. Noisy Parallel Corpus Filtering through Projected Word Embeddings. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 277–281, Florence, Italy. Association for Computational Linguistics.

Lample, Guillaume, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. In *6th International Conference on Learning Representations, ICLR 2018, Conference Track Proceedings*, Vancouver, Canada.

Lamraoui, Fethi and Philippe Langlais. 2013. Yet Another Fast, Robust and Open Source Sentence Aligner. Time to Reconsider Sentence Alignment? In *Proceedings of Machine Translation Summit XIV: Papers*, Nice, France.

Landis, J. Richard and Gary G. Koch. 1977. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159–174.

Läubli, Samuel, Sheila Castilho, Graham Neubig, Rico Sennrich, Qinlan Shen, and Antonio Toral. 2020. A Set of Recommendations for Assessing Human-Machine Parity in Language Translation. *Journal of Artificial Intelligence Research*, 67:653–672.

Lee, Katherine, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating Training Data Makes Language Models Better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8424–8445, Dublin, Ireland. Association for Computational Linguistics.

Levenshtein, Vladimir Iosifovich. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Li, Peng, Maosong Sun, and Ping Xue. 2010. Fast-Champollion: A Fast and Robust Sentence Alignment Algorithm. In *Coling 2010: Posters*, pages 710–718, Beijing, China. Coling 2010 Organizing Committee.

Ling, Wang, Chris Dyer, Alan W Black, Isabel Trancoso, Ramon Fermandez, Silvio Amir, Luis Marujo, and Tiago Luis. 2015. Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1520–1530, Lisbon, Portugal. Association for Computational Linguistics.

Ling, Wang, Luís Marujo, Chris Dyer, Alan W. Black, and Isabel Trancoso. 2014. Crowdsourcing High-Quality Parallel Data Extraction from Twitter. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 426–436, Baltimore, Maryland, USA. Association for Computational Linguistics.

Liu, Lemao, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. 2016. Neural Machine Translation with Supervised Attention. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3093–3102, Osaka, Japan. The COLING 2016 Organizing Committee.

Liu, Siyou, Yuqi Sun, and Longyue Wang. 2021. Recent Advances in Dialogue Machine Translation. *Information*, 12(11).

Liu, Xiaodong, Kevin Duh, Liyuan Liu, and Jianfeng Gao. 2020a. Very Deep Transformers for Neural Machine Translation. *ArXiv*, abs/2008.07772.

Liu, Yinhan, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020b. Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Lo, Chi-kiu and Eric Joanis. 2020. Improving Parallel Data Identification using Iteratively Refined Sentence Alignments and Bilingual Mappings of Pre-trained Language Models. In *Proceedings of the Fifth Conference on Machine Translation*, pages 972–978, Online. Association for Computational Linguistics.

Lo, Chi-kiu and Michel Simard. 2019. Fully Unsupervised Crosslingual Semantic Textual Similarity Metric Based on BERT for Identifying Parallel Data. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 206–215, Hong Kong, China. Association for Computational Linguistics.

Loftsson, Hrafn, Sigrún Helgadóttir, and Eiríkur Rögnvaldsson. 2011. Using a Morphological Database to Increase the Accuracy in POS Tagging. In *Recent Advances in Natural Language Processing*, RANLP 2011, pages 49–55, Hissar, Bulgaria. Association for Computational Linguistics.

Loftsson, Hrafn, Ida Kramarczyk, Sigrún Helgadóttir, and Eiríkur Rögnvaldsson. 2009. Improving the PoS tagging accuracy of Icelandic text. In *Proceedings of the 17th Nordic Conference on Computational Linguistics*, NODALIDA 2009, pages 103–110, Odense, Denmark. Northern European Association for Language Technology (NEALT).

Loftsson, Hrafn and Robert Östling. 2013. Tagging a Morphologically Complex Language Using an Averaged Perceptron Tagger: The Case of Icelandic. In *Proceedings of the 19th Nordic Conference of Computational Linguistics (NODALIDA 2013)*, pages 105–119, Oslo, Norway. Linköping University Electronic Press, Sweden.

Loftsson, Hrafn, Jökull H. Yngvason, Sigrún Helgadóttir, and Eiríkur Rögnvaldsson. 2010. Developing a PoS-tagged corpus using existing tools. In *Proceedings of 7th SaLTMiL Workshop on Creation and Use of Basic Lexical Resources for Less-Resourced Languages*, LREC 2010, pages 53–60, Valetta, Malta.

Lohar, Pintu, Debasis Ganguly, Haithem Afli, Andy Way, and Gareth J. F. Jones. 2016. FaDA: Fast Document Aligner using Word Embedding. *The Prague Bulletin of Mathematical Linguistics*, 106:169–179.

Lommel, Arle, Hans Uszkoreit, and Aljoscha Burchardt. 2014. Multidimensional Quality Metrics (MQM): A Framework for Declaring and Describing Translation Quality Metrics. *Revista Tradumàtica: Tecnologies de la Traducció*, pages 455–463.

Lu, Jun, Xin Ge, Yangbin Shi, and Yuqi Zhang. 2020. Alibaba Submission to the WMT20 Parallel Corpus Filtering Task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 979–984, Online. Association for Computational Linguistics.

Luong, Minh-Thang, Hieu Pham, and Christopher D. Manning. 2015a. Bilingual Word Representations with Monolingual Quality in Mind. In *NAACL Workshop on Vector Space Modeling for NLP*, pages 151–159, Denver, Colorado. Association for Computational Linguistics.

Luong, Thang, Hieu Pham, and Christopher D. Manning. 2015b. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

Ma, Qingsong, Johnny Wei, Ondřej Bojar, and Yvette Graham. 2019. Results of the WMT19 Metrics Shared Task: Segment-Level and Strong MT Systems Pose Big Challenges. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 62–90, Florence, Italy. Association for Computational Linguistics.

Ma, Xiaoyi. 2006. Champollion: A Robust Parallel Text Sentence Aligner. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC 2006)*, pages 489–492, Genoa, Italy. European Language Resources Association (ELRA).

Mareček, David. 2008. Automatic Alignment of Tectogrammatical Trees from Czech-English Parallel Corpus. Master's thesis, Charles University.

Mariana, Valerie, Troy Cox, and Alan Melby. 2015. The Multidimensional Quality Metrics (MQM) Framework: A New Framework for Translation Quality Assessment. *The Journal of Specialised Translation*, 23:137–161.

Martin, Joel, Howard Johnson, Benoit Farley, and Anna Maclachlan. 2003. Aligning and Using an English-Inuktitut Parallel Corpus. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 115–118, Edmonton, Canada. Association for Computational Linguistics.

Martinez, Raquel, Joseba Abaitua, and Arantza Casillas. 1998. Bitext Correspondences through Rich Mark-up. In *COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics*, pages 812–818, Montreal, Canada. Association for Computational Linguistics.

Masoud, Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High Quality Word Alignments Without Parallel Training Data Using Static and Contextualized Embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics.

Mazumder, Mark, Colby R. Banbury, Xiaozhe Yao, Bojan Karlavs, William Gaviria Rojas, Sudnya Diamos, Gregory Frederick Diamos, Lynn He, Douwe Kiela, David Jurado, David Kanter, Rafael Mosquera, Juan Ciro, Lora Aroyo, Bilge Acun, Sabri Eyuboglu, Amirata Ghorbani, Emmett D. Goodman, Tariq Kane, Christine R. Kirkpatrick, Tzu-Sheng Kuo, Jonas Mueller, Tristan Thrush, Joaquin Vanschoren, Margaret J. Warren, Adina Williams, Serena Yeung, Newsha Ardalani, Praveen K. Paritosh, Ce Zhang, James Y. Zou, Carole-Jean Wu, Cody Coleman, Andrew Y. Ng, Peter Mattson, and Vijay Janapa Reddi. 2022. DataPerf: Benchmarks for Data-Centric AI Development. *ArXiv*, abs/2207.10062.

McHugh, Mary L. 2012. Interrater reliability: the kappa statistic. *Biochemia Medica*, 22(3):276–282.

Melamed, I. Dan. 2000. Models of Translation Equivalence among Words. *Computational Linguistics*, 26(2):221–250.

Mihalcea, Rada and Ted Pedersen. 2003. An Evaluation Exercise for Word Alignment. In *Proceedings of the HLT-NAACL 2003 Workshop on Building and Using Parallel Texts: Data Driven Machine Translation and Beyond*, pages 1–10, Edmonton, Canada.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations, ICLR 2013, Workshop Track Proceedings*, Scottsdale, Arizona.

Miller, George A. 1994. WordNet: A Lexical Database for English. In *Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994*.

Moore, Robert C. 2002. Fast and Accurate Sentence Alignment of Bilingual Corpora. In *Machine Translation: From Research to Real Users, 5th Conference of the Association for Machine Translation in the Americas*, pages 135–144, Tiburon, California. Springer.

Müller, Mathias. 2017. Treatment of Markup in Statistical Machine Translation. In *Proceedings of the Third Workshop on Discourse in Machine Translation*, pages 36–46, Copenhagen, Denmark. Association for Computational Linguistics.

Munteanu, Dragos Stefan and Daniel Marcu. 2006. Extracting Parallel Sub-Sentential Fragments from Non-Parallel Corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 81–88, Sydney, Australia. Association for Computational Linguistics.

Nikulásdóttir, Anna, Jón Guðnason, Anton Karl Ingason, Hrafn Loftsson, Eiríkur Rögnvaldsson, Einar Freyr Sigurðsson, and Steinþór Steingrímsson. 2020. Language Technology Programme for Icelandic 2019-2023. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 3414–3422, Marseille, France. European Language Resources Association.

Nikulásdóttir, Anna Björk, Þórunn Arnardóttir, Jón Guðnason, Þorsteinn Daði Gunnarsson, Anton Karl Ingason, Haukur Páll Jónsson, Hrafn Loftsson, Hulda Óladóttir, Einar Freyr Sigurðsson, Atli Þór Sigurgeirsson, Vésteinn Snæbjarnarson, and Steinþór Steingrímsson. 2022. Help Yourself from the Buffet: National Language Technology Infrastructure Initiative on CLARIN-IS. In *Selected Papers from the CLARIN Annual Conference 2021*, pages 109–125, Online. Linköping Electronic Press.

Nikulásdóttir, Anna Björk, Jón Guðnason, and Steinþór Steingrímsson. 2017. *Language Technology for Icelandic. Project Plan*. Icelandic Ministry of Science, Culture and Education.

Och, Franz Josef. 2003. 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, Japan. Association for Computational Linguistics.

Och, Franz Josef and Hermann Ney. 2000. Improved Statistical Alignment Models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong. Association for Computational Linguistics.

Och, Franz Josef and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.

Östling, Robert and Jörg Tiedemann. 2016. Efficient word alignment with Markov Chain Monte Carlo. *Prague Bulletin of Mathematical Linguistics*, 106:125–146.

Ott, Myle, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Ozdowska, Sylwia and Andy Way. 2009. Optimal Bilingual Data for French-English PB-SMT. In *Proceedings of the 13th Annual conference of the European Association for Machine Translation*, pages 96–103, Barcelona, Spain. European Association for Machine Translation.

Paetzold, Gustavo, Fernando Alva-Manchego, and Lucia Specia. 2017. MASSAlign: Alignment and Annotation of Comparable Documents. In *Proceedings of the IJCNLP 2017, System Demonstrations*, pages 1–4, Taipei, Taiwan. Association for Computational Linguistics.

Pal, Santanu, Sudip Kumar Naskar, Mihaela Vela, Qun Liu, and Josef van Genabith. 2017. Neural Automatic Post-Editing Using Prior Alignment and Reranking. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 349–355, Valencia, Spain. Association for Computational Linguistics.

Papageorgiou, Harris, Lambros Cranias, and Stelios Piperidis. 1994. Automatic Alignment in Parallel Corpora. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 334–336, Las Cruces, New Mexico. Association for Computational Linguistics.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania. Association for Computational Linguistics.

Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pind, Jörgen, Friðrik Magnússon, and Stefán Briem. 1991. *Íslensk orðtíðnibók [The Icelandic Frequency Dictionary]*. The Institute of Lexicography, University of Iceland, Reykjavik, Iceland.

Pinnis, Mārcis. 2018. Tilde's Parallel Corpus Filtering Methods for WMT 2018. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 939–945, Belgium, Brussels. Association for Computational Linguistics.

Pirinen, Tommi, Francis M. Tyers, Trond Trosterud, Ryan Johnson, Kevin Unhammer, and Tiina Puolakainen. 2017. North-Sámi to Finnish rule-based machine translation system. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 115–122, Gothenburg, Sweden. Association for Computational Linguistics.

Plank, Barbara, Anders Søgaard, and Yoav Goldberg. 2016. Multilingual Part-of-Speech Tagging with Bidirectional Long Short-Term Memory Models and Auxiliary Loss. In *Proceedings of the* 54^{th} *Annual Meeting of the Association for Computational Linguistics*, pages 412–418, Berlin, Germany. Association for Computational Linguistics.

Poncelas, Alberto, Maja Popović, Dimitar Shterionov, Gideon Maillette de Buy Wenniger, and Andy Way. 2019. Combining PBSMT and NMT Back-translated Data for Efficient NMT. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 922–931, Varna, Bulgaria. INCOMA Ltd.

Popović, Maja. 2015. 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. Association for Computational Linguistics.

Post, Matt. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Press, Ofir and Noah A. Smith. 2018. You May Not Need Attention. *ArXiv*, abs/1810.13409.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9.

Rajitha, M. D. C, Piyarathna L.L. C, Nayanajith M. M.D. S, and Surangika S. 2020. Sinhala and English Document Alignment using Statistical Machine Translation. pages 29–34, Colombo, Sri Lanka. IEEE.

Ramesh, Gowtham, Sumanth Doddapaneni, Aravinth Bheemaraj, Mayank Jobanputra, Raghavan AK, Ajitesh Sharma, Sujit Sahoo, Harshita Diddee, Mahalakshmi J, Divyanshu Kakwani, Navneet Kumar, Aswin Pradeep, Srihari Nagaraj, Kumar Deepak, Vivek

Raghavan, Anoop Kunchukuttan, Pratyush Kumar, and Mitesh Shantadevi Khapra. 2022. Samanantar: The Largest Publicly Available Parallel Corpora Collection for 11 Indic Languages. *Transactions of the Association for Computational Linguistics*, 10:145–162.

Ramírez-Sánchez, Gema, Jaume Zaragoza-Bernabeu, Marta Bañón, and Sergio Ortiz-Rojas. 2020. Bifixer and Bicleaner: two open-source tools to clean your parallel data. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 291–298, Lisbon, Portugal. European Association for Machine Translation.

Rapp, Reinhard, Pierre Zweigenbaum, and Serge Sharoff. 2020. Overview of the Fourth BUCC Shared Task: Bilingual Dictionary Induction from Comparable Corpora. In *Proceedings of the 13th Workshop on Building and Using Comparable Corpora*, pages 6–13, Marseille, France. European Language Resources Association.

Rarrick, Spencer, Chris Quirk, and Will Lewis. 2011. MT Detection in Web-Scraped Parallel Corpora. In *Proceedings of Machine Translation Summit XIII: Papers*, pages 422–429, Xiamen, China.

Rei, Ricardo, José G. C. De Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 Submission for the Metrics Shared Task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Rei, Ricardo, Ana C Farinha, Chrysoula Zerva, Daan van Stigt, Craig Stewart, Pedro Ramos, Taisiya Glushkova, André F. T. Martins, and Alon Lavie. 2021. Are References Really Needed? Unbabel-IST 2021 Submission for the Metrics Shared Task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 1030–1040, Online. Association for Computational Linguistics.

Rei, Ricardo, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. 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. Association for Computational Linguistics.

Rikters, Matiss. 2018. Impact of Corpora Quality on Neural Machine Translation. In *Human Language Technologies–The Baltic Perspective: Proceedings of the Eighth International Conference Baltic HLT 2018*, pages 126–133, Tartu, Estonia. IOS Press.

Rossenbach, Nick, Jan Rosendahl, Yunsu Kim, Miguel Graça, Aman Gokrani, and Hermann Ney. 2018. The RWTH Aachen University Filtering System for the WMT 2018 Parallel Corpus Filtering Task. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 946–954, Belgium, Brussels. Association for Computational Linguistics.

Rozis, Roberts and Raivis Skadiņš. 2017. Tilde MODEL - Multilingual Open Data for EU Languages. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 263–265, Gothenburg, Sweden. Association for Computational Linguistics.

Rögnvaldsson, Eiríkur, Kristín M. Jóhannsdóttir, Sigrún Helgadóttir, and Steinþór Steingrímsson. 2012. *Íslensk tunga á stafrænni öld – The Icelandic Language in the Digital Age.* META-NET White Paper Series. Georg Rehm and Hans Uszkoreit (Series Editors). Springer.

Sagot, Benoît and Héctor Martínez Alonso. 2017. Improving neural tagging with lexical information. In *Proceedings of the* 15^{th} *International Conference on Parsing Technologies*, pages 25–31, Pisa, Italy. Association for Computational Linguistics.

Salvador, Stan and Philip Chan. 2007. Toward Accurate Dynamic Time Warping in Linear Time and Space. *Intelligent Data Analysis*, 11(5):561–580.

Samy, Doaa, Antonio Moreno Sandoval, José M. Guirao, and Enrique Alfonseca. 2006. Building a Parallel Multilingual Corpus (Arabic-Spanish-English). In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 2176–2181, Genoa, Italy. European Language Resources Association (ELRA).

Sánchez-Cartagena, Víctor M., Marta Bañón, Sergio Ortiz-Rojas, and Gema Ramírez. 2018. Prompsit's submission to WMT 2018 Parallel Corpus Filtering shared task. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 955–962, Belgium, Brussels. Association for Computational Linguistics.

Santos, Cicero Dos and Bianca Zadrozny. 2014. Learning Character-level Representations for Part-of-Speech Tagging. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1818–1826, Bejing, China. PMLR.

Sarikaya, Ruhi, Sameer Maskey, R. Zhang, Ea-Ee Jan, D. Wang, Bhuvana Ramabhadran, and Salim Roukos. 2009. Iterative sentence-pair extraction from quasi-parallel corpora for machine translation. In *Proceedings of INTERSPEECH 2009*, pages 432–435, Brighton, United Kingdom.

Scarton, Carolina and Lucia Specia. 2016. A Reading Comprehension Corpus for Machine Translation Evaluation. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3652–3658, Portorož, Slovenia. European Language Resources Association (ELRA).

Schwenk, Holger. 2018. Filtering and Mining Parallel Data in a Joint Multilingual Space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, Melbourne, Australia. Association for Computational Linguistics.

Schwenk, Holger, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135M Parallel Sentences in 1620 Language Pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361, Online. Association for Computational Linguistics.

Schwenk, Holger and Matthijs Douze. 2017. 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. Association for Computational Linguistics.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016a. Improving Neural Machine Translation Models with Monolingual Data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Sennrich, Rico, Barry Haddow, and Alexandra Birch. 2016b. Neural Machine Translation of Rare Words with Subword Units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sennrich, Rico and Martin Volk. 2010. MT-based Sentence Alignment for OCR-generated Parallel Texts. In *Proceedings of the 9th Conference of the Association for Machine Translation in the Americas: Research Papers*, Denver, Colorado. Association for Machine Translation in the Americas.

Sennrich, Rico and Martin Volk. 2011. Iterative, MT-based Sentence Alignment of Parallel Texts. In *Proceedings of the 18th Nordic Conference of Computational Linguistics*, pages 175–182, Riga, Latvia. Northern European Association for Language Technology (NEALT).

Sennrich, Rico and Biao Zhang. 2019. Revisiting Low-Resource Neural Machine Translation: A Case Study. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 211–221, Florence, Italy. Association for Computational Linguistics.

Shi, Haoyue, Luke Zettlemoyer, and Sida I. Wang. 2021. Bilingual Lexicon Induction via Unsupervised Bitext Construction and Word Alignment. 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 813–826, Online. Association for Computational Linguistics.

Simard, Michel. 1999. Text-Translation Alignment: Three Languages Are Better Than Two. In *1999 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 2–11, College Park, Maryland. Association for Computational Linguistics.

Simard, Michel, George F. Foster, and Pierre Isabelle. 1992. Using Cognates to Align Sentences in Bilingual Corpora. In *Proceedings of the Fourth Conference on Theoretical and Methodological Issues in Machine Translation of Natural Languages*, pages 67–81, Montreal, Canada.

Símonarson, Haukur Barri and Vésteinn Snæbjarnarson. 2021. Icelandic Parallel Abstracts Corpus. *ArXiv*, abs/2108.05289.

Símonarson, Haukur Barri, Vésteinn Snæbjarnarson, Pétur Orri Ragnarson, Haukur Jónsson, and Vilhjalmur Thorsteinsson. 2021. Miðeind's WMT 2021 Submission. In *Proceedings of the Sixth Conference on Machine Translation*, pages 136–139, Online. Association for Computational Linguistics.

Snover, Matthew, Bonnie Dorr, Rich Schwartz, Linnea Micciulla, and John Makhoul. 2006. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of the 7th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 223–231, Cambridge, Massachusetts, USA. Association for Machine Translation in the Americas.

Steingrímsson, Steinþór. 2021. Icelandic-English test set for sentence alignment 21.10. CLARIN-IS.

Steingrímsson, Steinþór and Starkaður Barkarson. 2021. ParIce: English-Icelandic parallel corpus (21.10). CLARIN-IS.

Steingrímsson, Steinþór, Sigrún Helgadóttir, and Eiríkur Rögnvaldsson. 2015. Analysing Inconsistencies and Errors in PoS Tagging in two Icelandic Gold Standards. In *Proceedings of the* 20^{th} *Nordic Conference of Computational Linguistics*, NODALIDA 2015, pages 287–291, Vilnius, Lithuania. Linköping University Electronic Press.

Steingrímsson, Steinþór, Sigrún Helgadóttir, Eiríkur Rögnvaldsson, Starkaður Barkarson, and Jón Guðnason. 2018. Risamálheild: A Very Large Icelandic Text Corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation*, LREC 2018, pages 4361–4366, Miyazaki, Japan.

Steingrímsson, Steinþór, Örvar Kárason, and Hrafn Loftsson. 2019. Augmenting a BiLSTM Tagger with a Morphological Lexicon and a Lexical Category Identification Step. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1161–1168, Varna, Bulgaria. INCOMA Ltd.

Steingrímsson, Steinþór, Hrafn Loftsson, and Andy Way. 2021a. CombAlign: a Tool for Obtaining High-Quality Word Alignments. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 64–73, Reykjavik, Iceland (Online). Linköping University Electronic Press, Sweden.

Steingrímsson, Steinþór, Hrafn Loftsson, and Andy Way. 2023. Filtering Matters: Experiments in Filtering Training Sets for Machine Translation. In *Proceedings of the 24th Nordic Conference on Computational Linguistics (NoDaLiDa)*, Torshavn, Faroe Islands.

Steingrímsson, Steinþór, Pintu Lohar, Hrafn Loftsson, and Andy Way. 2021b. Effective Bitext Extraction From Comparable Corpora Using a Combination of Three Different Approaches. In *Proceedings of the 14th Workshop on Building and Using Comparable Corpora (BUCC 2021)*, pages 8–17, Online (Virtual Mode). INCOMA Ltd.

Steingrímsson, Steinþór, Luke O'Brien, Finnur Ingimundarson, Hrafn Loftsson, and Andy Way. 2022. Compiling a Highly Accurate Bilingual Lexicon by Combining Different Approaches. In *Proceedings of Globalex Workshop on Linked Lexicography within the 13th Language Resources and Evaluation Conference*, pages 32–41, Marseille, France. European Language Resources Association.

Steingrímsson, Steinþór, Hrafn Loftsson, and Andy Way. 2021c. Pivotalign: Leveraging High-Precision Word Alignments for Bilingual Dictionary Inference. In *Proceedings of the Workshops and Tutorials held at LDK 2021*, pages 190–199, Zaragoza, Spain. CEUR-WS.

Stymne, Sara, Christian Hardmeier, Jörg Tiedemann, and Joakim Nivre. 2013. Tunable Distortion Limits and Corpus Cleaning for SMT. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 225–231, Sofia, Bulgaria. Association for Computational Linguistics.

Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA. MIT Press.

Taghipour, Kaveh, Shahram Khadivi, and Jia Xu. 2011. Parallel Corpus Refinement as an Outlier Detection Algorithm. In *Proceedings of MT Summit XIII*, pages 414–421, Xiamen, China.

Tanaka, Kumiko and Kyoji Umemura. 1994. Construction of a Bilingual Dictionary Intermediated by a Third Language. In *COLING 1994 Volume 1: The 15th International Conference on Computational Linguistics*, pages 297–303, Kyoto, Japan.

Tang, Yuqing, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2021. Multilingual Translation from Denoising Pre-Training. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3450–3466, Online. Association for Computational Linguistics.

Thompson, Brian and Philipp Koehn. 2019. Vecalign: Improved Sentence Alignment in Linear Time and Space. 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 1342–1348, Hong Kong, China. Association for Computational Linguistics.

Tiedemann, Jörg. 2012. Parallel Data, Tools and Interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Tiedemann, Jörg. 2016. Finding Alternative Translations in a Large Corpus of Movie Subtitle. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3518–3522, Portorož, Slovenia. European Language Resources Association (ELRA).

Tiedemann, Jörg and Santhosh Thottingal. 2020. OPUS-MT – Building open translation services for the World. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 479–480, Lisboa, Portugal. European Association for Machine Translation.

Toral, Antonio, Sheila Castilho, Ke Hu, and Andy Way. 2018. Attaining the Unattainable? Reassessing Claims of Human Parity in Neural Machine Translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 113–123, Brussels, Belgium. Association for Computational Linguistics.

Tran, Chau, Shruti Bhosale, James Cross, Philipp Koehn, Sergey Edunov, and Angela Fan. 2021. Facebook AI's WMT21 News Translation Task Submission. In *Proceedings of the Sixth Conference on Machine Translation*, pages 205–215, Online. Association for Computational Linguistics.

Tschorn, Patrick and Anke Lüdeling. 2003. Morphological knowledge and alignment of English-German parallel corpora. In *Proceedings of the Corpus Linguistics 2003 conference*, pages 818–827, Lancaster, UK.

Úlfarsdóttir, Þórdís. 2014. ISLEX — a Multilingual Web Dictionary. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC 2014)*, pages 2820–2825, Reykjavik, Iceland. European Language Resources Association (ELRA).

Vamvas, Jannis and Rico Sennrich. 2022. NMTScore: A multilingual analysis of translation-based text similarity measures. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 198–213, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Varga, Dániel, Péter Halácsy, András Kornai, Nagy Viktor, Nagy László, Németh László, and Tron Viktor. 2005. Parallel corpora for medium density languages. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2005)*, pages 590–596, Borovets, Bulgaria. INCOMA Ltd.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pages 5999–6009, Long Beach, California.

Viera, Anthony J. and Joanne M. Garrett. 2005. Understanding Interobserver Agreement: The Kappa Statistic. *Family Medicine*, 37.5:360–363.

Vilar, David, Gregor Leusch, Hermann Ney, and Rafael E. Banchs. 2007. Human Evaluation of Machine Translation Through Binary System Comparisons. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 96–103, Prague, Czech Republic. Association for Computational Linguistics.

Volk, Martin, Noah Bubenhofer, Adrian Althaus, Maya Bangerter, Lenz Furrer, and Beni Ruef. 2010. Challenges in Building a Multilingual Alpine Heritage Corpus. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, pages 1653–1659, Valletta, Malta. European Language Resources Association (ELRA).

Vulić, Ivan and Marie-Francine Moens. 2012. Sub-corpora Sampling with an Application to Bilingual Lexicon Extraction. In *Proceedings of COLING 2012*, pages 2721–2738, Mumbai, India. The COLING 2012 Organizing Committee.

Wagner, Robert A. and Michael J. Fischer. 1974. The String-to-String Correction Problem. *Journal of the Association for Computing Machinery*, 21(1):168–173.

Way, Andy. 2018. Quality Expectations of Machine Translation. In Joss Moorkens, Sheila Castilho, Federico Gaspari, and Stephen Doherty, editors, *Translation Quality Assessment: From Principles to Practice*, pages 159–178. Springer International Publishing.

White, John S., Theresa A. O'Connell, and Francis E. O'Mara. 1994. The ARPA MT Evaluation Methodologies: Evolution, Lessons, and Future Approaches. In *Proceedings of the First Conference of the Association for Machine Translation in the Americas*, pages 193–205, Columbia, Maryland, USA. Association for Machine Translation in the Americas.

Wolk, Krzysztof, Emilia Rejmund, and Krzysztof Marasek. 2016. Multi-domain machine translation enhancements by parallel data extraction from comparable corpora. In Ewa Gruszczyńska and Agnieszka Leńko-Szymańska, editors, *Polish-Language Parallel Corpora*, pages 157–179. Instytut Lingwistyki Stosowanej, Warsaw, Poland.

Wu, Dekai. 1994. Aligning a Parallel English-Chinese Corpus Statistically With Lexical Criteria. In *32nd Annual Meeting of the Association for Computational Linguistics*, pages 80–87, Las Cruces, New Mexico, USA. Association for Computational Linguistics.

Xu, Hainan and Philipp Koehn. 2017. Zipporah: a Fast and Scalable Data Cleaning System for Noisy Web-Crawled Parallel Corpora. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2945–2950, Copenhagen, Denmark. Association for Computational Linguistics.

Xu, Runxin, Zhuo Zhi, Jun Cao, Mingxuan Wang, and Lei Li. 2020. Volctrans Parallel Corpus Filtering System for WMT 2020. In *Proceedings of the Fifth Conference on Machine Translation*, pages 985–990, Online. Association for Computational Linguistics.

Yang, Yinfei, Gustavo Hernandez Abrego, Steve Yuan, Mandy Guo, Qinlan Shen, Daniel Cer, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Improving Multilingual Sentence Embedding using Bi-directional Dual Encoder with Additive Margin Softmax. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 5370–5378, Macao, China. International Joint Conferences on Artificial Intelligence Organization.

Yusuf, Mirza, Praatibh Surana, Gauri Gupta, and Krithika Ramesh. 2021. Curb Your Carbon Emissions: Benchmarking Carbon Emissions in Machine Translation. *ArXiv*, abs/2109.12584.

Zaragoza-Bernabeu, Jaume, Gema Ramírez-Sánchez, Marta Bañón, and Sergio Ortiz Rojas. 2022. Bicleaner AI: Bicleaner Goes Neural. In *Proceedings of the Language Resources and Evaluation Conference*, pages 824–831, Marseille, France. European Language Resources Association.

Zariņa, Ieva, Pēteris Ņikiforovs, and Raivis Skadiņš. 2015. Word Alignment Based Parallel Corpora Evaluation and Cleaning Using Machine Learning Techniques. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, pages 185–192, Antalya, Turkey. European Association for Machine Translation.

Zhang, Wu. 2022. Improve Sentence Alignment by Divide-and-conquer. *ArXiv*, abs/2201.06907.

Zhao, Bing and Stephan Vogel. 2002. Adaptive Parallel Sentences Mining from Web Bilingual News Collection. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, ICDM '02, pages 745–748, Maebashi City, Japan. IEEE.

Zweigenbaum, Pierre, Serge Sharoff, and Reinhard Rapp. 2016. Towards Preparation of the Second BUCC Shared Task: Detecting Parallel Sentences in Comparable Corpora. In *Proceedings of the Ninth Workshop on Building and Using Comparable Corpora*, pages 38–43, Portorož, Slovenia. ELDA.

Zweigenbaum, Pierre, Serge Sharoff, and Reinhard Rapp. 2017. 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. Association for Computational Linguistics.

Zweigenbaum, Pierre, Serge Sharoff, and Reinhard Rapp. 2018. Overview of the Third BUCC Shared Task: Spotting Parallel Sentences in Comparable Corpora. In *Proceedings of the 11th Workshop on Building and Using Comparable Corpora at LREC 2018*, pages 39–42, Miyazaki, Japan. European Language Resources Association (ELRA).