Technology
Arts Sciences
TH Köln

Gender Bias in Machine Translation – A Comparison of Systems Trained with Biased and Unbiased Datasets

Master's Thesis
to obtain the *Master of Arts* in Specialised Translation
at the Institute of Translation and Multilingual Communication
at the Cologne University of Applied Sciences

vorgelegt von: Janiça Hackenbuchner
Matrikel-Nr.: 11136410
Adresse: Borsteler Chaussee 72
22453 Hamburg
janica.hackenbuchner@th-koeln.de

eingereicht bei: Prof. Dr. Ralph Krüger
Zweitgutachter/in: Prof. Dipl.- Übers. Peter Lammers

Hamburg, 11.02.2022

Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbstständig verfasst zu haben. Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer oder der Verfasserin/des Verfassers selbst entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben. Die Arbeit hat mit gleichem Inhalt bzw. in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Hamburg, 11.02.2022	Janica Hackenbuchner
Ort, Datum	Rechtsverbindliche Unterschrift

Acknowledgements

This dissertation was inspired and made possible by my supervisor, Professor Dr. Ralph Krüger, whose continued guidance supported me on this research path, from ideation to realisation. The experimental part of this research was highly spurred by the insightful feedback and voluntary support of research scientist Stefan Miedzianowski. A special thanks goes to my fellow translating student Franziska Brachthäuser and to Kim Hackenbuchner for being my proof-readers.

Abstract

Machine translations reflect biases inherently present in datasets used to train machine translation (MT) systems. Bias in machine translations not only offends individuals directly affected but also spreads and increases bias in society. MT systems need to be trained on large parallel corpora with millions of sentences. These datasets are generated by society and therefore contain stereotypes and biases, such as gender-bias and racial bias, inherently present in society. Online available commercial MT systems are widely used in industry and by individuals, and not only by translators who studied how to post-edit machine translations and correct potential biases. The following dissertation sheds light on this issue by analysing and comparing translations produced by Neural Machine Translation (NMT) systems trained on datasets exhibiting a different degree of gender-bias. Findings clearly demonstrate that the more biased a MT training dataset is, the more biased a resulting translation will be. Similarly, if a MT training dataset exhibits less gender-bias, the resulting translations will noticeably reflect less gender-bias. The results of this dissertation highlight the need for an approach to neutralise biased datasets to produce gender-fair translations. Gender-fair NMT systems will help fight gender-bias in machine translated texts, positively influence individuals reading such MT-texts, and create a more gender-fair society.

Key words: translation, machine translation, neural machine translation, gender bias, natural language processing

Table of content

List of Tables

List of Figures

Introduction

The field of translation is rapidly evolving due to the introduction and application of machine translation spurred by developments in artificial intelligence (AI) and online available commercial machine translation MT systems. AI-aided MT majorly accelerates the speed at which one language can be translated into another language. But the way this is done needs to be constantly developed and evaluated. With Deep Learning being a quickly evolving field (Klein et al., 2017: 70), MT is one of the fastest evolving AI fields and is taking the field of translation by storm. As in any other AI-based field, the applications feed and rely on enormous amounts of data (minimum millions of sentences). Based on the data chosen to be an input of AI-machines, these AI-machines will predict certain, most probable, outcomes. Therefore, the type of data that is fed into AI-machines directly determines the outcome predictions.

In machine translation this is specifically relevant since a MT-system can only predict translation outputs based on the vocabulary it has learned from the data the system has been trained with. A MT-system needs to be trained on huge amounts of data per language pair (Way, 2019: 320), preferably in addition to monolingual language datasets, to provide adequate translation outputs. Today, online available commercial MT systems have mastered this task to an astonishing extent, providing moderately fluent translation outputs for a select number of language combinations. However, as rapidly as the field of MT is advancing, there continue to be flaws in MT-outputs that do not occur in human translation. Flaws that stem from the data with which MT engines are being trained, including bias. There are different types of bias including racial or gender bias present in MT. This research specifically focusses on binary gender bias for machine translation from English into German.

To translators, and particularly translation students, but increasingly also to the wider population working with MT, bias in MT-outputs is an annoyance and not seldomly also a very personal matter. Germans are particularly aware of gender bias as German, like Spanish and French, is a gendered language. Particularly in the last few years, the topic of gender bias in everyday life speech, academic texts, books and on the news in Germany has become very debated with increased awareness when it comes to written and spoken language that includes all genders. For that reason, it is even more important to adequately update MT-outputs appropriately to not contain gender bias.

This dissertation is spurred by personal experience of reading gender bias in MT-outputs when completing translation tasks and is influenced by existing research on the topic of gender bias in MT that present possible solutions to solve the issue. Current research and studies in this field will be referred to in this dissertation. The beauty of this research is that it combines two disciplines – computational sciences and translation studies. It is an interdisciplinary approach to tackle an issue that affects the wider population. It is a particularly relevant field of research, since "[m]ost people acknowledge that MT is one of the hardest problems we are trying to address in computer science." (Way, 2019: 324). The aim of this dissertation is to add to the research and to raise awareness about the issue of gender bias in MT by making it comprehensible didactically and understandable so that the written code, a major part of this dissertation, can be run by anyone to provide a deeper understanding of the implications of how datasets, with which MT-engines are trained, influence MT-outputs. The following dissertation should be accessible to a wider audience in line with the following statement:

> “[t]here are currently broader efforts to make the workings of neural MT accessible to a wider audience beyond computational linguistics and language-oriented AI research. For the goal of adequate machine translation literacy in the field of translation/specialized communication, these efforts should be beneficial.” (Krüger, 2021:43, *my own EN translation*)

The methodology for this research will be outlined in detail. To raise awareness of how datasets influence MT-outputs, NMT systems were trained with differently biased datasets and the results were analysed and compared with a specific focus on gender bias in the translation outputs. The research was conducted on the online platform Google Colab Pro by using the open-source toolkit OpenNMT-py (Klein et al., 2017). The exact methodology will be outlined in section 4. The translation outputs produced with the differently biased datasets will be analysed using different automatic evaluation metrics including sacreBLEU, COMET and BERTScore.

The hypothesis is that if a NMT system is trained on a more gender-biased dataset, the translation outputs will reflect this gender-bias. Whereas if a NMT system is trained on a less biased dataset, the translation outputs will reflect less gender-bias. The amount of bias present in the output translations directly correlate with the amount of bias present in the datasets. The hypothetical conclusion to draw be drawn is that bias in training data must be reduced to equally reduce bias in machine translated outputs and simultaneously reduce bias in society. The result of this research highlights the necessity for unbiased or balanced datasets to train NMT systems.

As an addition to the python code with individually trained NMT models and the translation outputs, a classification of linguistic phenomena (language patterns) of trigger words and pointers that currently lead to gender bias in MT are highlighted and discussed in section 3 of this dissertation. Such linguistic phenomena and trigger words include nouns/professions, adjectives, and the context of words in sentences.

Section 1 will outline the background to machine translation, specifically neural machine translation, and how it works. Section 2 provides an overview of gender bias in machine translation and methods to eliminate it as presented by current research. Section 3 outlines a classification of linguistic phenomena and its biases found in current MT sources. Section 4 presents the experimental setup, including data preparation and training the NMT system. Section 5 presents an analysis and discusses results of the research. Section 6 outlines limitations of this research and possibilities for further research. The research and dissertation are concluded in section 7.

1 Machine Translation

1.1 Translation

The field of translation is rapidly evolving as a result of the introduction of machine translation. The work of translators has been irreversibly shaped by and largely impacted by MT systems. It is continuously becoming more difficult to leave this technology out of translation and technical communication science theory as well as model building (Krüger, 2021: 2).

Machine translation is widely used in industry but also in translation research and by translators. Given that the quality of MT is rapidly improving, the translation community has been wondering about the effects this will have on the field of translation and the profession as a translator. One substantive development has been that the tasks of a translator have progressed to include Post-Editing, referring to a translator not creating a human-translation on his or her own but rather revising and adapting a machine-translated text (Krüger, 2019: 56). Human translators need to accept and embrace the inclusion of advanced technology in their field since MT is one of the fastest developing fields of AI and its guaranteed to continue playing an increasingly important role in translation. Way (2019) believes that the

> “human-in-the-loop will always remain the most important link in the chain […]. All that MT system developers are trying to do is improve the output from their systems to make technology-savvy translators more productive. MT systems are unlikely ever to ‘bridge the gap’ or achieve human parity with human quality translation.” (326).

With this development in mind, translators need to adapt their ways of working and embrace the rapidly developing technology, as is required in many fields. The positive side-effect of MT is the increased attention on the field of translation in industry and in research.

1.2 The Development of (Neural) Machine Translation

In this section, the principles of Machine Translation (MT) and specifically Neural Machine Translation (NMT) will be presented and explained. The research this dissertation is based on was conducted by training an NMT engine. This section aims to provide a short historical and theoretical background to MT, specifically NMT, to both provide an understanding of what role MT/NMT plays in the field of translation and also give a simple theoretical explanation of how a NMT system works, i.e., what happens in the background, when an NMT engine is trained to then provide a translated output.

In recent years there has been a noticeable increase in the digitisation and datafication of work processes in many industries including translation due to rapid advancements in artificial intelligence (AI), with MT developing alongside neural network-based machine learning (ML) techniques (Jooste, Haque and Way, 2021: 290). The performance of machine learning systems has improved significantly in recent years (Romanov et al., 2019: 4187). Advances in digitisation and datafication include AI-assisted machine processing of natural language and the creation and potential availability of large digital translation datasets (Krüger, 2021). Large enough bilingual datasets for the training of MT systems, however, remain a rare occurrence and therefore pose an obstacle in the research and continued development of MT systems (Way, 2019: 320), as highlighted and discussed in this dissertation. Aside from the current lack of large bilingual datasets for training purposes, today NMT plays an increasingly important role in professional technical translation (Krüger, 2021: 1) thanks to overall advances in the field of AI.

The predecessor of today's most widely used Machine Translation system, Neural Machine Translation, was phrase-based statistical machine translation (PBSMT), often referred to simply as SMT (Forcada, 2017: 291). Even earlier MT systems were rule-based systems (RBMT) but corpus-based MT systems, such as SMT and today NMT, quickly became dominant, and from the mid-1990s until recently, the corpus-based SMT had been the dominant paradigm for MT (Way, 2019: 311). SMT, however, provided rather word-for-word translations due to such systems being trained by a word-for-word input to deliver a word-for-word translation output. SMT systems are not able to place a word in context by paying attention to its surrounding words (the sentence context). Once it seemed that SMT could no longer be noticeably improved and successes in MT came to a halt, that void was filled with NMT, which is based on artificial neural networks. Astonishingly, the entire research field of MT went neural within just two years (Koehn, 2020: 10).

Phrase-based SMT models continue to be more effective on small data samples, whereas NMT models are more effective on large data samples, and when handling unknown and low frequency words, NMT systems outperform SMT systems (Jooste, Haque and Way, 2021: 296). In research conducted comparing SMT and NMT translation outputs for English to German, the NMT result led to a decreased post-editing effort of 25% (Way, 2019: 317). In comparison to the latest PBSMT, NMT produces significantly fewer errors: -19% morphological errors, -17% lexical errors, -50% fewer word order errors with almost 70% fewer misplacements of verbs and around 50% fewer misplacements of nouns (Way, 2018: 171). In MT research, and increasingly in industrial pipelines, NMT is considered the state-of-the art (Jooste, Haque and Way, 2021: 290). In comparison to SMT, NMT can generalise better by exploiting word embedded similarities and can condition on larger context (on the entire input as well as previous output words) (Koehn, 2017: 106).

Just like its predecessor, NMT is a corpus-based MT and must be trained on huge bilingual corpora, meaning datasets of a source language and its equivalent translated output. To train an NMT system, even larger corpora are required than to train an SMT system (Way, 2019: 320). Several remaining disadvantages for training NMT systems include the requirement of large datasets, the necessary training times for NMT engines being much longer than for SMT, and the necessity of expensive hardware – GPUs (graphics processing units) where thousands of CPUs (central processing units) perform calculations in parallel – to train NMT engines since the number of calculations needed to train an NMT model is astronomical (ibid.: 324). Most problematic today is that only a handful of such large datasets necessary to train NMT engines exist openly, limiting (academic) research in this field.

1.3 Neural Networks and Neural Language Models

NMT is based on neural networks, which are powerful in modelling conditional probability distributions with a number of inputs, being robust to unseen or out-of-domain data (vocabulary or context not present in the training data) (Koehn, 2020: 37). Neural networks are sets of connected neurons – in an NMT system these are thousands of artificial units resembling neurons – defined by their behaviour, or activation state. This activation state refers to the degree to which these neurons are excited or inhibited, depending on what stimuli they receive from other neurons and how strong this connection is over a certain distance (Forcada, 2017: 292).

In SMT, linear models as a network were essential, but for NMT, neural networks rely on multiple layers including a hidden layer (Koehn, 2020: 11–12). Input values do not directly lead to output values, as in SMT, but rather a hidden layer is placed between them, where the mechanism connecting the input and output values cannot be seen; it is *hidden* (ibid.: 12). Computationally, these values are 'nodes'. Weighted input nodes are linearly combined to produce the hidden node values, and are then linearly combined to produce each output node value (ibid.: 12). For neural networks, the term 'deep learning' comes into play, where layers of hidden nodes can be stacked together 'deeply' (ibid.: 14), which improves the neural network's performance.

A neural network predicts the (most likely) correct output based on a certain input. In order to do this, the weight values need to be optimised (Koehn, 2020: 16). Thus, during training of a neural network, the input is repeatedly fed into the network, and the computed output from the network is then compared with the correct output of the training example. The weights are updated accordingly. The process repeats multiple times, and each loop is called an 'epoch' (ibid.: 16). The most common training method, also used in NMT, is called 'back-propagation', where the weights are updated to the output layer and error information is propagated back to the previous layers (ibid.: 16).

When training a neural network, such as an NMT system, the error on the training data continuously decreases as the network learns and improves. At one point, however, the network can simply memorise the data. This is termed 'over-fitting' and is something that wants to be avoided entirely (Koehn, 2020: 23). To avoid over-fitting, a validation set is used to validate a neural network alongside the training set. The phenomenon of over-fitting and validating is exemplified in figure 1 below.

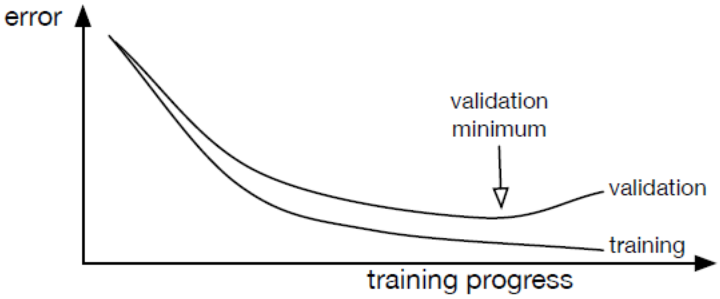


Figure 1. Training progress of a neural network over time (Koehn, 2020: 23)

The validation set is a different dataset than the training set. As the error continuously decreases on the training set, it will at one point increase on the validation set. This increase is over-fitting and wants to be avoided. Therefore, when a minimum on the validation has been reached (where the lines in figure 1 converge before diverging again), training will be stopped. This prevents over-fitting and thus the memorisation of vocabulary.

For some training sets, average values at one layer may become too large, while for other training sets, average values may become too small (Koehn, 2020: 25–26). Values that are too small or too large would feed into the following layer and produce values that are equally too small or too large – this is detrimental to train neural networks. To counter this, a layer normalisation is added. Layer normalisation 'normalises' values in each layer (ibid.: 26). This is also done in NMT.

Each word in a neural network is represented as a high-dimensional vector, having one dimension (Koehn, 2020: 38). Referring back to the multiple layers mentioned above, another layer is now introduced between the input layer and the hidden layer. Each source word in this layer is projected into a lower dimensional space. This projection leads to a continuous space representation for each source word, no matter its position in the sentence context (ibid.: 38). These vector representations are deep, which is where the term *deep learning* comes into play (Forcada, 2017: 295), and are known as 'word embeddings', a crucial aspect in NMT (Koehn, 2020: 38).

Word embeddings enable the generalisation between words, clustering words in a similar context together. Large amounts of data are necessary to train word embeddings to learn associations and relationships about words in the text data (Bolukbasi et al., 2016: 3). The clustering of semantically similar words is a powerful aspect in NMT, and as John Rupert Firth once said: "You shall know a word by the company it keeps" (Koehn, 2020: 41). Word embeddings are concatenated as input to a feed-forward layer and then mapped to the hidden layer in the model (Koehn, 2020: 39). The output layer is a "probability distribution over words" (ibid.: 40), and to ensure that this probability distribution is correct, a (mathematical) 'softmax' activation function is used, where all values end up adding up to 1.

Another important aspect in NMT is a ‘recurrent neural network’ (RNN) (ibid.: 44). In RNNs, each word is learned based on the previous word’s prediction: the neurons are filled with values from the hidden layer of the prediction for the previous word (ibid.: 45). The neurons ‘encode’ the previous sentence context and learn from the previous words (the context) in the sentence. To encode the source text, an RNN maps each source word to a vector, and then processes these vectors to a sequence of hidden word vectors (Klein et al., 2017: 68), where a vector is a word embedding. For longer sentences, this method gets a little tricky, which is where ‘back-propagation through time’ comes into play. In back-propagation, the RNN is fixed over a certain number of steps, for example over a number of five words (Koehn, 2020: 45).

In some cases, the importance of words in a sentence decreases along the length of the sentence. For example, “The *woman* bought a beautiful house in a little neighbourhood *she* grew up in”, where ‘she’ refers back to ‘the woman’ at the beginning of the sentence. In this sentence, words like ‘neighbourhood’ and ‘house’ are less relevant to 'she' than 'the woman'. RNNs focus on the most recent context before a certain word. This can be very useful, but at times very problematic. If a word at the end of a sentence directly relates to a word at the beginning of a sentence, like in the example sentence, this needs to be considered. The long short-term memory (LSTM) architecture addresses this issue (Koehn, 2020: 47). LSTMs are regulated by gate-parameters as outlined in Koehn, 2020: 47,

1. The *input gate* parameter regulates how much new input changes the memory state.
2. The *forget gate* parameter regulates how much of the prior memory state is retained or forgotten).
3. The *output gate* parameter regulates how strongly the memory state is passed on to the next layer.

Just like RNNs, LSTMs are trained with back-propagation through time. An alternative to LSTMs are gated recurrent units (GRUs) (ibid.: 49). However, many NMT systems are based on LSTMs, just like the one used for the research in this dissertation.

1.4 Neural Machine Translation Architectures

Based on conditional language modelling, NMT builds and trains a single, large neural network to produce a translation by modelling the probability of a target sentence based on a given source sentence (Bahdanau, Cho and Bengio, 2014: 2; Klein et al., 2017: 68). Both in academics and in industry, NMT has become the dominant MT system and has quickly increased MT standards reached by SMT (Way, 2019: 311–312). NMT is based on artificial neural networks, which, as is the case with other known AI-based neural networks, “stimulate the structure of the human brain consisting of billions of information-processing cells (neurons) and interconnections between these cells” (Krüger, 2020b: 263).

In an NMT system words, sub-word units such as characters, or character sequences are processed in a parallel, distributed way, where a large set of neurons composed of individual neurons and their individual activation states are trained to build *distributed representations* (Forcada, 2017: 293). These distributions are estimated using an attention-based encoder-decoder architecture (Bahdanau, Cho and Bengio, 2014: 2). NMT systems are frequently based on an encoder-decoder framework extended with an attention model (Way, 2019: 316). NMT architectures are outlined in the following sections.

1.4.1 Encoder-Decoder Framework

NMT frameworks belong to a family of encoder-decoders for each language, "jointly trained to maximise the probability of a correct translation given a source sentence" (Bahdanau, Cho and Bengio, 2014: 1). This NMT encoding-decoding architecture is also known as the 'seq2seq' (sequence to sequence) architecture (Forcada, 2017: 299) and refers to the system sequentially encoding the source text to make it computer-readable, processing that information, and then decoding it into a target text.

As outlined in section 1.3, the NMT system encodes the source words or short phrases known as word-embeddings in the form of vectors (Way, 2019: 317). This vector information is then processed by the system and finally decoded into an output text. A NMT engine trains on millions of words to produce a translated output for a source text. Traditionally, the source sentence was compressed into a fixed-length vector, which lowered performance for longer input sentences. Bahdanau, Cho and Bengio (2014) extended this basic encoder-decoder framework by allowing the model to align and translate jointly, with a bidirectional RNN as encoder and decoder and the implementation of a mechanism of attention in the decoder (4). With this mechanism, the decoder decides to pay closer 'attention' to parts of the source sentence, as further explained in section 1.4.2. The attention mechanism prevents the encoder from needing to encode all source information into a fixed-length vector and thus facilitates encoding longer sentences (ibid., 2014: 9). The bidirectional RNNs combined with an attention-mechanism leads to the recurrent-attention model frequently used in NMT today. A rough outline of this architecture is depicted in figure 2 below.

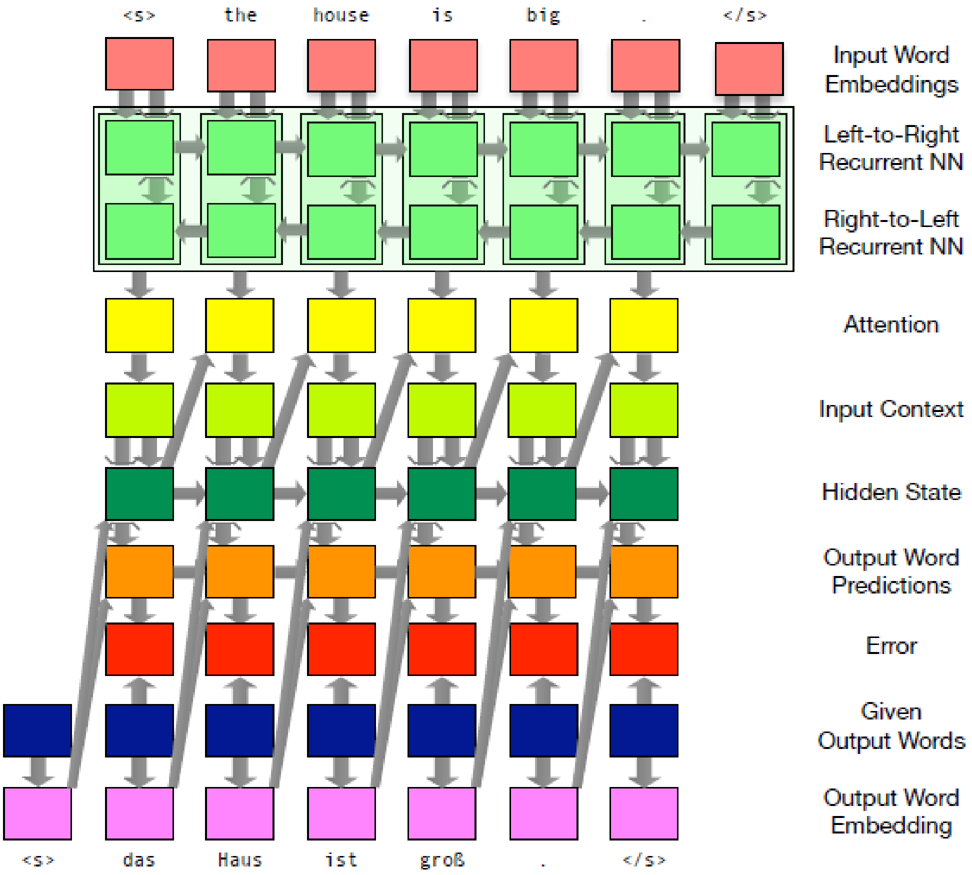


Figure 2. Encoder-decoder framework extended with an attention model (Koehn, 2020: 59)

This figure depicts how an English source sentence “the house is big.” is machine translated through an encoder-decoder architecture with attention into a German target sentence “das Haus ist groß.”, where <s> and </s> respectively depict the start and end of the sentence. First, the word embeddings are being input, then the bidirectional RNN is applied, followed by the attention mechanism, the information is then sent through the hidden states, and output predictions are being proposed until a correct (most likely predicted) output is delivered as a word embedding.

The aim of training an NMT system is to achieve translation outputs as close as possible to the corresponding reference or ‘gold-standard’ translation, which is ideally produced by specialised human translators (Forcada, 2017: 295). During training, the weights are modified to lower the value of the ‘loss function’ of how far the machine translation output is away from said reference sentence (ibid.: 295–296). The original NMT architecture is based on RNNs, referring to the neural network being repeatedly applied and at each step, the computed output is fed back to the next step (Forcada, 2017: 297). The neural networks in an NMT engine are thus trained by “repeatedly feeding training examples into the network and updating the weights according to the outputs” (Jooste, Haque and Way, 2021: 291). The RNNs implemented today are based on soft-alignment rather than hard-alignment, which allows the model to accurately encode the parts of the input sentence that surround a particular word (Bahdanau, Cho and Bengio, 2014: 7). Since soft-alignment naturally deals with source and target phrases of different lengths, it can be used well to translate longer sentences (ibid., 2014: 7–8).

As defined in section 1.3, encoders in the encoder-decoder architecture arrange their layers in so-called ‘gating structures’ predominantly known as LSTMs or gated recurrent units (GRU) to learn to either forget irrelevant past inputs or remember relevant past inputs (Forcada, 2017: 297). Only one year after introducing NMT, NMTs based on RNNs with LSTM units already achieved translation outputs almost of equal quality as the previously conventional PBSMTs (Bahdanau, Cho and Bengio, 2014). The research conducted for this dissertation was done by training an NMT engine based on an RNN-model with LSTM units and attention.

Below, the functionality of an encoder-decoder architecture is outlined for an example translation of an English sentence “He is going for a walk.” into German.

Encoding

- The vector embeddings of each individual word together e(“he”), e(“is”), e(“going”), e(“for”), e(“a”), e(“walk”) and e(“.”) shape the representation of the sentence.
- These embeddings will be encoded into an empty sentence E(“”): E(“He”), E(“is”), E(“going”), E(“for”), E(“a”), E(“walk”) and E(“.”)
- These representations are then combined embedding for embedding by the encoder network until the entire sentence is completed: E(“He is”), E(“He is going”), E(“He is going for”), E(“He is going for a”), E(“He is going for a walk”), and finally E(“He is going for a walk.”)

To decode vector representations, an RNN hidden representation of previously generated words is combined with source hidden vectors to then predict scores for every possible word outcome (Klein et al., 2017: 68).

Decoding

- The system produces two vectors: 1. An initial decoder state D(“He is going for a walk.”, “”) where the last “” represent a still empty sequence of target words, 2. A vector of probabilities for all possible translations per word of the target sentence p(x|”He is going for a walk”, “”)
- For each encoded word in the representation, the decoder will produce the most likely translated word. So next we would get: 1. The decoder state D(“He is going for a walk.”, “Er”), 2. The vector of probabilities p(x|”He is going for a walk”, ”Er”)
- Just like in the encoder stage, the decoder works in recurrent steps adding each newly translated word: 1. D(“He is going for a walk.”, “Er geht”), and 2. p(x|”He is going for a walk”, ”Er geht”) until we get 1. D(“He is going for a walk.”, “Er geht

spazieren”), and 2. p(x|”He is going for a walk”, ”Er geht spazieren”). The final translated output would then be “Er geht spazieren.” and the end-of-decoding marker has been reached.
Note: this is just one translation example, “Er macht einen Spaziergang” could also be an option and it would of course depend on the translation system to produce either possibility.

To predict a next-word distribution, a softmax layer is applied (Klein et al., 2017: 68). Each source word is weighted relative to its expected contribution to the target prediction using an attention pooling layer, which in turn is influenced by the source hidden vectors (ibid.: 68). A complete NMT model is then trained end-to-end to predict a most probable translation outcome given a certain source text. This is outlined in more detail in the following sections.

1.4.2 Attention Mechanism

In SMT systems, a source sentence is never translated *en bloc* but rather using only lexical and phrasal chunks (Way, 2019: 317). In comparison, the NMT architecture is enriched with an attention model, resembling the word and phrase alignment in SMT, where ‘attention’ is being paid to the context in a sentence, not just a word alone. The decoder pays ‘attention’ to the entire sequence of representations built during encoding (Forcada, 2017: 299). The attention model helps the system analyse words that are important to hypothesise target-language equivalents and equally ignore less relevant words (Way, 2019: 317). This hypothesis is done by the decoder, which, informed by all input word representations, provides the most likely word at each position of the output sentence (Forcada, 2017: 296). The aim is to compute an association between the decoder state and each input word (Koehn, 2020: 57). This attention model leads to a significant decrease in morphological, lexical and specifically word-order errors in NMT outputs (Way, 2019: 317) and was “considered to be the bread-and-butter of NMT in 2017” (Forcada, 2017: 299).

An RNN requires a long sequential process, which makes it less able to identify the correct word dependencies over longer distances (Krüger, 2021: 3). This limits an RNNs capabilities because the processing of all words at once cannot be done in parallel, which in turn limits the efficiency GPUs can provide (Koehn, 2020: 93). Due to this limitation, other architectures exist and are being developed. One example is the ‘convolutional architecture’, where the sentence is not recursively encoded but rather where the decoder generates representations of each word by considering a few words on each side (left and right) of said word (Forcada, 2017: 299). The convolutional architecture is considered to be a good choice for image processing (Koehn, 2020: 93). By paying attention to the surrounding words, a translation is predicted. Since 2017, however, the newest NMT architecture is based solely on attention mechanisms. Thanks to the ground-breaking work by Vaswani et. al. (2017), the Transformer model based on self-attention has been dominant leading to a decrease in costs and an increase in efficiency and quality, as discussed in section 1.5. NMT systems can, however, still be trained on the architecture based on RNNs combined with attention mechanisms, as shown in the research conducted for this dissertation.

1.5 Transformer Architecture

In 2017 Vaswani et al. (2017) revolutionised the field of NMT by introducing the Transformer architecture, on which an NMT system could be trained solely based on attention mechanisms. Sequence-aligned RNNs and the convolutional architecture play no role in this new Transformer architecture. Vaswani et al.'s (2017) Transformer model is considered to be more superior by producing translation outputs of higher quality, exhibiting behaviour related to the syntactic and semantic structure of sentences. Furthermore, the Transformer model is more parallelisable (able to train simultaneously), which requires less training time and less computational complexity, in comparison to previous model architectures. The Transformer architecture "avoids recurrence completely and gives better translations depending on the stacked self-attention and fully connected layers between encoder and decoder" (Basta, Costa-jussà and Follonosa, 2020: 99). In the first experiment conducted by Vaswani et al. (2017) to introduce the Transformer, their big Transformer model outperformed the previously best recurrent or convolutional models by more than 2 BLEU scores. BLEU scores and other automatic evaluation metrics will be outlined in section 1.6.

Until the Transformer was introduced, a model architecture was used to train NMT systems where attention mechanisms are used together with an RNN. The experimental part of this research was conducted on such a model architecture. Attention mechanisms have been the key part of such model architectures, "allowing modelling of dependencies without regard to their distance in the input or output sequences" (Vaswani et al., 2017: 2). The Transformer architecture, the first of its kind, relies solely on attention mechanisms, specifically self-attention, which relates different positions of a single sequence to compute a representation of said sequence (ibid.: 2).

The Transformer architecture follows the encoder-decoder framework while making use of layered self-attention. The idea of self-attention is to extend the capability of the decoder to pay attention to the input words in the encoder, where association could be computed between any input word and any other output word (Koehn, 2020: 97). Figure 3 below displays the model architecture of the Transformer. This is merely introduced for background information on NMT considering that the Transformer architecture is the state-of-the-art model used to train NMT systems both in industry and in research. Since this dissertation is not based on the Transformer, but rather the RNN-attention model, only a short overview of the Transformer model is presented for explanation.

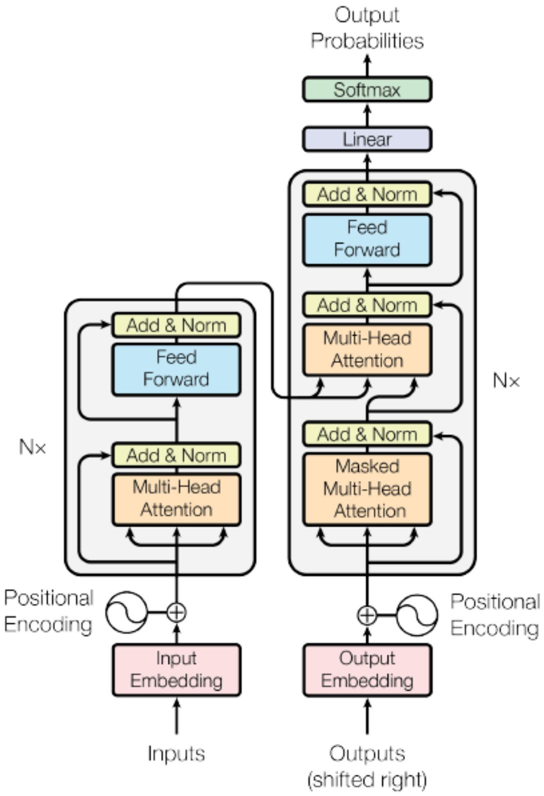


Figure 3. The Transformer Model Architecture as presented by Vaswani et al. (2017: 3)

As described above, the Transformer equally encodes the source text and then decodes it into a target output. However, it does it quite differently by applying a Multi-Head self-attention mechanism. The encoder has two sub-layers: one layer based on the multi-head self-attention and the other layer based on a feed-forward network (as outlined in Figure 3). The decoder has three sub-layers: similarly, one layer is based on the multi-head self-attention, another layer is based on a feed-forward network, and the third layer performs multi-head attention over the output received by the encoder (Vaswani et al., 2017: 3). Furthermore, the Transformer employs residual connections around each sub-layer both in the encoder and the decoder, followed by layer normalisation (ibid.: 3). What marks the aspect of self-attention is that each position in the encoder can attend to all positions in the previous layer of the encoder. The same holds true for the decoder, which can attend to all positions in the decoder up to and including the current position as well as attend over all positions in the input sequence (output from the encoder) (ibid.: 5). This functionality mimics the encoder-decoder attention mechanism known from the sequence-to-sequence models outlined above.

As in previous model architectures, the Transformer vectorises word embeddings in the training process. What further differentiates the Transformer is a fully connected feed-forward network as well as positional encoding, both depicted in Figure 3. To the embeddings inputted into the encoder and then into the decoder, a ‘positional encoding’ of the same embedding-dimension is added. This function helps place the word embeddings in context by referring to an encoded position.

Whereas Vaswani et al. (2017) defend the Transformers functionality solely relying on self-attention, Hahn (2020) outlines theoretical mathematical limitations for just that. Hahn mathematically outlines strong theoretical limitations of the computational abilities of self-attention, showing that self-attention can neither model non-counter-free regular languages nor hierarchical structure in both soft and hard-attention (2020: 167). However, these theoretical limitations do not seem to be confirmed in practice since Transformers provenly outperform previous state-of-the art NMT models. Nevertheless, it is an interesting aspect to consider theoretical limitations of the current “work-horse” of NLP (ibid.: 156). Transformers limit their expressiveness because they do not process input sequentially (as done in recurrent models for example). As outlined by Hahn (2020), self-attention present mathematical limits for hierarchical structures, which are considered essential to modelling natural language, specifically its syntax. Experiments conducted show that Transformers are less capable of modelling hierarchical structures than LSTMs (ibid.: 156). Practically, however, Transformer-methods are very successful in modelling natural language assumed too weak in theoretical linguistics (ibid.: 167). Hahn’s outline of the theoretical limitations of Transformers is very interesting and supports the choice of this dissertation basing its research on a recurrent-attention model architecture.

1.6 Automatic Evaluation for NMT

In this section, a few, of many, automatic evaluation models for MT will be outlined. Even though the field of MT, specifically training MT systems, has recently seen increased research interest, research of MT evaluation has not kept up (Rei et al., 2020: 2685). The development of MT models has therefore been the centre of research attention with MT quality increasing. At the same time, however, the development of adequate and comparable automatic evaluation methods has been lacking. This makes score comparisons between research papers and new methodologies rather difficult, even when using the same automatic evaluation, because different parameters may be applied. It is still widely agreed that human evaluations of MT are extensive and more accurate than automatic evaluations, yet they take much longer and, consequently, are expensive (Papinei et al., 2002: 311). Automatic evaluations are calculated much quicker, inexpensive in comparison, language-independent and, some more than others, highly correlate with human evaluation (ibid.: 311). Automatic evaluation metrics are calculated by comparing the machine translation mostly referred to as the *hypothesis* with a human reference translation, referred to as the *reference*. The idea is that the closer the hypothesis is to the human-created reference translation, the better the quality of the machine translation.

A selected few automatic evaluation metrics will be outlined in this section, including traditional string-based metrics such as precision, recall, F-measure, chrF (Popovic, 2015), METEOR (Banarjee and Lavie, 2005), the popularly used BLEU score (Papinei et al., 2002) and its adapted sacreBLEU score (Post, 2018), as well as rather new embedding-based metrics such as COMET (Rei et al., 2020) and BERTSscore (Zhang et al., 2020). Some of these metrics will be calculated in the analysis section (section 5.2) of this research merely to metrically compare results of the different trained NMT models for this research. However, the main analysis will be based on calculating and comparing the presence and reflection of gender in the relevant translation outputs in section 5.1 and 5.3.

1.6.1 Precision, Recall, F-Measure, chrF

To compute string-based automatic evaluations, the concept of *n*-grams needs to be understood. The 'n' in *n*-grams gets replaced by the number of words looked at in a sentence and can be a 1-gram (unigram), 2-grams, 3-grams etc. Using *n*-grams, a sentence can be divided into sub-sequences that the computer can process and compare.

Here is an example to outline *n*-grams:

Sentence 1 (MT): The cat is wearing a hat.

Sentence 2 (human): The cat is wearing a cap.

The first "The" present in both sentences represents a 1-gram. "The cat" represents a 2-gram, "The cat is" represents a 3-gram and so on. The maximum equivalent n-gram of these two sentences is a 5-gram for "The cat is wearing a". The last word (either 'hat' or 'cap') differs for the two sentences.

Precision is computed by counting the number of words (unigrams) – they do not need to be in order – in a MT sentence that also occur in a human reference translation, and then dividing this by the total number of words in the MT sentence (Papinei et al, 2002: 312):

$$Precision = \frac{number\ of\ words\ in\ hypothesis\ that\ are\ the\ same\ as\ in\ reference}{total\ number\ of\ words\ in\ hypothesis\ sentence}$$

Take the same two sentences as above, where sentence 1 is a MT output and sentence 2 is a human reference translation. There is a total of 6 words in the human reference sentence as well as in the hypothesis. As described, there are 5 words (unigrams) that appear in both sentences – here they are in order, but they do not need to be next to each other: "The hat is wearing the cat" would also yield 5 unigrams – which would yield a precision of 5/6 = 0.83.

Recall measures the proportion of the matched *n*-grams from the total number of *n*-grams in the human reference translation (Banarjee and Lavie, 2005: 67):

$$Recall = \frac{number\ of\ words\ in\ hypothesis\ that\ are\ same\ as\ in\ reference}{total\ number\ of\ words\ in\ reference\ sentence}$$

To calculate recall, the number of words in a machine translated sentence (hypothesis) that are the same in the human-reference sentence (reference) is divided by the number of words in the reference. In this case, the recall for the sentences above would also be 5/6 because sentences 1 and 2 are, by chance, of equal length. If they are unequal in length, however, the precision and recall score will differ.

Both scores are calculated between 0 and 1, with 1 being the highest possible score, meaning that the machine translation exactly equals the reference translation and is thus said to have a high quality. Precision shows how many words were ‘wrong’, words that were translated by the MT but are not present in the reference. Recall, on the other hand, shows how many words were ‘missing’, words that were in the reference but not translated by the MT. Important to note is that recall and precision are both string-based metrics and superficially compare *n*-grams of a hypothesis and a reference without taking synonyms or similar into account.

F-Measure is the combination of recall and precision (Wang and Li, 2019: 4136):

$$F - Measure = \frac{2x(precision\ x\ recall)}{precision + recall}$$

F-Measure is thus calculated by multiplying precision and recall by each other and then by 2, and then dividing this by the sum of precision and recall. Recall, precision and F-Measure do not take the order of words in a sentence into account. These metrics simply compare *n*-grams and would ironically yield the same results for a sentence “The cat is wearing a hat” as for “A hat wearing is the cat”. Such limitations of string-based metrics calculated by comparing *n*-grams is tackled by embedding-based evaluation metrics.

chrF (Popovic, 2015) short for character *n*-gram F-score is also a string-based metric but works on a character-level rather than a word-level. For chrF, *n*-grams do not represent words, with 1-gram representing one word, but rather represents characters, with 1-gram representing one character. The ‘F’ in chrF refers to F-Measure, so chrF is the “F-score based on character *n*-grams” (ibid.: 392). As described in section 1.6.2 below, BLEU scores are fixed to be calculated on 4-grams (words), and chrF to be calculated on 6-grams (characters). This has been taken as the standard method because scores based on these calculations yield the closest correlation to human judgement (human instead of an automatic evaluation) of a machine translation.

1.6.2 BLEU

BLEU, short for Bilingual Evaluation Understudy, was developed by Papinei et al. in 2002 and has since been the dominant metric to calculate and analyse MT results with (Papinei et al, 2002; Post, 2018: 186). In presenting BLEU, Papinei et al.’s (2002) aim followed the notion that “[t]he closer a machine translation is to a professional human translation, the better it is” (311). BLEU measures – with shortcomings, as later described – the closeness of a MT sentence to one or more human reference translations and scores it as a numerical metric. The BLEU metric equally ranges from 0 to 1, where a score of 0 means the MT sentence is furthest away from the human reference translation (and therefore of low/no quality). A score of 1 means the translation exactly resembles the human reference translation (and is therefore considered a perfect translation). When BLEU was introduced in 2002, it was considered to correlate highly with human judgements (Papinei et al, 2002) but since, the field of automatic evaluation has developed and weaknesses of the BLEU metric have been discussed with new metrics being presented (Post, 2018).

The BLEU metric compares *n*-grams of a MT sentence with the *n*-grams of a human reference sentence. The number of position-independent *n*-gram matches are counted, and the more matches, the better the MT sentence (Papinei et al., 2002: 312). The BLEU metric is based on *precision*, re-formalising it as *modified unigram precision* (ibid.: 312). The exact details of which will not be outlined, but it is important to note that this modified *n*-gram precision focusses on adequacy and fluency, where equal unigrams satisfy adequacy, and longer *n*-gram matches satisfy fluency. The basic BLEU metric calculates a score by comparing up to and including 4-grams between the MT sentence and the human reference sentence.

The formula behind a BLEU calculation is (Papinei et al., 2002: 316):

$$BLEU = \mathrm{BP} * \exp(\sum_{n=1}^{N} w_n log\, p_n)$$

Where *BP* stands for the brevity penalty, *N* for the n-grams analysed up to length *N*, w_n for the positive weights and p_n for the modified unigram precision. While long hypothesis sentences are penalised by *n*-gram precision, short hypothesis sentences that are too short are penalised by the brevity penalty. The brevity penalty equals 1 when the length of the hypothesis sentence equals the length of the reference sentence (ibid.).

What makes automatic evaluation tricky is that there can be many correct translations of a given source sentence that may vary in word choice or word order (Papinei et al., 2002: 312). Any given five specialised human translators could translate the same source sentence differently. Nevertheless, all translation outcomes, even though they might be phrased differently, could all be perfectly good translations. This means, that it is somehow faulty to compare a MT sentence to a single human translated sentence. A calculated metric becomes more accurate if compared to numerous human reference sentences. BLEU yields a higher (and more accurate) score for MT sentences when compared to numerous reference translations, which may vary in word choice and phrasing. In the case of BLEU, therefore, “quantity leads to quality” (ibid.: 318). However, in practice, there may often only be one human reference sentence, especially in the case of larger corpora, as those used to train MT systems. Updated automatic evaluation metrics, as described in the following sections, try to calculate scores that are less dependent on the exact phrasing or word-choice.

Many aspects of BLEU can be altered for evaluation, such as the number of *n*-grams being compared or the number of human references a MT sentence is compared with. This can lead to the calculation of very different scores, and not enough attention is given to this disparity (Post, 2018: 187). Such a disparity in scores leads to the fact that the actual numerical BLEU value cannot simply be compared as a number, even though BLEU is a preferred-chosen metric to evaluate MT translations in MT research. Since the underlying parameters – ranging from differing *n*-grams and number of references to different applied tokenization and normalisation schemes – can and do vary to calculate BLEU scores, such scores cannot equally be compared as a numerical value (ibid.: 186). BLEU has been a favoured automatic evaluation metric across MT research, however, the mentioned shortcomings have led to the introduction of new evaluation metrics such as sacreBLEU and COMET.

1.6.3 METEOR

METEOR, short for Metric for Evaluation of Translation with Explicit Ordering, is based on matching unigrams – either on their surface form, their stemmed form, or meanings – between the MT sentence and a human reference sentence (Banarjee and Lavie, 2005: 65). METEOR was developed to address the weakness of BLEU such as surface-based word matching and not taking recall into account. METEOR can match synonyms and words that are morphological variants of each other (ibid.: 66).

A combination of unigram-precision, unigram-recall and F-measure are calculated to produce a METEOR score. If more than one reference translation is available, the best scoring match (to a human reference sentence) is used to compute a combined score for the MT system over the entire test set (Banarjee and Lavie, 2005: 69). An alignment, a mapping between unigrams, is used to map each unigram from one string to zero or one (but not more) in the other string (ibid.: 67).

METEOR scores higher than BLEU in correlation with human judgements, particularly at the segment level, which is important to include minor differences. Today, the traditional METEOR metric is similarly rather outdated, with adapted METEOR-versions and more advanced metrics taking over. However, in 2005 METEOR's objective was to address weaknesses of the BLEU metric, leading to more advanced automatic evaluations methods today.

1.6.4 sacreBLEU

Inconsistency in the calculation of BLEU scores in MT research due to differing parameters partially leads to discrepancies (of current published BLEU scores) often bigger than many reported gains (Post, 2018: 186). As the main cause of this inconsistency Post (ibid.) identifies individual user-supplied tokenization (independently applied per research) and suggests the application of only a 'metric-supplied' reference tokenization. Traditional BLEU scores can only be compared if the pre-processing is the same. However, user-supplied pre-processing is prone to errors and differs widely, making it inadequate to compare across research papers (ibid.).

Post's solution, sacreBLEU[1], to this is a new methodology with an internal metric-supplied tokenization, meaning that it will be identical for each user applying it. This would enable a direct comparison of numerical scores. sacreBLEU is a Python script and expects detokenized MT sentences, as it then applies its own metric-internal pre-

[1] https://github.com/mjpost/sacrebleu

processing for comparability (ibid.: 189). sacreBLEU also automatically downloads and stores reference sentences for commonly used test sets and with that, introduces a 'protective layer' between these references and the user (ibid.: 187). The aim is to have a fast and cheap way for researchers to accurately evaluate and compare their models.

1.6.5 COMET

As mentioned, there are numerous automatic evaluation metrics for MT, some of which are widely applied, such as BLEU. However, Rei et al. criticise that many existing metrics "struggle to accurately correlate with human judgement at segment level and fail to adequately differentiate the highest performing MT systems" (2020: 2685). Existing string-based metrics based on comparing *n*-grams in a MT sentence and a human reference sentence by nature usually fail to recognise and assess semantic similarity. Embedding-based metrics have emerged that, unlike string-based metrics, create soft-alignments between a human reference sentence and a MT hypothesis sentence in an embedding space. A score is then computed, reflecting the semantic similarity between said reference and hypothesis segments (ibid.: 2692). Capturing semantic similarity brings automatic evaluation one step closer to human judgement.

Rei et al. (2020) present a new neural PyTorch[2]-based framework 'COMET'[3] to train multilingual MT evaluation models that correlates highly with, and can easily be adapted and optimised to, different types of human judgements. COMET shows progress to achieving a better correlation to human judgement at segment level and in achieving better robustness to high-quality MT (ibid.: 2686). The COMET framework supports two architectures, depicted in figure 4 below: the *Estimator model* – trained to regress directly on a quality score – and the *Translation Ranking model* – trained to minimise the distance between a 'better'-ranked hypothesis and its corresponding reference and original source (ibid.: 2686). These two models in turn are composed of a *cross-lingual encoder* – such as BERT, which contain transformer encoder layers that uncover the relationship between masked tokens and their surrounding ones to recreate masked tokens – and a *pooling layer*, where information from the most important encoder layers is 'pooled' into a single embedding for each token using a layer-wise attention mechanism (Rei et al., 2020: 2686).

[2] PyTorch will be further explained in section 1.7

[3] https://github.com/Unbabel/COMET

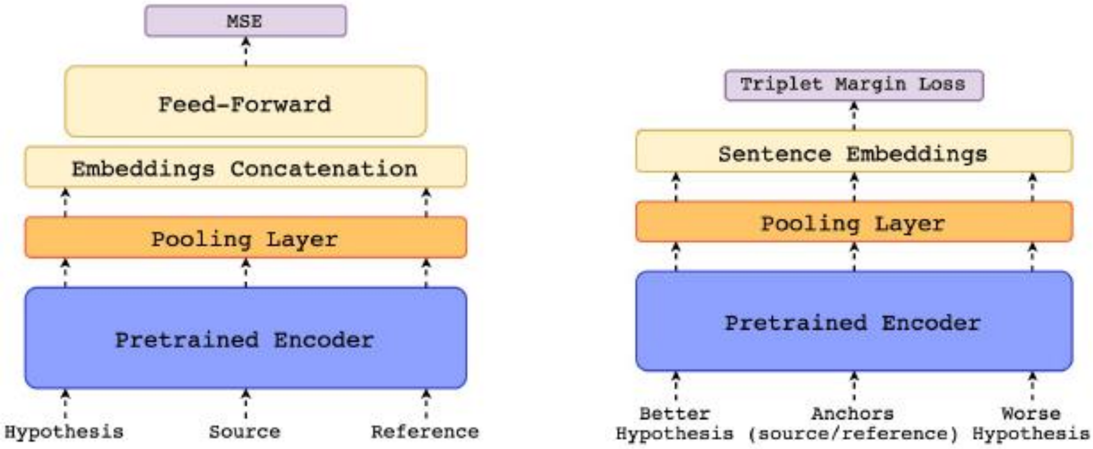


Figure 4. Estimator model architecture (left). Translation Ranking model architecture (right). (Rei et al., 2020: 2687)

In both architectures, the three inputs – (better / worse) hypothesis, source and reference sentences – are each encoded by the cross-lingual encoder. The resulting word embeddings in both architectures are then passed through the pooling layer to create a single sentence embedding for each segment. In the Estimator model architecture, these sentence embeddings are concatenated into a single vector, which is passed to the feed-forward regressor. the Estimator model architecture is trained by “minimizing the Mean Squared Error (MSE)” (Rei et al., 2020: 2687), the last step depicted in figure 4. The Translation Ranking model architecture does not have this concatenation and feed-forward step but instead, the embedding spaces are optimised (using the triplet margin loss depicted in figure 4) to minimise the distance between the ‘better’ hypothesis and the source and reference (anchors).

In experiments conducted by Rei et al. (2020) COMET scores outperform metrics including BLEU, chrF and BERTScore. The COMET models achieve advanced results for segment-level correlation with human judgements and they show an improved promising ability to differentiate high-performing systems (ibid.: 2685).

1.6.6 BERTScore

Zhang et al. (2020: 1) similarly criticise that string-based metrics only count and compare overlapping *n*-grams between the MT hypothesis sentence and the human reference sentence, not accounting for semantic and compositional diversity. BERTScore[4] is a language generation evaluation metric that calculates a similarity score using one BERT pre-trained contextual embedding for every token in a MT sentence compared to every token in the human reference sentence (ibid.: 4). This similarity score is computed as a sum of cosine similarities between the token embeddings of the MT sentence and the human reference sentence (ibid.: 4).

[4] https://github.com/Tiiiger/bert_score

The contextual embeddings are based on word embeddings, learned dense token representations, that capture the individual use of a certain token in a sentence, and potentially also capture sequence information (Zhang et al., 2020: 3). Contextual embeddings, as used in calculating BERTScore, can generate different vector representations for one word in different sentences depending on the surrounding words, i.e. the surrounding context (ibid.: 3). Such vector representations allow for a soft measure of similarity instead of exact-string (as done for the calculation of BLEU) or heuristic matching (as done for the calculation of METEOR) (ibid.: 4).

BERTScore calculates a modified version of the traditional metrics recall and precision: R_{BERT} and P_{BERT} (Zhang et al., 2020: 4). To calculate R_{BERT}, each token in a tokenized reference sentence is matched to a token in the tokenized hypothesis. To calculate P_{BERT}, each token in a tokenized hypothesis is matched to a token in the tokenized reference sentence (ibid.: 4). Matching each token to the most similar token in the other sentence is called the 'matching similarity score', which is maximised by applying greedy matching (ibid.: 4). F-Measure (F_{BERT}), the combination of R_{BERT} and P_{BERT}, is also calculated, and Zhang et al. (2020) recommend to primarily use F_{BERT} for measurement as it performs reliably well across different settings (7). Figure 5 depicts how R_{BERT}, P_{BERT} and F_{BERT} are calculated for a reference (*x*) and a hypothesis ($\hat{x}$):

$$R_{\text{BERT}} = \frac{1}{|x|} \sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \; , \quad P_{\text{BERT}} = \frac{1}{|\hat{x}|} \sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j \; , \quad F_{\text{BERT}} = 2\frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}$$

Figure 5. Calculations of BERTScores: R_{BERT}, P_{BERT} and F_{BERT} (Zhang et al., 2020: 4)

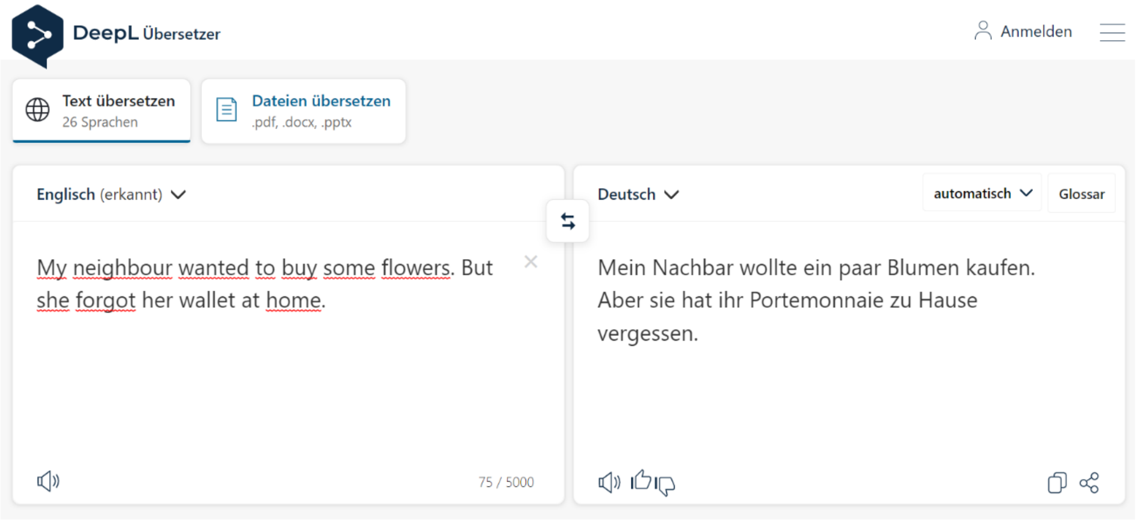
BERTScore more effectively detects paraphrasing, distant dependencies and ordering, compared to *n*-gram (string-)matching from the surface of a sentence (Zhang et al., 2020: 1, 9). Zhang et al. (ibid.: 4) further demonstrate that BERTScore can incorporate importance weighting for example of rare words and, as demonstrated in previous research, rare words can be a higher indicator for sentence similarity than common words.

1.7 Plattforms & Tools

1.7.1 Commercial MT Systems

There are a number of commercial MT systems available online, including the widely used Google Translate, DeepL and Microsoft Bing, employ NMT to produce translation outputs. As the name suggests, Google Translate is under the wing of Google and is being developed by the Google AI and Research Team. Microsoft Bing is under the wing of Microsoft and is developed by relevant employees. DeepL, an AI company created in Cologne, Germany, is comparably new to the field but outperforms the other two in many aspects, such as fluency. Microsoft Bing and Google Translate cover more languages, whereas DeepL focusses on fewer but widely used ones.

In research conducted by Hovy, Bianchi and Fornaciari (2020), they discovered that these commercial MT systems Microsoft Bing, DeepL and Google Translate all show systematic translation biases, most likely due to the use of biased training data. Figures 6 and 7 below depict just two of such biases present in commercial MT systems.

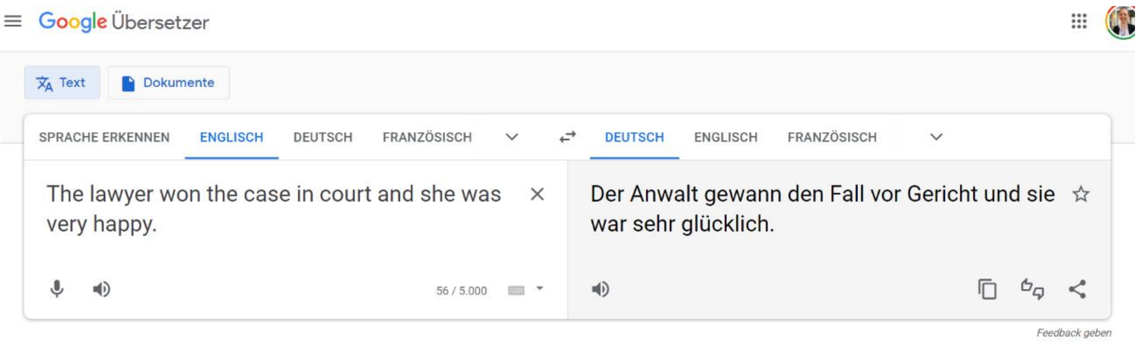


My neighbour wanted to buy some flowers. But she forgot her wallet at home.

Mein Nachbar wollte ein paar Blumen kaufen. Aber sie hat ihr Portemonnaie zu Hause vergessen.

Figure 6. Example Translation from DeepL. Screenshot taken on January 31st, 2022.

In this example, an English source sentence contains the gender-ambiguous word 'neighbour', which in this case refers to a woman, as clarified in the second sentence. DeepL, however, wrongly translated it as 'Nachbar' (male) instead of 'Nachbarin' (female), not being able to refer the gender learned in the second sentence back to the first sentence.



The lawyer won the case in court and she was very happy.

Der Anwalt gewann den Fall vor Gericht und sie war sehr glücklich.

Figure 7. Example Translation from Google Translate. Screenshot taken on January 31st, 2022.

In this example, an English source sentence contains the gender-ambiguous word 'lawyer', which in this case also refers to a woman, as clarified in the same sentence. Google translate, however, wrongly translated it as 'Anwalt' (male) instead of 'Anwältin' (female), equally not being able to refer the gender learned in the second half of the sentence back to the first half of the sentence.

These are just two examples, one taken from DeepL and one taken from Google Translate. There are countless examples of the presence of gender-bias in MT. A pattern for some linguistic phenomena is outlined in section 3. This dissertation highlights the existence of gender-bias in MT, by describing how they arise and by measurably showing how they can be reduced. The aim is to help generate more gender-balanced texts to consequently help shape a more gender-fair society.

1.7.2 NMT Toolkits

For SMT, Moses was the primary toolkit, which could be applied and extended for research purposes (Koehn, 2017: 90). For NMT and natural language generation, there are numerous freely available open-source toolkits such as OpenNMT, Fairseq, Sockeye and Marian (Klein et al., 2020: 102). These open-source toolkits are specifically made available for researchers to apply them, learn from, and extend upon for their research purpose. NMT toolkits are meant to serve as a foundation that can be built upon. A NMT toolkit should function as a shared framework to develop and compare open-source systems (Klein et al., 2017: 67). There have been a few cases in which the industry has made use of these open-source NMT toolkits, where OpenNMT, for example, was deployed by companies like SYSTRAN and Booking.com (Klein et al., 2020: 104). Generally, however, the industry, such as Google, Microsoft, and Baidu, develops their own closed NMT implementations (Klein et al., 2017: 67).

Through modelling and translation support in documentation and communities, researchers in the field of MT are intended to follow these toolkits. However, as Forcada states, “installing, configuring, and using [these toolkits] requires skills that are not usually possessed by professional translators” (2017: 302). Knowledge of computational science and specifically of the relevant computer language such as Python is required. Furthermore, NMT systems are difficult to train since they require large parallel corpora, which are not abundantly publicly available, dedicated hardware such as GPUs with sufficient computing power, and training times of minimally days (Forcada, 2017: 302). These restrictions do not make it easy for less computationally experienced researchers to train an NMT system using available toolkits in a relatively short timeframe. The rather small models trained during research conducted for this dissertation, as further discussed in section 4, took around 4 hours to train. An online platform with dedicated hardware including one GPU was purchased for this dissertation, as outlined in section 1.7.3.

Before diving into the details of how an NMT system was trained for this dissertation, which is explained in section 4, the open-source NMT toolkit used for this research, OpenNMT, will be outlined. OpenNMT is likely the most widely-applied NMT toolkit and has already been cited in over 700 research papers, with this number largely increasing every year (Klein et al., 2020). OpenNMT supports numerous model architectures and training procedures for NMT but also for language modelling or natural language generation (ibid.: 102). OpenNMT is an extensive library for training and deploying NMT models, including so-called vanilla NMT models (Klein et al., 2017: 68). Although the classification of ‘vanilla’ depends on the researcher’s computational skills in this area. The idea is that with OpenNMT (specifically a vanilla system), anyone can “throw together a competitive system using the right data, processing and training procedures” (Klein et al., 2020: 106). OpenNMT covers the complete machine learning workflow, starting with data preparation, and leading to interference acceleration (ibid.: 102). The toolkit comprises code for the core translation tasks, modelling and translation support, documentation about underlying techniques and an active forum for community feedback. The OpenNMT toolkit was designed to:

"(a) prioritize first training and test efficiency, (b) maintain model modularity and readability, (c) support significant research extensibility" (Klein et al., 2017: 68),

while ensuring competitive performance and reasonable training requirements. The aim of OpenNMT is to yield state-of-the-art MT results by supporting research, with a focus on model architectures, "while maintaining API stability and competitive performance for production usages" (Klein et al., 2020: 102).

OpenNMT, first released in 2016, is a successor of the previous 'seq2seq-attn' system developed by Harvard, which has been rewritten to be made more efficient, readable, and generalisable (Klein et al., 2017: 68). Its code is open-source, available on GitHub[5], and MIT licensed (ibid.: 68). The underlying OpenNMT system is implemented in the mathematical Lua/Torch framework and has been extended by Facebook AI Research to also support the Python/PyTorch framework (ibid.: 68). Torch is a ML library and script language based on the Lua programming language, where any PyTorch in turn is a variant of Torch (Koehn, 2020: 34).

To compare, the code for language modelling of the SMT Moses framework was written in over 100.000 lines, whereas the python version for the OpenNMT code spans less than 1.000 lines (Klein et al., 2017: 69). There are two current implementations of OpenNMT with its own design and set of features: OpenNMT-py for Python and OpenNMT-tf for TensorFlow 2. For this dissertation, the python code available under OpenNMT-py was applied to train an NMT model. OpenNMT-py contains command line utilities and a Python library to configure, train, and run models based on the OpenNMT principles of ease of use, efficiency, modularity, extensibility, and production readiness (Klein et al., 2020: 103).

OpenNMT has numerous design aspects and extensions, not all of which will be discussed here as they have not been directly used for the research of this dissertation. One aspect of NMT that does not yet directly affect MT but may well do so in the near future, is that NMT-like systems are provenly effective for image-to-text generation (Klein et al., 2017: 70). OpenNMT has also been used for tasks in the machine learning realm such as summarisation and automatic speech recognition (Klein et al., 2020: 104).

Originally, OpenNMT has been developed on a sequence-to-sequence model, as an extension of the previous 'seq2seq-attn' system, implementing the recurrent with attention model architecture, as initially presented by Bahdanau, Cho and Bengio (2014). As part of this dissertation, this 'recurrent with attention' OpenNMT model architecture was used to conduct train NMT models. Other frequently used model architectures for MT

[5] https://github.com/OpenNMT/OpenNMT-py

that can be implemented with OpenNMT include the self-attentional model, also known as the Transformer architecture, as well as the convolutional model (Klein et al., 2020: 104).

Important to know is that for OpenNMT, attention, through the global attention mechanism, is applied over the encoder at each step of translation: a crucial aspect for the model to perform well (Klein et al., 2017: 70). Different attentions, such as local or hierarchical attention, can be used as a substitute for the standard attention applied but these will not be discussed in detail. Another important aspect of OpenNMT is that it includes a reversible tokenizer, where the model allows for tokenization and detokenization. The OpenNMT-included tokenizer can perform Unicode-based segmentation, subword training and encoding, either Byte Pair Encoding (BPE), a method for sub-word tokenization in NMT systems, or SentencePiece[6] (Klein et al., 2020: 106). Furthermore, OpenNMT allows for simplification of the process of pretrained word embeddings, including the automatic download of embeddings for many languages (Klein et al., 2017: 71). This allows for training even when relatively little data is available. OpenNMT also provides an optimised interference engine for Transformer models, CTranslate2, which will not be further discussed here since this research was not trained using Transformer models but rather the RNN-model provided by OpenNMT-py.

Restrictions include NMT systems taking minimally days and weeks for larger corpora and more complex models to train as well as memory size restrictions being the most common limiter of batch size and therefore training time when training GPU-based NMT models (Klein et al., 2017: 69). The aim of OpenNMT is to provide a state-of-the art ecosystem for NMT, for which Klein et al. (2020) re-work published papers to further develop promising features and implement technologies at the core of translation. Klein et al. (ibid.) aim to release ready-to-use state-of-the-art models for a number of language pairs to simplify the adoption of NMT. Koehn argues that frameworks such as OpenNMT are

> "less geared towards ready-to-use neural network architectures, but provide efficient implementations of the vector space operations and computation of derivatives, with seamless support of GPUs" (2020: 35).

A simplified and updated ready-to-use version would noticeably simplify research, like this dissertation, that rely on NMT toolkits, such as OpenNMT.

1.7.3 Google Colab Pro

Google Colab is an online coding interface made available by Google (Google Colab, n. d.a). Google Colab enables Google users the execution of code in a notebook that can be saved in ones Google Drive folder. Google Colab can further be actively linked to the Google Drive folder to save data. Data and files used in codes on Google Colab and created outputs can be saved in the Drive folder and be easily accessed in the notebook.

[6] https://github.com/google/sentencepiece

The entire process is cloud-based, meaning that the data does not need to be saved locally on one's computer and use up local RAM. However, it also means that for large or multiple datasets and files a lot of cloud-space is needed per Drive. Extra cloud-storage was purchased for this research so that the parallel corpus and all model steps and files could be saved in the Drive folder.

Multiple Google Colab notebooks can be created for coding. Google Colab users can work on one notebook at a time, whereas on Google Colab Pro, a paid version of Google Colab for around 10€ per month, users can simultaneously work on multiple notebooks (Google Colab, n.d.b). Google Colab Pro was purchased for this research for a more efficient workflow. For more storage and potential, Google Colab Pro+ can be purchased. Copies of one's notebooks can be created and shared with other Google users, so that others can edit and run through the (copied) written code and obtain their own results without altering the original notebook.

This research was conducted using the Python language, which can be written on Google Colab notebooks. At times, adjustments need to be made to the code when working on Google Colab in comparison to working on a local python programming interface. As long as users are aware of these slight differences, Google Colab notebooks are ideal for coding with Python.

The general setup for Google Colab is that the user uses a CPU runtime on the interface, but they do provide the possibility to change the runtime to one GPU, and to include more RAM. This was necessary for this project since the OpenNMT-py model requires at least one GPU for training. Furthermore, even while using a GPU, it still takes numerous hours to train a basic NMT model. Apart from the fact that the NMT model in question cannot be trained on a CPU runtime, it would take days to complete the training.

2 Bias in MT

Bias on prediction is a common occurrence in the field of natural language processing (NLP) and is therefore automatically also found in the field of machine translation. Naturally occurring bias in statistical models is almost inevitable, which is why its acknowledgement is so crucial, and must be addressed with proactive measures. Blindly applying machine learning to areas such as translation risks amplifying bias present in data (Bolukbasi et al., 2016: 1). Once naturally occurring bias in MT has been acknowledged, the next step is to detect and measure bias in MT. Bias detection can then help mitigate bias and improve the generalisation ability of MT systems. Even though performance in machine learning and MT systems is improving overall, this performance is imbalanced, with certain groups of people being represented more than others. This over- and underrepresentation further harms groups that are already marginalised, leading to heightened inequality (Romanov et al., 2019: 4187). The overall aim of tackling bias in MT is to fundamentally provide unbiased translation outputs for all demographic groups, leading to more fairness in society.

In NLP, predictive models such as MT engines are sensitive to primarily unintended biases through the development and training process (Shah, Schwartz and Hovy, 2020: 5248). Shah, Schwartz and Hovy state that “every predictive model with errors is bound to have disparities over human attributes” (ibid.: 5257). Such disparities can originate either in the embedding model, the feature sample, the fitting process, or the outcome sample (ibid.: 5257). Word embeddings trained on large parallel corpora (stemming from humans) naturally exhibit gender stereotypes thus amplifying biases. From a geometrical view, directions in word embeddings show gender bias, and vectors of words with similar semantics cluster together (Bolukbasi et al., 2016: 1). The (gender) stereotypes these word embeddings reflect directly stem from the data on which they were trained, representing stereotypes present in broader society (ibid.: 3). The result is that models are less able to generalise on unseen data and therefore less able to provide reliable output. This in turn can lead to undesirable social effects like underserving or wrongly predicting certain user groups (Shah, Schwartz and Hovy, 2020: 5248).

The result of bias in NLP is, on the one hand, a result of restricted domains but, on the other hand, also of the human-generated language that reflects social individuality of the people who use it. NLP, such as NMT models, often cannot generalise social individuality to other author demographics (Shah, Schwartz and Hovy, 2020: 5248). Biases in machine learning stem from stereotypes among a group of people, with ‘biased’ word embeddings then being used to train MT systems, projecting gender stereotypes present in society (Bolukbasi et al., 2016: 4). Per se, biases are not negative, they are, as Daniel Kahneman and Amos Tversky defined, mental ‘shortcuts’ that help us react faster to situations and are therefore a result of psychological heuristics (Shah, Schwartz and Hovy, 2020: 5262). These heuristics can be very useful in certain situations but, if left unscrutinised and over-generalised, they can have negative consequences, like leading to the unwanted bias in MT.

There are varying formulations by researchers of what bias is or does, including bias being an unfairness in algorithms, bias being an aggregate effect for demographic groups and bias potentially harming the data (ibid.: 5254). When focussing on the category of personal traits or demographics, the types of existing biases are manifold, with racial and specifically gender bias being some of the most focussed on and researched biases in the field of MT. This is due to limited availability of parallel corpora for research, and that gender bias is more easily detectable and measurable in translation outputs. Like most recent research papers, this dissertation focusses on binary gender bias in MT, with a focus on comprehensibly explaining how and why gender bias occurs in MT in the first place. Even within the category of gender bias in MT, there are numerous sources and type of occurrences of said bias as explained in section 2.1 and 2.2.

There are multiple reasons why the issue of gender-biased MT outputs needs to be focussed on and resolved. An obvious reason is that an individual translating a text could be directly offended by the biased machine translation output. This may be a woman who is a lawyer, but the MT makes her appear to be male. It may be a man who is a primary teacher, but the MT makes him sound female. It may be a binary person, but the MT makes them sound either male or female. There are countless examples of how a biased MT can directly offend the subject of the text. Another reason is that a person who is not fluent in the language they translate will pass on this potentially biased machine translation on without understanding or proofing it. If this person is not proficient in the resulting machine translated language, it is impossible for them to assess and correct potential bias in the MT outputs. In turn, this can lead to a potentially biased translation being passed on to a third party. Missing biases in MT-outputs can lead to increased gender bias in the wider population. If potentially biased translation outputs are passed on, uploaded, or published, this bias would be spread and the public would, likely subconsciously, consume texts containing such bias. This could subconsciously form public opinion of gender-affected topics in society and form their way of speaking, which in turn will continue to feed the cycle of spreading gender-bias.

2.1 Sources of Bias in MT

Recent research has increasingly concerned itself with addressing countermeasures for existing predictive biases in MT, as outlined in section 2.3, focussing on the effects and symptoms of bias. However, often it is not explicitly addressed, where such bias originates (Shah, Schwartz and Hovy, 2020: 5252). In research conducted by Stanovsky, Smith and Zettlemoyer (2019) and Font and Costa-jussà (2019) it seems that cultural and societal biases are primarily reflected in machine translation systems with bias originating in data selection and embeddings (Hovy, Bianchi and Fornaciari, 2020: 1689).

Biases in MT outputs primarily arise from the dataset used to train the engine. If this training data is skewed in a certain direction, the machine learns this skew and reflects it in the translation outputs. Datasets often contain more examples of men than of women and if a MT system is trained on this data, they will exhibit or even amplify these biases

(Saunders and Byrne, 2020: 7724). MT training data is mainly created by humans, such as from human speech or human-written texts. The machine simply takes a given input (in many cases inherently skewed, biased datasets) and learns from this data to form most-likely predictions based on this training data. As Gonen and Goldberg (2019: 609) state, word representations in MT provenly reflect social bias (including racial and gender bias) because they naturally occur in the data used to train them. However, the way a MT model is configurated to learn from a certain data set, such as overamplifying certain tendencies, also plays a role in forming biases.

Numerous different definitions and categorisations of bias in MT have been formulated. Shah, Schwartz and Hovy (2020: 5248) split sources of bias into four categories: selection bias, label bias, model overamplification, and semantic bias. According to Shah, Schwartz and Hovy (ibid.: 5252), bias in labels may arise from erroneous demographic attribute of interest, a non-representative group of annotators, a lack of domain expertise, or by annotators holding preconceived notions and stereotypes. Non-representative data is the origin for selection bias, which may lead to the source not reflecting the ideal distribution (expected output), and hence leading to a lower accuracy for a given demographic (ibid.: 5252). ‘Overamplification’ and ‘semantic bias’ are likely the most relevant types of biases for this dissertation.

The MT system can lead to ‘overamplification’ during the learning phase, where “a model relies on a small difference between human attributes with respect to the objective […] but amplifies this difference to be much more pronounced in the predicted outcomes” (Shah, Schwartz and Hovy, 2020: 5253). For example, if a MT model is trained on data where 8/10 apples are red, the model would likely overamplify this by always outputting apples to be red.

‘Semantic biases’ are often attributed to word embeddings, which frequently contain unintended social stereotypes (Shah, Schwartz and Hovy, 2020: 5254). Embeddings used to train MT systems are often used without access to the original data, therefore without being placed in context. Problematic is that male authors provide more training data for embeddings and that gendered pronouns are mentioned alongside certain occupations that are not ideally distributed (ibid.: 5254). Gonen and Goldberg (2019: 609) agree that one ‘source’ of bias in MT are word embeddings. Word embeddings derived from the parallel corpora used to train MT models reflect gender bias in society. This is further outlined in section 2.3. Caliskan, Bryson and Narayanan (2017) propose the Word-Embedding Association Test (WEAT), which makes semantic bias quantifiable based on the distance between words with demographic associations in the embedding space. Hovy and Spruit (2016) categorised three types of biases: demographic bias, overgeneralization, and topic exposure. There are many more attempts at defining and categorising bias in MT. In practice, biases combine and do not occur in isolation, increasing their effects (Shah, Schwartz and Hovy, 2020: 5254). This dissertation focusses primarily on semantic bias or demographic bias, in combination with other bias categories such as overgeneralisation/overamplification where relevant.

2.2 Gender Bias

Gender bias is distinctly present in NLP such as MT in varying forms, which are further outlined in section 3. Gender bias can be seen in translation outputs opting for wrong pronouns, professions being translated into the wrong gender directly or by implication (‘doctor’, ‘nurse’), adjectives defining a certain gender (‘fragile’, ‘competent’), trigger words leading to the choice of a certain gender (‘flower’, ‘hammer’), or general word choices or syntactics being skewed by assuming the wrong gender. More detailed examples are presented in section 3 but explanations of why this occurs and how this has been researched is outlined in this section.

Gender bias is manifested in datasets that naturally feature or refer to more men than women (Saunders and Byrne, 2020: 7724). NMT amplifies stereotypes and yield a lower performance of speech recognisers for women than for men (Costa-jussà and de Jorge, 2020: 26). MT systems are dominantly trained on unbalanced data, which will perpetuate bias by MT engines training on these datasets and by people who use these systems and will learn incorrect associations between words, unknowingly perpetuating social biases (ibid.: 26). To eliminate bias in translations and to create a fairer social distribution in outputs, MT systems must be trained on balanced data.

Gender bias in MT also occurs by MT systems dominantly translating on a sentence-by-sentence basis, ignoring the context. This phenomenon leads to translations of professions being translated with stereotyped genders (Basta et al., 2020: 99). Take “The lawyer won in court yesterday. She was extremely happy” as an example. The first sentence is gender-ambiguous where the lawyer could be male, female, or of course non-binary. The second sentence defines the lawyer to be a woman. By translating sentence-by-sentence a MT system would not pick up on this and likely translate the lawyer as a male, similar to the examples shown in section 1.7.1. Demographic factors, such as gender, influence our natural use of language in terms of word choice or on the level of syntactical constructions, and this information is integrated into the MT systems we train (Vanmassenhove, Hardmeier and Way, 2018: 3003). Gender bias in MT consistently and demonstratively appears across different word embeddings and, for example, associates specific neutral professions with males and others with females (Gonen and Goldberg, 2019: 613).

Many languages other than English have grammatical gender systems, such as German, and such gender knowledge is encoded into MT systems (Vanmassenhove, Hardmeier and Way, 2018: 3006–3007). In research conducted by Hovy, Bianchi and Fornaciari, (2020: 1687), a male bias exists in most predictions of the original language, with translated English versions creating an even stronger skew. Furthermore, the way males and females use language differs in terms of style and syntax, like women using less assertive speech. Research about ‘author-profiling’ or author classification is increasing and high accuracies on domain-specific data is reached (Vanmassenhove, Hardmeier and Way, 2018: 3003–3004). Such author traits are often lost in human and especially in machine translation. These losses “harm the overall fluency and adequacy of the translated sentence” (ibid.: 3003).

In addition to producing gender-skewed translations, demographic groups may be undermined in datasets leading to an unfair representation of society. This occurs partially because different demographic groups use different dialects and word-choices, which may not be included in ‘standard’ training data, thus excluding these demographic groups in MT systems (Bolkbasi et al., 2016: 5). Current MT systems trained on data sets provided “have a tendency to perpetuate a male bias which amounts to negative discrimination against half the population” (Vanmassenhove, Hardmeier and Way, 2018: 3003). This refers mainly to binary gender, not even taking non-binary gender into detailed consideration. Hovy, Bianchi and Fornaciari (2020: 1686) further show that there are substantial discrepancies in perceived demographics in MT translated content and that MT translated texts tend to appear as written by writers older and considerably more male than the original.

This is due to the fact that current MT systems do not, as is done by human translators, take contextual information into account to produce gender-adequate translations. Rather, MT systems translate sentences in isolation by exploiting “statistical dependencies on the sentence level that have been learned from large amounts of parallel data” (Vanmassenhove, Hardmeier and Way, 2018: 3003). This, however, partially stems from MT training data often not including the original demographic traits of the author, making it very difficult for a MT system to determine the author’s gender (ibid.: 3004). Author traits such as gender should be integrated in MT systems since this also affects word choice and the level of syntactic constructions.

There are different words and contexts for which gender bias occurs in MT, as further summarised in section 3. In their work, Gonen and Goldberg (2019) show that male- and female-biased words cluster together due to male/female stereotyping. Professions is a typical category where words can be strongly female- or male-biased. Words that are inherently biased due to gender-stereotyping cluster together in MT systems, with words being neighboured by semantically similar words (Gonen and Goldberg, 2019: 610, 614). For example, the male-biased profession ‘doctor’ will cluster with words like ‘banker’ or ‘coach’, whereas female-biased professions like ‘hairdresser’ will cluster with and neighbour words like ‘nanny’ or ‘librarian’. The MT systems learns this and clusters these words accordingly from the training data it is being fed, in which doctors may often be male, and hairdressers female. This naturally biased data leads to a pre-empted, biased MT system. MT engines over-bias words by forming such biased word clusters. If, for example, a MT system is trained on a very small dataset with only one lawyer being mentioned and this lawyer is male, the MT engine will learn this information and will likely depict every lawyer in the output as male (like the overamplification of the red apples example mentioned previously). Algorithmic discrimination as seen in MT systems is therefore likely to happen by

> "associating one implicitly gendered term with other implicitly gendered terms, or picking up on gender-specific regularities in the corpus by learning to condition on gender-biased words, and generalizing to other gender-biased words" (Gonen and Goldberg, 2019: 614).

Concerns about bias in MT have led to an increasingly "growing body of work on fairness in machine learning" (Romanov et al., 2019: 4187–4188). The following section outlines several methods proposed in recent research with the aim to reduce gender bias in MT.

2.3 Proposed Methods to Reduce Gender Bias in MT

Within the field of bias in MT, gender-bias is the most researched aspect, and several approaches to tackle and find solutions to address gender-bias in translations have been proposed. However, Shah, Schwartz and Hovy (2020) agree that methods proposed to reduce bias in MT for one specific bias often do not apply to other biases. Recent research include adding gender information in the process of training (Vanmassenhove, Hardmeier and Way, 2018), debiasing word embeddings (Font and Costa-jussà, 2019), adding the previous sentence (PreSent) and concatenating two sentences with a separator token as well as speaker information (SpeakerID) by adding the gender tag before each sentence (Basta, Costa-jussà and Follonosa, 2020), or by training a very small but gender-balanced NMT system (Saunders and Byrne, 2020). The last approach of training a small NMT system on more gender-balanced data was taken in this research.

Research in this area is leading machine translation to take stylistic considerations into account, which has still been lacking due to machine translation primarily focussing on conveying the correct content (Hovy, Bianchi and Fornaciari, 2020: 1686). The development of MT has so far predominantly focussed on translating 'what' is being said, rather than 'how' something is being said (ibid.). With continued awareness and research in 'how' MT systems translate, demographic and other aspects of language can help personalise and unbias MT.

Based on manual analysis and evaluation, Basta, Costa-jussà and Follonosa (2020: 100) found that adding both a previous sentence for context as well as speaker information to the training of an MT system helps towards "named entity disambiguation", reducing gender-bias, as well as leading to an overall improvement of "morphological agreement and quality of translation style" (ibid.: 101). By evaluating and comparing the accuracy, Basta, Costa-jussà and Follonosa (2020) show that the PreSent methodology of adding a previous sentence for context mostly detects the gender correctly, more so than adding a gender tag.

Bolukbasi et al. (2016: 1) show that word embeddings (from a monolingual English corpus) exhibit gender stereotypes to a "disturbing extent" and provide a methodology to modify such embeddings to remove gender stereotypes. In their research, Bolkbasi et al. (2016: 8) identify a gender direction of word embeddings that captures gender. They show how word embeddings are naturally classified, clearly separating gender-specific words from gender-neutral words (ibid.: 11,13) This classification is depicted in figure 8 below.

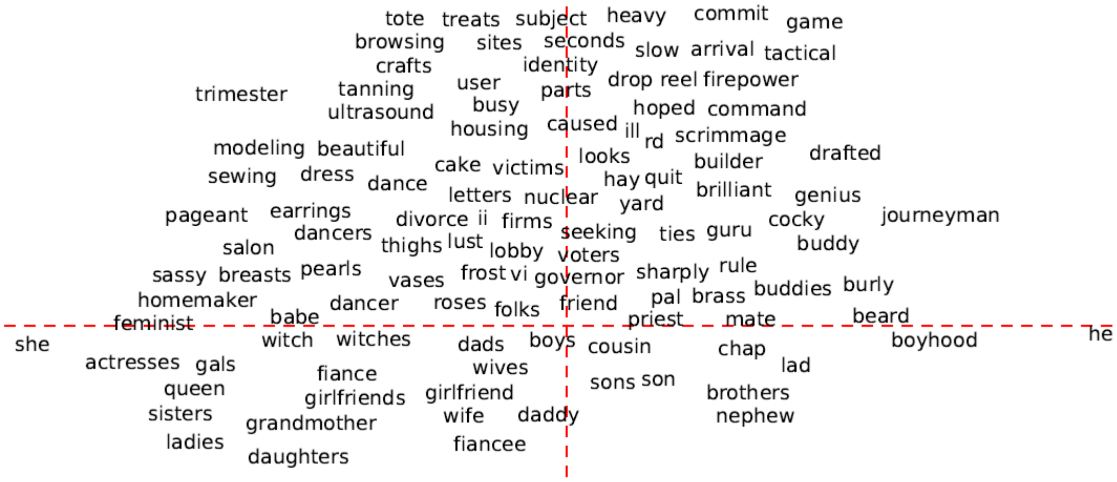
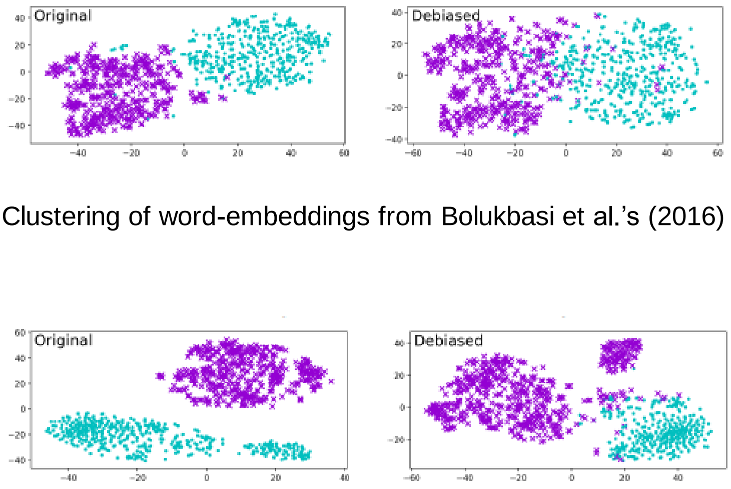


Figure 8. Classification of gender-neutral and gender-specific words. (Bolukbasi et al., 2016: 11)

In figure 8, gender-specific words like ‘queen’ or ‘brother’ are classified below the horizontal line, and gender-neutral words like ‘dancers’ or ‘genius’ are classified above the line. Words to the left are those associated closest to ‘she’ and words to the right are associated to ‘he’. In their research, Bolkbasi et al. (2016: 14–15) developed two algorithms to ‘debias’ such word embeddings by removing gender associations from gender neutral words. Through hard debiasing, gender-bias in word embeddings could be significantly reduced while related concepts were still clustered (ibid.: 15). In 2018, Zhao et al. built on Bolukbasi et al.’s (2016) research by demonstrating that gender information in word embeddings can be isolated without sacrificing the functionality of the embedding model, where the gender feature is a ‘protected attribute’ (2018: 4847).

A few years later, Gonen and Goldberg (2019: 614) found that the very interesting and seemingly convincing proposals to reduce gender bias in word embeddings are actually superficial. Reduced gender bias can be clearly evaluated in the output translations but, in reality, the bias is hidden rather than removed. According to certain proposed methods, gender bias can be reduced in word embeddings in the post-processing step either through hard de-biasing or soft de-biasing, as proposed by Bolukbasi et al. (2016), or as part of the training procedure, as proposed by Zhao et al. (2018). However, gender bias is still reflected in the distances between inherently gender-neutral words like professions (Gonen and Goldberg, 2019: 609).

In the following figures, Gonen and Goldberg (2019: 612) showed that the intended techniques by Bolukbasi et al. (2016) and Zhao et al. (2018) to decrease gender bias in MT, which might have superficially worked in the translation outputs, have not worked on a word embedding level. Without going into the specifics of Gonen and Goldberg’s (2019) experimental set-up to demonstrate the continued underlying bias in MT systems, even after methods to decrease or remove gender bias were applied, the graphs in figure 9 show that systematic gender bias in the MT system remains.



(a) Clustering of word-embeddings from Bolukbasi et al.'s (2016) research.

(b) Clustering of word-embeddings from Zhao et al.'s (2018) research.

Figure 9. Clustering of word-embeddings before and after applying intended debiasing methods (Gonen and Goldberg, 2019: 612).

In both depictions, the graph on the left-hand side represents the word-embedding clusters of the original MT system before applying the debiasing method, and the graph on the right-hand side represent the word-embedding clusters of the 'debiased' MT systems after applying the debiasing method (either as a post-processing step or integrated in the training step). The graphs clearly show that even after applying potential debiasing methods, the word-embeddings within the MT system still largely cluster.

That means that bias may not be visible or measurable in the output translations anymore (with seemingly unbiased translations being produced) but bias information actually remains embedded in the representation after 'debiasing' (Gonen and Goldberg, 2019: 612). Bias is therefore still manifested by semantically related words such as professions being socially marked (ibid.). Feminine-marked word-embeddings such as 'nurse' or 'caregiver' will continue to remain clustered closer to words like 'secretary' or 'teacher', whereas male-marked word-embeddings such as 'pilot' will continue to be clustered closer to 'boss'. A more detailed categorisation of for what words and in what context, gender bias occurs in MT is outlined in section 3.

Good attempts have been made at debiasing MT systems, and translation outputs superficially seem unbiased. However, the underlying MT system still contains bias information by clustering word-embeddings in a biased manner, as shown in figure 9. Further research is needed to find possible solutions in not just superficially removing bias from translation outputs, but rather removing bias information in the MT system as a whole.

Vanmassenhove, Hardmeier and Way (2018) aimed at decreasing gender bias in MT by adding a speaker-gender tag ('FEMALE' or 'MALE') to the training data. Their results show improvements, achieving higher BLEU scores for the NMT systems trained with their gender-tagged data, in comparison to an NMT system trained on an un-tagged data set. Interestingly, this did not yield an improvement for German. Non-automated manually evaluated assessments could potentially help analyse the outputs further.

Hovy, Bianchi and Fornaciari (2020) empirically show how translations affect the demographic profile of a text, finding that in MT translations authors sound on average older and more male. Interestingly, the measured gender skew appears strongest for German with all tested systems producing estimates that are a lot more female than the original data (ibid.: 1688). This goes against their hypothesis and was a surprising finding. Hovy, Bianchi and Fornaciari (ibid.: 1689) specifically found that translations into English sound older and more male than in other languages, which might be due to large uneven data for English.

According to Shah, Schwartz and Hovy (2020: 5256), one of their mentioned bias categories, semantic bias, can be reduced by adjusting the parameters of the embedding model to reflect a more accurate target distribution. Romanov et al. (2019) explore countermeasures to reduce gender bias in MT by reducing the correlation between the professions of people and the word embedding of their names without relying on protected attributes such as age, race, or gender. Romanov et al. (ibid.: 4195) does not require a specification of which biases should be mitigated but rather allows for the mitigation of multiple biases simultaneously.

Saunders and Bryne (2020) took a different approach to reduce gender bias in machine translation. Their approach is similar to the approach taken in this dissertation. In their research, Saunders and Byrne (ibid.) treat gender debiasing as a domain adaptation problem and manually created a very small dataset including professions (that are often stereotyped for genders) to train a NMT system. Their conviction is that a "small, trusted gender-balanced set could allow more efficient and effective gender debiasing than a larger, noisier set" (ibid.: 7724). The issue that arises, however, is that improvement on the gender-debiased domain, by training on a very small dataset, correlates with a lower translation quality. Nevertheless, continuously training an NMT system on a much smaller but gender-balanced dataset does yield consistent improvements in gender-debiasing (ibid.: 7725). In their research, Saunders and Byrne succeed in minimising 'catastrophic forgetting', the decrease in translation quality, by taking either of the two approaches called Elastic Weight Consolidation (EWC) or lattice rescoring for NMT, both of which allow for gender debiasing while maintaining translation quality (ibid.: 7733). This enables Saunders and Byrne to experimentally show that gender bias in NMT systems can be reduced by continuously training them on a small, handcrafted gender-balanced dataset, without having an overall decrease in translation quality. In the research conducted for this dissertation, similarly, a NMT system was trained on small datasets that were differently gender-balanced, to show that a more balanced dataset yields a more balanced translation. However, as Saunders and Byrne had warned, the improvement in gender-balance comes at the expense of translation quality. More detail on this can be found in the analysis section 5.

3 Language Patterns and Gender

This section covers the linguistic phenomena of which language patterns and trigger words assume a certain gender in machine translation when translating from English to German. Compared to English, German is a gendered and morphological language, where professions and adjectives are gendered. Since datasets, with which machine translations are trained, are often inherently gender-biased, machine translation outputs from English to German from such MT systems reflect this bias. The types of gender-bias present in datasets, and therefore also in resulting MT translations, are diverse, but patterns can be drawn.

Three such patterns for binary-gender will be discussed in this section, in 3.1, 3.2 and 3.3 respectively: plural generic masculine, direct trigger words like professions and adjectives, and indirect trigger words affecting the context of sentences. The first, generic masculine, is the focus of this dissertation's research, with the experiments training different MT systems and then evaluating the English-German translations by analysing whether the generic masculine was used, or whether the feminine term is used as well. The first two patterns are quick to demonstrate gender-bias in machine translation and have been the most researched. The third pattern distances itself from direct trigger words that lead to a gendered-translation but rather focus on specific words in the sentence that indirectly affect whether translated sentence will contain gender-bias.

Gender shapes language much more than simply through the expression of a (binary) gendered noun referring either to a man or a woman. The way men and women use language in terms of word choices or syntactical constructions to convey their context can be very different (Vanmassenhove, Hardmeier and Way, 2018: 3003). Language expresses much more about an individual than the direct gendered-terms visible in MT: "[t]he language that we produce reflects our personality, and various personal and demographic characteristics can be detected in natural language texts" (Rabinovich et al., 2017: 1074). This reflection of personality and personal and demographic traits is highly reflect in MT.

This section focusses on the linguistic phenomena that can be observed as to how machine translation produced gender-biased translation from language texts.

3.1 (Plural) Generic Masculine

This section covers the common use of the plural generic masculine in German, and section 3.2 will cover singular generic masculine in the form of professions. The Duden (Dudenredaktion, 2022a) outlines that in German, the generic masculine plays a major role in gender-neutral language. In English, 'ministers' refers to both male and female ministers, equally in German 'die Minister' was traditionally used to refer to both male and female ministers as it is the generic masculine, which until recently has been prominent in German language. In recent years, however, emphasis has been placed on a gender-neutral language and the generic masculine is, by the general public, no longer considered to be the correct formulation as it is not gender-neutral. Next to ‘Minister’ being the generic masculine, German is a gendered language, so the term ‘Ministerinnen’ refers to female ministers. In an effort to create a more gender-fair language, emphasis has been placed on referring to ‘ministers’ as ‘Ministerinnen und Minister’ since ‘Minister’ could be a generic masculine, but it could also simply refer to male ministers, this being unclear to a reader. By including the plural female term ‘Ministerinnen’ and not opting for the generic masculine, the reader directly understands that both female and male ministers are included, and the image conveyed to the reader will be of both female and male ministers. Subconsciously, when reading ‘Minister’, a reader might just picture male ministers even though women were also included. By including female ministers in the choice of words, the reader’s association will be more gender neutral. The aim of gender-neutral language is to convey gender-neutral associations to the reader, with the hope of forming a gender-fair society.

To help create a more gender-neutral language, at least for binary gender, numerous alternatives have been presented in the past few years. The Duden (Dudenredaktion, 2022b) gives an overview of these alternatives. A norm for the one correct form to gender in German language does not yet exist, resulting in the diverse use of all these alternatives. As mentioned above, in paragraph 106 (1) the Duden (ibid.) outlines the naming of both male and female terms such as ‘Kolleginnen und Kollegen’ for ‘colleagues’, which is a very common and formal choice of gendered-language. In written text this can sometimes be portrayed as ‘Kolleginnen/Kollegen’. In paragprah 98 (2) the Duden (ibid.) outlines two further alternatives through the use of a ‘/-‘, or brackets: ‘Minister/-innen’ or ‘Minister(innen)’. These first three alternatives are represented in the German official rules and regulations used by German authorities. They do not, however, include non-binary gender (ibid.).

Not represented in the German official rules and regulations are four other alternatives. These are, however, most commonly used by individuals, academics, or news portals in Germany as they aim to be completely gender-neutral by referring to both binary and non-binary gender. As outlined by Duden (Dudenredaktion, 2022b), these four versions are the following:

4. Using a gender star: Minister*innen
5. Using a capital ‘I’ within the word: MinisterInnen
6. Using a gender gap (underscore, colon): Minister_innen, Minister:innen
7. Using a slash without a dash: Minister/innen

In practice, it can be seen that versions 1, 2 and 3 (using a colon) are becoming more and more prominent. The gender star or colon dominantly represent gender-neutrality for both binary and non-binary gender, with the colon frequently being used for web-based texts since it can be detected by machines, for example for text-to-speech applications.

Focus of this research lies on the pattern of whether a plural generic masculine was used in a machine translation or whether both genders were explicitly stated. The specific focus of this research is the use of both terms, ‘Kolleginnen und Kollegen’ as well as ‘KollegInnen’, which are the most represented in the chosen dataset and further discussed in section 4.3.3. A few examples are provided here to showcase how commercial machine translation systems such as Google Translate and DeepL translate English gender-ambiguous terms into German.

Tool	English source sentence	German MT target sentence
Google Translate	My colleagues started the meeting on time.	Meine Kollegen haben pünktlich mit dem Meeting begonnen.
DeepL		Meine Kollegen begannen die Sitzung pünktlich.

Table 1. An example of how DeepL and Google Translate translate ‘colleagues’, which could refer purely to male, or to male and female colleagues, into German. Both MT systems opt for the generic masculine.

Apart from the fact that the MT tools render a different translation in German of the same English source sentence, as seen in table 1, both tools chose the plural generic masculine for the term ‘colleagues’: ‘Kollegen’.

When simply typing “My colleagues” into Google Translate, the German translation yielded is “Meine Kollegen”. Equally, DeepL yields “Meine Kollegen” as the main choice but it offers two alternatives below: “Meine Mitarbeiter” and “Meine Kolleginnen und Kollegen”. This shows that the first choice for both MT tools is the plural generic masculine. Even the second DeepL choice of “Meine Mitarbeiter” is another form of a plural generic masculine for “My colleagues”. Only as a third option, we get a plural female term with both genders being directly included. At least, however, DeepL gives this as an option, whereas Google Translate only offers “Meine Kollegen”.

This is just one example of the plural generic masculine generally being chosen as the primary translation by MT software. The research demonstrates that this choice directly relates to the dataset used to train such a MT software. If the training data is male-skewed (gender-biased), the translations produced will also be gender-biased, whereas if the training data is less male-skewed, the translations produced will also be less male-skewed and less gender-biased. The aim being to create a gender-neutral, or equally skewed translation. The experimental set-up, results and analysis of this research will be discussed in sections 4 and 5.

3.2 Professions and Adjectives

Next to the generic masculine being most commonly chosen for plural terms leading to gender-bias in MT outputs, the translations of professions also exhibit gender-bias. Commonly, machine translations of professions in their plural form, like ‘doctors’, are translated into the plural generic masculine in German, like ‘Ärzte’. Next to the generic masculine choice for plural terms, however, gender-bias is greatly exhibited in reference to certain professions. Opting for a plural generic masculine, as outlined in section 3.1, had until recently not been wrong since it was accepted and recognised in the German language to use the generic masculine when referring to both genders in the plural form. However, wrong translations of professions based on stereotypes in society are more problematic. A few examples of how DeepL translates certain English sentences into German, where the profession could refer to either a man or a woman, are outlined in table 2 below.

English source sentence	(DeepL) Machine-translated German sentence
The **doctor** saved the patient's live.	Der **Arzt** rettete das Leben des Patienten.
The **nurse** saved the patient's live.	Die **Krankenschwester** rettete das Leben des Patienten.
The **soldier** returned home.	Der **Soldat** kehrte nach Hause zurück.
The **feminist** went on strike.	Die **Feministin** streikte.
The **pilot** arrived late.	Der **Pilot** kam zu spät.
The **flight attendant** arrived late.	Die **Flugbegleiterin** kam zu spät.

Table 2. An example of how DeepL translates English sentences containing gender-ambiguous professions into German. The professions are represented as highly stereotyped in the German machine translation.

In table 2, each of the professions or terms could refer to either a man or a woman. Yet, the gender roles appear to be clearly pre-defined, resulting in a certain choice of gender of each of the professions in the machine translated output. ‘Doctor’ is most commonly translated as ‘Arzt’, whereas ‘nurse’ is most commonly translated as ‘Krankenschwester’. These gender-stereotypes are not specifically created by machine translation tools. The stereotypes and resulting gender-biases stem from the data that is used to train these MT systems: predominantly due to stereotyping of certain professions in society. Today, greater emphasis is placed on roles being filled more equally by men and women. However, this has not always been the case and, traditionally, certain professions have been filled by men, such as ‘doctors’ or ‘engineers’, whereas certain other professions have been traditionally filled by women such as ‘nurses’ or ‘flight attendants’. Due to this traditional gender-imbalance in professions, a gender-bias exists in datasets used to train MT systems, which in turn is reflected in MT outputs, as exemplified in table 2 above.

Next to professions, adjectives also directly impact the choice of gender in a machine translation, as shown in table 3 below.

English source sentence	(DeepL) Machine-translated German sentence
The **famous** lawyer said hi	Der berühmte **Anwalt** lässt grüßen.
The **pretty** lawyer said hi.	Die hübsche **Anwältin** lässt grüßen.
The **arrogant** dancer performed a successful show.	Der arrogante **Tänzer** führte eine erfolgreiche Show auf.
The **graceful** dancer performed a successful show.	Die anmutige **Tänzerin** führte eine erfolgreiche Show auf

The **bright** student was accepted to college.	Der begabte **Student** wurde am College angenommen.
The **beautiful** student was accepted to college.	Die schöne **Studentin** wurde am College angenommen.

Table 3. An example of how DeepL translates English sentences containing adjectives next to professions/nouns into German. The adjectives noticeably affect the MT-choice of female or male nouns in the German machine translation.

Just as certain professions are machine translated showing a clear gender bias, certain adjectives equally lead to obvious gender bias in machine translations. As depicted in table 3, certain adjectives regarding the skill of a person, for example, often lead the machine translation to opt for a male gender, whereas adjectives referring to the look of the person lead to a female choice of gender. Women are often mentioned in contexts, where the individual is referred to being ‘pretty’, ‘beautiful’, ‘sensitive’ or ‘caring’, whereas men are often mentioned in contexts where the individual is referred to being ‘successful’, ‘intelligent’, ‘capable’ or ‘arrogant’.

Adding an adjective next to a profession has also been termed “Fighting bias with bias” (Stanovsky, Smith and Zettlemoyer, 2019: 1682). If normally, a MT system would opt to translate ‘lawyer’ to ‘Der Anwalt’, simply by adding a ‘pretty’ in front of lawyer, the translation output will become ‘Die hübsche Anwältin’. Just by adding certain adjectives in front of professions or nouns, MT systems will opt for different genders. For example, DeepL translates “My **arrogant coworker** brought me coffee” to “Mein **eingebildeter Mitarbeiter** brachte mir Kaffee”, whereas “My **devoted coworker** brought me coffee” was translated to “Meine **treue Mitarbeiterin** brachte mir Kaffee”.

Once again, these gender-biases most likely stem directly from the inherently biased data used to train a MT system.

3.3 The Context of Sentences

This section covers the influence of certain trigger words that skew the MT sentence to be male or female biased. Next to adjectives, other words, such as verbs or nouns, can directly influence how the MT system perceives the gender in a sentence. Table 4 below depicts a few examples of English gender-ambiguous source sentences and their relevant translations into German (by DeepL).

English source sentence	(DeepL) Machine-translated German sentence
My **co-worker works** all day.	Mein **Kollege arbeitet** den ganzen Tag.
My **co-worker gossips** all day.	Meine **Kollegin quatscht** den ganzen Tag.
My roommate **cleaned the car**.	Mein **Mitbewohner** hat das **Auto** gereinigt.
My roommate **cleaned the flat**.	Meine **Mitbewohnerin** hat die **Wohnung** geputzt.
My **neighbour** goes to the **gym**.	Mein **Nachbar** geht ins **Fitnessstudio**.

My **neighbour** goes to **yoga**.	Meine **Nachbarin** geht zum **Yoga**.
The **professor** is an expert on **machine translation**.	Der **Professor** ist Experte für **maschinelle Übersetzung**.
The **professor** is an expert on **gender bias** in machine translation.	Die **Professorin** ist Expertin für **Gender Bias** in der maschinellen Übersetzung.

Table 4. An example of how DeepL translates English sentences containing a gender-ambiguous profession/noun and an indirectly related trigger word such as ‘car’ or ‘yoga’ into German. The trigger words highly affect the MT-choice of female or male nouns in the German machine translation.

Table 4 outlines just a few, of countless, examples where certain trigger words skew the sentence to either refer to a man or a woman. In the examples above, the individual referred to such as ‘neighbour’ or ‘co-worker’ could either be male or female. Depending on the context these words are placed in, the MT system clearly skews these terms to be translated as either male or female in German. If the co-worker ‘works’, the MT system depicts the co-worker as a man, but if the co-worker ‘gossips’, the MT system depicts the co-worker as a woman. Very interesting is the last example, where an expert professor on ‘machine translation’ is depicted as a male professor in the German machine translation, an expert professor on ‘gender bias in machine translation’ is depicted as a female professor in the German machine translation. The context in which an individual is placed therefore directly influences the MT system’s choice of gender for that individual.

There are countless of examples demonstrating the presence of stereotyping and gender-bias in machine translation outputs. The examples outlined here are simply a small sample to highlight this issue. These occurrences of stereotyping and gender-bias in machine translation outputs directly arise from the datasets with which the MT systems are trained, which in turn are a direct reflection of choice of language in society. Specifically, the use of plural generic masculine instead of both the female and the male plural terms is analysed in this research, showing that it is directly linked to how gender-biased the training data is. By skewing the training datasets to becoming more gender-neutral, machine translations will automatically reflect this and also become more gender-neutral. In turn, this will hopefully help create a more gender-neutral use of language in society.

4 Experimental Setup

In this section, the experimental setup will be explained step by step, outlining the platforms needed, the choice and preparation of datasets, the choice and training of the NMT models, and how the dataset was skewed to show differences in gender occurrences (biasing). The following section, section 5, will cover the analysis of the results of this research.

For this research, one bilingual dataset available for download online was used to train an NMT model from scratch. This dataset was then split up and skewed, as further outlined in section 4.3.2, 4.3.3 and 4.3.4, to train different models and obtain different translation outputs from each model. Without manipulating the dataset, smaller samples of the main dataset were used that contained either more or less gendered sentences. An NMT model was then trained on each of these smaller datasets to analyse whether or how much of a difference the presence of gender makes in the translation output based on how many gendered sentences were used to train the model in the first place.

Recommendations for ethics in AI suggested that social biases in datasets need to be addressed before training the NMT model on said datasets (Saunders and Byrne, 2020: 7725). This research therefore does not attempt to debias a NMT model once it has been trained but rather focusses on balancing the datasets from the beginning. Not enough data is available to train a model on a fully gender-balanced dataset and still produce qualitative-acceptable translations. However, the difference in training a NMT model from scratch on a more or a less gender-biased dataset could be shown.

4.1 Hypothesis

The hypothesis of this research is that if there is gender-bias inherent in a dataset used to train an NMT system, translation outputs will reflect this gender-bias. More specifically, the presence of gender in the data used to train an NMT model makes a measurable difference on the presence of gender in the output translation based on said model. The aim is to show that by having more or a less gendered training data for an NMT model, the output translations would equally be more or less gendered.

4.2 Platform

As specified in section 1.7, the two key platforms/toolkits used for this research are Google Colab Pro[7] and OpenNMT-py[8] (Klein et al., 2017). OpenNMT-py is the open-source toolkit available to train NMT models from scratch as used in this research. How this worked will be closer described in section 4.4. As outlined in section 1.7.3 the online interface Google Colab Pro was used for this research to execute code and save datasets, model steps and outputs files in Google Drive.

[7] https://colab.research.google.com/signup

[8] https://github.com/OpenNMT/OpenNMT-py

4.3 Datasets & Preparation

As mentioned by in section 1, huge publicly available bilingual corpora, as those needed to train NMT systems are very scarce and therefore difficult to come by. These corpora usually need to contain millions of sentences to yield good translation results. There are a common handful of bilingual training data that are frequently used by researchers for the topic of MT.

4.3.1 Finding Datasets

For the research conducted in this dissertation, the English-German Europarl dataset[9] (Koehn, 2005) was used. This parallel corpus is 189MB in size and contains a total of 1,920,209 sentences per language, 44,548,491 German words and 47,818,827 English words. The Europarl corpus is extracted from the proceedings of the European Parliament and includes 21 European languages (ibid.). The Europarl parallel corpora are ideally suited to train an NMT system because the sentences have been pre-aligned so that per line, each sentence from one language corresponds to that same line to the sentence in the respective other language. Sentence alignment has already been done using a pre-processor to identify sentence boundaries and by using a sentence alignment tool called Church and Gale algorithm (ibid.). Sentence alignment is an important and necessary step for training data for NMT systems (Koehn, 2017: 108).

Aside from its online availability and sentence alignment, however, the Europarl parallel corpus has two characteristics that make this research on analysing the effects of gender bias specifically difficult. One reason is that the Europarl corpus consists of formal spoken language, since it is compiled from sessions by the European Parliament, meaning that it “does not contain many sentences using the first-person singular pronoun” (Vanmassenhove, Hardmeier and Way, 2018: 3006) and generally does not reflect the informal type of language used in day-to-day conversations. Secondly, shaped by its source – predominantly male speakers at the European Parliament – there is a very noticeable “gender unbalance in the Europarl dataset, which will be reflected in the translations that MT systems train on this data produce” (ibid.: 3005). These two aspects make the use of the English-German parallel corpus quite difficult for this specific research focussed on gender in language. After skewing the dataset and using rather small samples, the desired difference in gender could be noticed.

4.3.2 Data Preparation

As stated above, the Europarl parallel corpus was already pre-aligned so that each sentence in the English dataset corresponded on the same line to the German dataset. For this research, English was the designated source language and German was the designated target language. The source language was chosen to be German so that

[9] https://www.statmt.org/europarl/

occurrences of the female plural form such as ‘Kollegen und Kolleginnen’ in comparison to the generic male could be measured in the different output translations.

The English-German Europarl parallel corpus was used as the bilingual dataset to train the NMT model. This bilingual dataset, both the English and the sentence-aligned German dataset, were saved in Google Drive. In reality, three datasets are needed to train and test a MT model. One dataset is needed to actually *train* the model, which is called the ‘train data’. Another much smaller dataset is needed for *validation*, to validate the training steps at certain intervals (as described in section 1), which is called the ‘validation data’ or ‘val data’. Lastly, another smaller dataset is needed to then test the model and produce a translation. This is called the ‘test data’ and is an English text that will then be translated into German by each MT model. These three datasets – train, val and test – must all originate from the main parallel corpus. The test dataset could also be a different, out-of-domain, dataset but, this is more feasible for MT systems trained on larger training corpora with more vocabulary that have a higher likelihood of translating text that may be very different to the text used to train the model.

To train a MT system, the main parallel corpus is therefore usually split into three sub-datasets where the train data makes up around 75%, and both the val and the test data make up around 12.5%, as depicted in figure 10. For this research, the train data for each model made up around 76% and the validation and the test sub-dataset each made up around 12%. No matter the size of the dataset, no matter how many sentences are used to train the data, the split by percentage should always be around the same. Especially, if models are compared using different datasets, as in this research, the split for train, val and test data must always be equal for each model.

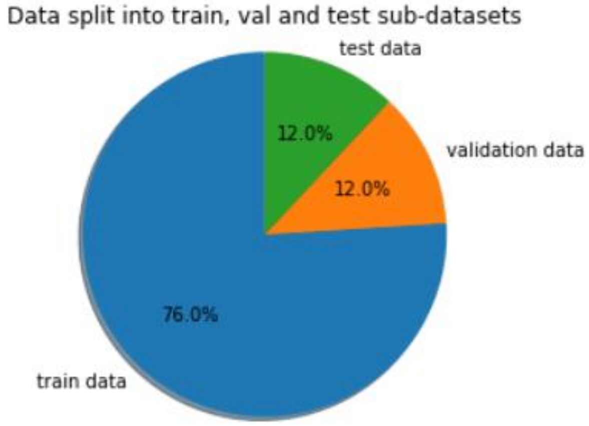


Figure 10. How the data is split per percentage: train data, validation data, test data.

Naturally, the bigger the datasets are, the more likely the MT model will learn well, based on more vocabulary as input and produce good translation outputs. The aim is to have the train data be the largest sub-dataset to actually train the MT model. The bigger the validation data, the better the train data can be validated, which improves the model and translation quality. Since the size of the val data directly impacts the train data, however, the val data is kept smaller, to allow for a maximisation of the train data.

For this research, the English-German corpus was also split into these three sub-datasets, the exact length of which will be described in the next section. All sub-datasets were created for both English and German, where the English and German train and validation datasets were used to train the NMT model, and the English test dataset was used to produce a German translation output per model. The German test dataset was the reference dataset against which the translation outputs were then analysed using automatic evaluation metrics. Naturally, each of these sub-datasets was also sentence-aligned. Empty lines were removed from the datasets for clarity and more efficient training. The corpus is split up into these sub-datasets by coding on a Google Colab notebook and they are then saved in the Drive folder. During the actual NMT model training, these sub-datasets can be accessed by linking the Colab notebook to the Drive folder in which they are saved.

4.3.3 Gender in the Dataset

Together with section 4.3.4, this is likely the most crucial section to understand the decision made in this research of how to train the NMT models. This section describes how the sub-datasets were compiled. As described, this research aims to analyse how the choice of training data affects the translation in terms of gender. For this purpose, different smaller training datasets were compiled to train an NMT model from scratch and then let it translate a test text to analyse and compare the yielded translations. Since the Europarl corpus language is very formal and there is a large gender unbalance, the female plural form was measured as the gender-factor.

For example, whenever a speaker said:

“Dear colleagues….”

or a similar gender-ambiguous phrase in English, as explained in section 3.1, various German translations such as the following are possible:

“Liebe Kollegen …” (the generic masculine),

“Liebe Kollegen und Kolleginnen…“,

“Liebe Kolleg:innen…”,

“Liebe KollegInnen…”,

“Liebe Kolleg*innen…” and

“Liebe Kolleg/-innen…“,

to name some of the most common ones. In the German Europarl dataset, either due to different speaker formulations or due to different translators, three gender alternatives were present: “Kollegen und Kolleginnen”, “KollegInnen“, and “Kolleg / innen“ but the first one was by far the most common one, with the second one being the second common. This research therefore focussed on the female plural phrases such as “Kollegen und Kolleginnen” and “KollegInnen”. All German female plural forms, instead of the generic masculine, that could stem from ambiguous English terms such as ‘citizens’, ‘Europeans’, ‘athletes’ etc. present in the Europarl corpus were included.

Unfortunately, there is no easily applicable online available model to computationally filter for such female forms in German. A language model to filter many aspects is SpaCy[10] but this would filter both ‘Kollegen’ and ‘Kolleginnen’ equally as nouns (NN) or both female and male pronouns equally as pronouns. Therefore, filtering the datasets for female plural forms was done without applying an existing model. It was done by computationally filtering (using a regex function) the German dataset for all words that end in ‘innen’. This step was done computationally, meaning that the datasets were filtered at a high speed. Naturally, however, this included words such as ‘beginnen’ or ‘gewinnen’ that also end in ‘innen’. After manually looking at all words ending in ‘innen’ present in the Europarl corpus, it quickly became visible that often the same words are frequently used such as “Kollegen und Kolleginnen” for ‘colleagues or “Bürger und Bürgerinnen” for ‘citizens’. This is due to the fact that the language in the Europarl corpus is rather formal and the content covers limited topics.

Therefore, the German dataset filtered for words ending in ‘innen’ was manually skimmed through and all words that were female plural forms were noted down. Different words and spellings of that same word such as ‘Kolleginnen’, ‘KollegInnen’ and lowercase ‘kolleginnen’ were taken into account, leading to a total of around 243 different female plural words ending in ‘innen’. The German dataset was then computationally filtered (using the regex function) for this specific list of words. There likely is a deviation of filtering out all female plural forms but this deviation is very small. A total count of female plural words and of sentences containing such female plural words could be computationally filtered.

The German data was noticeably authored or translated by different individuals as there are different gender forms for ‘Kolleginnen’ present, as mentioned above, and mostly, the male plural term like ‘Kollegen’ is used as a general plural form without specifically including the female version. Just to give an idea, when looking at the complete German dataset of 1,920,209 sentences, only 1.53% of sentences include one or more female plural terms. That means that 98.47% do not contain a female plural term either because the male plural term was used or because those sentences in the English dataset generally do not contain gender-ambiguous terms such as ‘colleagues’ but might just be: “These programmes are funded by both government and industry”.

As mentioned above, the complete dataset was divided into three sub-datasets: train, validation and test datasets. In this train sub-dataset of 1,527,168 sentences, there were equally only 1.53% of sentences, so a total of 23,425, that contain one or more female plural terms. This is clearly not a large number and highlights the gender-imbalance of the German Europarl corpus. Obviously, taking into account the many neutral sentences in the corpus such as “Ihr Parlament wird dazu selbstverständlich konsultiert werden”

[10] https://spacy.io/models

that do not contain any gender. This small sample of sentences of mere 1.53% lead the overall sample of the data used to train the different NMT models in this research to be rather small. The following section 4.3.4 outlines how the different datasets were prepared to train different models.

4.3.4 Skewing the Dataset

This section directly connects to 4.3.3 and is crucial in understanding why certain outputs and results were obtained. For this research, five NMT models were trained from scratch with five different compiled train and validation sets. The test set is the same for all models so that in the end, the models are used to translate the same text so that the German translation outputs can be compared.

One of these models was trained with the complete dataset of 1,527,168 English and German sentences as the train data and 189,206 English and German sentences as validation data. This is the model, where 1.53% of sentences contain a female plural form and 98.47% do not. For this, the Europarl data has not been altered but simply split up into sub-datasets to show the natural gender imbalance in a German translation based on the original Europarl dataset and have a sacreBLEU[11], COMET[12] and BERTscores[13] to measure the other results by. This can be considered a 'baseline' model but the results of the other four models cannot be directly compared (due to different sizes).

All other four models contained an (unfortunately small) train dataset of 78,083 English and German sentences and a validation dataset of 9,370 sentences. In number, the val data is 12% of the train data but the sentences were not taken from the train dataset but rather from the big Europarl validation dataset. Clearly this number is a lot smaller than the original train dataset of 1,527,168: actually only 5.11% of the original train dataset. This is reflected in the final translations, which are of a noticeably lower quality, scoring lower evaluation scores due to numerous 'unknown' terms in the final translations. Nevertheless, the automatic evaluation scores are simply an interesting aspect, what actually matters, is the difference in how gender is reflected in each translation: the number of female plural terms present in each translation output for each model. And effectively, these four models give four very different responses. So, similar to the research conducted by Saunders and Byrne (2020), a very small (partially manually) selected dataset was compiled to train a NMT system. Their small, handcrafted dataset even just contained around 388 sentences (ibid.: 7728), which is an even tinier dataset than the ones used in this research. Just as this research, their research primarily focussed on reducing (eliminating) gender-bias in the output translations. To improve translation quality, Saunders and Byrne (ibid.: 7729) apply different approaches, something that is not covered in the scope of this dissertation.

[11] https://github.com/mjpost/sacrebleu

[12] https://github.com/Unbabel/COMET

[13] https://github.com/Tiiiger/bert_score

So, all four models have the same total amount of sentences per training dataset and per validation dataset to fairly compare the results. However, these datasets differ in the number of sentences containing female plural terms. The datasets per model with number of sentences used for each training and validation step that either contain or do not contain female plural pronouns are outlined in table 5 below.

Model	**Total number of sentences**	**Train data**		**Validation data (12%)**	
		Sentences containing female plural terms	**Sentences NOT containing female plural terms**	**Sentences containing female plural terms**	**Sentences NOT containing female plural terms**
Original Europarl model ('baseline', not to be compared with models 1-4)	Train dataset: 1,527,168 Val dataset: 189,206	1.53%	98.47%	1.52%	98.48%
Model nr. 1 *The hunnies*	Train dataset: 78,083 Val dataset: 9,370	0% 0	100% 78,083	0% 0	100% 9,370
Model nr 2. *The nineties*	Train dataset: 78,083 Val dataset: 9,370	10% 7,835	90% 70,248	10% 937	90% 8,433
Model nr. 3 *The eighties*	Train dataset: 78,083 Val dataset: 9,370	20% 15,617	80% 62,466	20% 1,874	80% 7,496
Model nr. 4 *The seventies*	Train dataset: 78,083 Val dataset: 9,370	30% 23,425	70% 54,658	30% 2,811	70% 6,559

Table 5. Showing all five NMT models, specifically the four that are to be compared, with the number of train and validation sentences they were trained with, and the number of sentences containing female plural terms. For quicker understanding, the columns referring to sentences with female plural terms are marked in pink.

Figure 11 depicts models 1-4 (*the hunnies, the nineties, the eighties,* and *the seventies*) to help visualise and understand the concept. The orange part of each pie chart refers to the sentences in the data that do not contain a female plural term, and the blue part of each pie chart refers to the sentences in the data that do contain a female plural term.

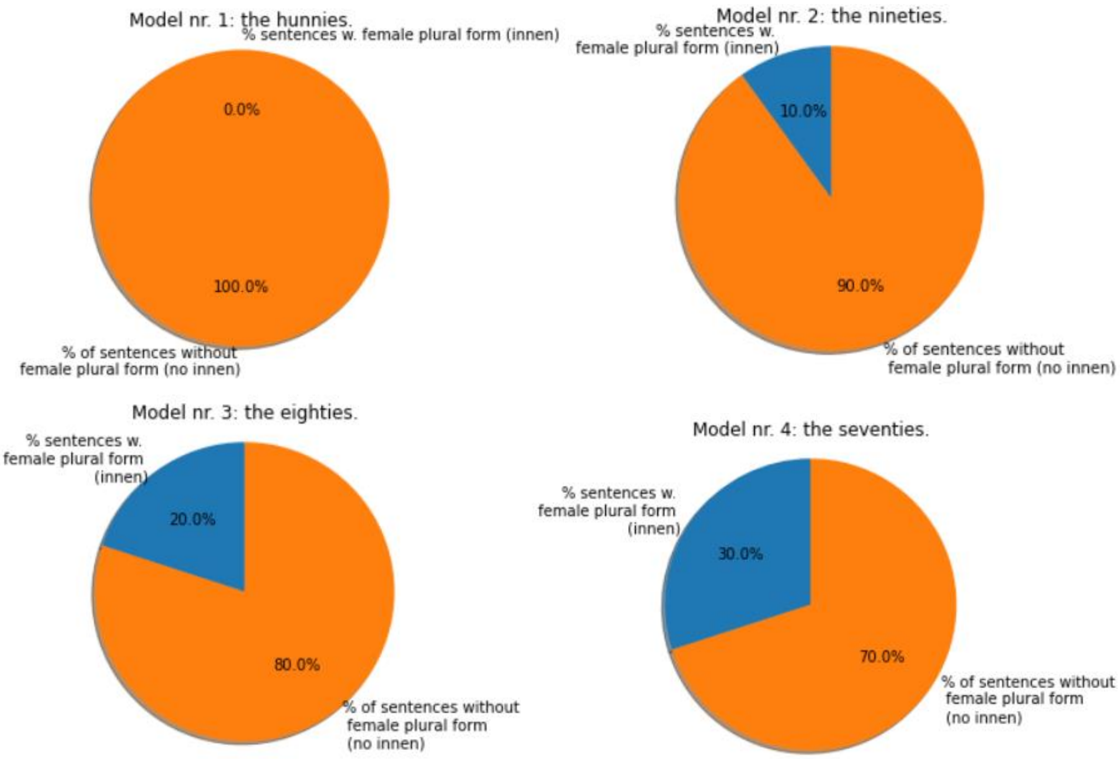


Figure 11. Depicts the four NMT models by sentence percentage either containing female plural terms or not containing female plural terms. All models have a total of 78,083 sentences in the training data and 9,370 sentences in the validation data.
For example, for model nr. 4 (*the seventies*), 70% of those sentences **do not** contain female plural terms and 30% of those sentences **do** contain female plural terms.

Once the above figures are understood, the next section outlines and explains the training procedure of the models.

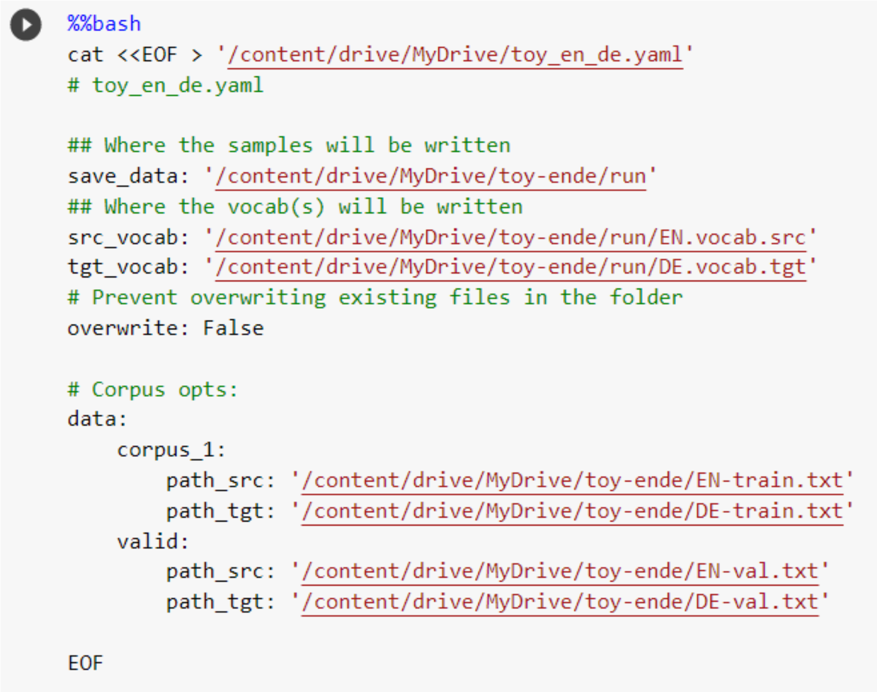
4.4 Training the Translation Model

In this section, specific focus will be placed on how the NMT models are actually trained with the above-mentioned data. The open-source OpenNMT-py toolkit (Klein et al., 2017) was used to train the NMT model, which provided pre-formatted code that then needs to be individually adapted and altered for each dataset. Each model is trained from scratch, meaning that it only trains on the data given from start of the training process. This research is based on the OpenNMT-py model that has "sequence-to-sequence encoder-decoders with LSTMs as the recurrent unit" (Vanmassenhove, Hardmeier and Way, 2018: 3005) with attention, as described in section 1.3 and specifically 1.7.2.

4.4.1 Outline of how an NMT Model is Trained

As mentioned before, this experimental coding was conducted on Google Colab Pro. Once the data has been prepared accordingly, which has been done separately and before training the NMT model, the code to train the NMT model is very compact, thanks to OpenNMT-py. Once the OpenNMT-py package has been downloaded (pip installed) in Google Colab notebook, the data can be prepared. The first step is to prepare a YAML configuration file. YAML is a powerful digestible data serialisation language often used to create configuration files with any programming language (Sharma, 2022), in this case with python.

In the figures below, all coding steps of how to train the NMT model will be shown and explained. The 'toy_en_de.yaml' are the yaml configuration files. Figure 12 below shows the beginning of the process, the code needed to build a YAML file. In the second line, such a YAML file is built. The next lines focus on where to save the data that will be produced. From the input training data, one file will be created only for English vocabulary (source language) and another file will be created for German vocabulary (target language).

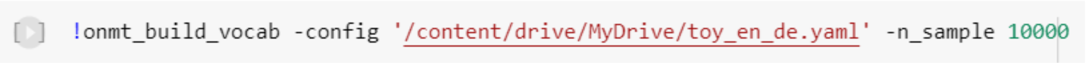


```
%%bash
cat <<EOF > '/content/drive/MyDrive/toy_en_de.yaml'
# toy_en_de.yaml

## Where the samples will be written
save_data: '/content/drive/MyDrive/toy-ende/run'
## Where the vocab(s) will be written
src_vocab: '/content/drive/MyDrive/toy-ende/run/EN.vocab.src'
tgt_vocab: '/content/drive/MyDrive/toy-ende/run/DE.vocab.tgt'
# Prevent overwriting existing files in the folder
overwrite: False

# Corpus opts:
data:
    corpus_1:
        path_src: '/content/drive/MyDrive/toy-ende/EN-train.txt'
        path_tgt: '/content/drive/MyDrive/toy-ende/DE-train.txt'
    valid:
        path_src: '/content/drive/MyDrive/toy-ende/EN-val.txt'
        path_tgt: '/content/drive/MyDrive/toy-ende/DE-val.txt'

EOF
```

Figure 12. Building a YAML file, creating paths to save the vocabulary per language, accessing the train and validation data per language.

The bottom half of figure 12 refers to accessing the training data by providing the paths to the English training data (source train data) and to the German train data (target train data), as well as the validation data by providing the paths to the English validation data (val train data) and to the German validation data (target val data). In the setup, these paths are the file paths to where the data is saved in the Google Drive folders.

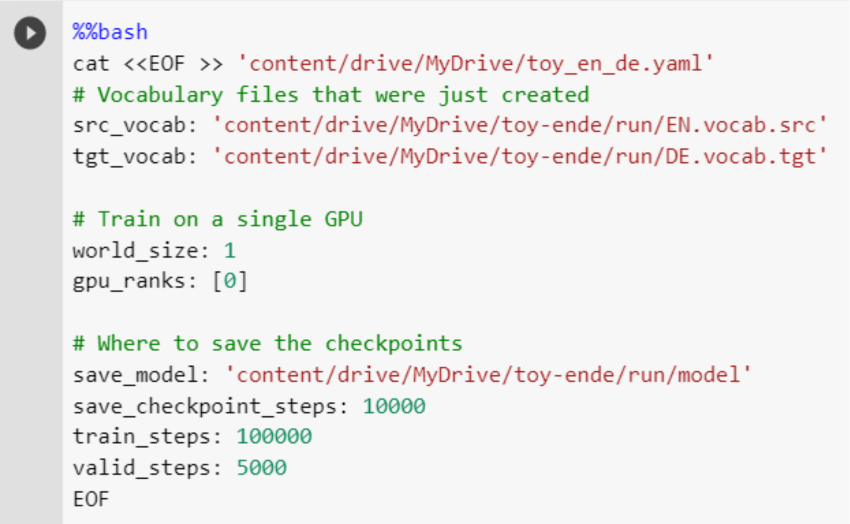
The next step, outlined in figure 13, is the configuration of the YAML file. So, in this step, the YAML file and the vocabulary (source and target) files will actually be created and saved to the relevant Google Drive folder.



```
!onmt_build_vocab -config '/content/drive/MyDrive/toy_en_de.yaml' -n_sample 10000
```

Figure 13. Configuration of YAML file. Creating the YAML file and the vocabulary files.

In figure 14, the paths to the source (English) and target (German) vocabulary are noted. This vocabulary is necessary to train the model. Furthermore, training specific parameters can be specified and changed. For example: 'gpu_ranks: [0]' actually means that there is one GPU available to train the model.



```
%%bash
cat <<EOF >> 'content/drive/MyDrive/toy_en_de.yaml'
# Vocabulary files that were just created
src_vocab: 'content/drive/MyDrive/toy-ende/run/EN.vocab.src'
tgt_vocab: 'content/drive/MyDrive/toy-ende/run/DE.vocab.tgt'

# Train on a single GPU
world_size: 1
gpu_ranks: [0]

# Where to save the checkpoints
save_model: 'content/drive/MyDrive/toy-ende/run/model'
save_checkpoint_steps: 10000
train_steps: 100000
valid_steps: 5000
EOF
```

Figure 14. Saving Files & Defining Model Parameters

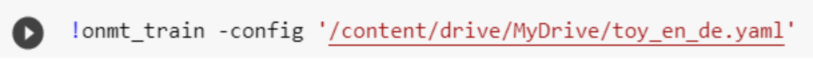
The bottom half shows a path where the model will be saved (in a Drive folder), how often checkpoints are saved, how many trainings steps should be taken, and how many validation steps should be taken.

The training parameters for each model used in this research were the following:

1. Save checkpoints every 10,000 steps,
2. Total training steps: 100,000,
3. Validation to be done every 5,000 steps.

The checkpoints can be saved in between in case the training crashes. The last checkpoint is then used to generate a translation. The higher the total training steps, the better the model will become and the better the quality of the output translation will become. Depending on the size of the training data, which for this research is quite small, at one point, the model will reach a point, where it can no longer be improved training on that same data (where overfitting could occur as mentioned in section 1). Similarly, the more frequent validation is done, the better the model will be validated, leading to a better quality.

Figure 15 depicts the last piece of code: training the model. Here, the YAML configuration file is used to train the model.



```
!onmt_train -config '/content/drive/MyDrive/toy_en_de.yaml'
```

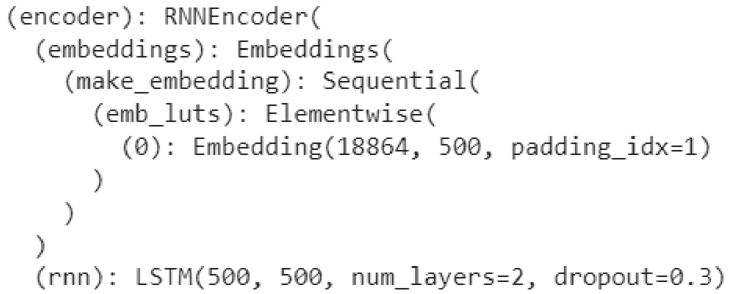
Figure 15. Train the model.

Each of the models trained for this research took around 3.5 hours to train the 100,000 steps on one GPU. For this research, this entire training procedure was done four times on the small sampled datasets (plus the fifth time with the entire Europarl corpus). Each time, the model is trained from scratch, so the different models do not build up on each other or learn from each other. Each model was trained from zero.

4.4.2 Behind the Scenes of Training an NMT Model

Section 4.4.1 describes how an NMT model is trained using the available OpenNMT-py code and what specific training parameters were used for this research. This section 4.4.2 gives a closer look at the training process. Understanding this section is not crucial to understand the results and analysis of this research. This section merely shows a few example steps of how the model is actually trained referring back to the theory described in section 1.3. The following figures demonstrate that the OpenNMT-py model trained in this research is indeed a recurrent attention model.

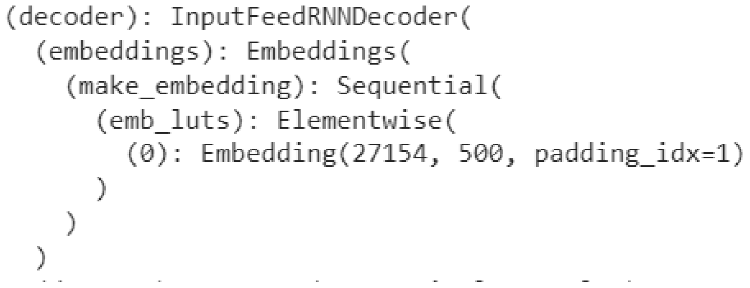
Figure 16 shows the encoder part of the model, where the input data is encoded with an RNN-encoder, sequentially going through word embeddings with LSTMs.



```
(encoder): RNNEncoder(
  (embeddings): Embeddings(
    (make_embedding): Sequential(
      (emb_luts): Elementwise(
        (0): Embedding(18864, 500, padding_idx=1)
      )
    )
  )
  (rnn): LSTM(500, 500, num_layers=2, dropout=0.3)
```

Figure 16. Encoder.

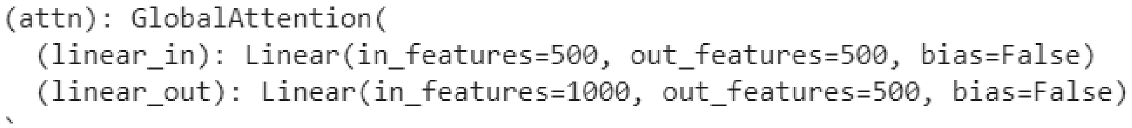
Figure 17 shows the decoder part of the model, where the information of the encoder is decoded with an RNN-decoder, also sequentially going through embeddings.



```
(decoder): InputFeedRNNDecoder(
  (embeddings): Embeddings(
    (make_embedding): Sequential(
      (emb_luts): Elementwise(
        (0): Embedding(27154, 500, padding_idx=1)
      )
    )
  )
```

Figure 17. Decoder.

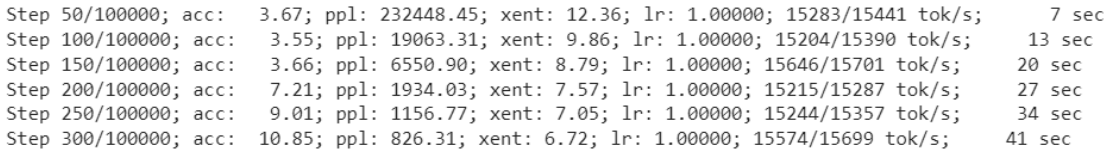
Figure 18 shows that the model uses attention, specifically global attention mechanisms. Attention is applied over the encoder at each step.



```
(attn): GlobalAttention(
  (linear_in): Linear(in_features=500, out_features=500, bias=False)
  (linear_out): Linear(in_features=1000, out_features=500, bias=False)
```

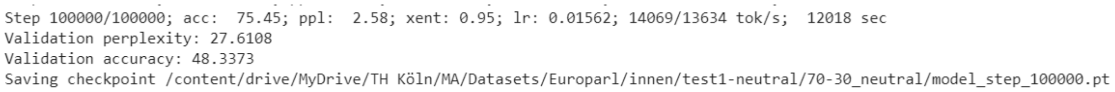
Figure 18. Attention.

The following two figures show what is printed and measured as the NMT model is being trained. In figure 19, for example the first few training steps are printed. These training steps are printed every 50 steps until the last one specified: 100,000. Once the last training step has been completed and saved, the model is done with training. Interesting specifics to note here is the “acc: 3.67”, which stands for accuracy, and the “xent: 12.36”, which stands for cross entropy, in the first line. Cross entropy „is a measure of the difference between two probability distributions for a given random variable or set of events” (Brownlee, 2020). The maximum possible accuracy is 100, whereas the cross entropy goes down to zero. As can be seen in this figure, the accuracy increases per step, whereas the cross entropy decreases per step. The higher the accuracy value and the lower the cross entropy, the better the model is trained. The numbers fall or rise very quickly during the first few steps, then slow down and drop very slowly during the latter steps.



```
Step 50/100000; acc:   3.67; ppl: 232448.45; xent: 12.36; lr: 1.00000; 15283/15441 tok/s;      7 sec
Step 100/100000; acc:   3.55; ppl: 19063.31; xent: 9.86; lr: 1.00000; 15204/15390 tok/s;     13 sec
Step 150/100000; acc:   3.66; ppl: 6550.90; xent: 8.79; lr: 1.00000; 15646/15701 tok/s;     20 sec
Step 200/100000; acc:   7.21; ppl: 1934.03; xent: 7.57; lr: 1.00000; 15215/15287 tok/s;     27 sec
Step 250/100000; acc:   9.01; ppl: 1156.77; xent: 7.05; lr: 1.00000; 15244/15357 tok/s;     34 sec
Step 300/100000; acc:  10.85; ppl: 826.31; xent: 6.72; lr: 1.00000; 15574/15699 tok/s;     41 sec
```

Figure 19. First few training steps.



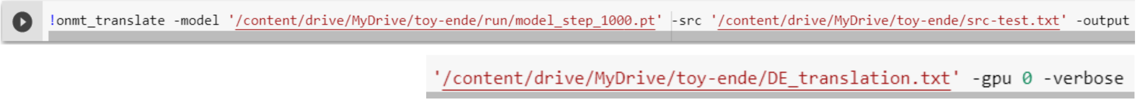
```
Step 100000/100000; acc:  75.45; ppl:  2.58; xent: 0.95; lr: 0.01562; 14069/13634 tok/s;  12018 sec
Validation perplexity: 27.6108
Validation accuracy: 48.3373
Saving checkpoint /content/drive/MyDrive/TH Köln/MA/Datasets/Europarl/innen/test1-neutral/70-30_neutral/model_step_100000.pt
```

Figure 20. Last training steps.

As can be seen in figure 20, the final accuracy for this specific model trained for this research was at 55.48 and the cross entropy at 2.16. These values appear to be quite good. A cross entropy is considered alright if it’s under 2. However, these numbers deceive because the training dataset was very small. Naturally, a model trained 100,000 steps on a small dataset will appear to have a high accuracy but if a model is trained 100,000 steps on a very large dataset, the accuracy will be lower, and the cross entropy will be much higher. For the model that was trained on the complete Europarl corpus of over 1.5 million training data, the final accuracy was 55.18 and the final cross entropy was 2.10. And 1.5 million is still a rather small dataset for training NMT models in reality, when expecting to render translations of high quality.

4.4.3 Running the Translation Model

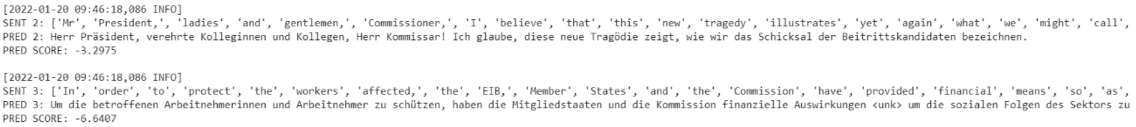
Once the model has completed training, which lasts minimally hours depending on the number of training steps and the size of the dataset, the last saved model checkpoint as mentioned in figure 20 – in this research it would be ‘model_step_100000’ for all four models – can be used to translate. Figure 21 depicts the one-line code (split over two lines here due to space limitations) to translate a text from a source language to a text of the target language.



```
!onmt_translate -model '/content/drive/MyDrive/toy-ende/run/model_step_1000.pt' -src '/content/drive/MyDrive/toy-ende/src-test.txt' -output
                                    '/content/drive/MyDrive/toy-ende/DE_translation.txt' -gpu 0 -verbose
```

Figure 21. Translating.

For the translation, three paths are needed: to the last saved checkpoint of the model, to the English text to be translated (source/input) and to the target/output to be created, in this case a German translation. This German translation will then automatically be saved to the given path, which in this scenario will also be in a specified Drive folder. Figure 22 below shows what is being outputted with the above line of a sentence showing the English source words and the below line depicting the predicted German translation output of that source sentence.



```
[2022-01-20 09:46:18,086 INFO]
SENT 2: ['Mr', 'President,', 'ladies', 'and', 'gentlemen,', 'Commissioner,', 'I', 'believe', 'that', 'this', 'new', 'tragedy', 'illustrates', 'yet', 'again', 'what', 'we', 'might', 'call',
PRED 2: Herr Präsident, verehrte Kolleginnen und Kollegen, Herr Kommissar! Ich glaube, diese neue Tragödie zeigt, wie wir das Schicksal der Beitrittskandidaten bezeichnen.
PRED SCORE: -3.2975

[2022-01-20 09:46:18,086 INFO]
SENT 3: ['In', 'order', 'to', 'protect', 'the', 'workers', 'affected,', 'the', 'EIB,', 'Member', 'States', 'and', 'the', 'Commission', 'have', 'provided', 'financial', 'means', 'so', 'as',
PRED 3: Um die betroffenen Arbeitnehmerinnen und Arbeitnehmer zu schützen, haben die Mitgliedstaaten und die Kommission finanzielle Auswirkungen <unk> um die sozialen Folgen des Sektors zu
PRED SCORE: -6.6407
```

Figure 22. Translation example

The model encompasses its own prediction score, as can be seen in figure 22, which for these two sentences are -3.2975 and -6.5407 respectively. The closer this prediction score is to zero, the better the translation quality. For a bad translation, this prediction can easily fall as low as -25.

Two examples of German translations predicted using the Model nr. 4 (*seventies*) are shown below with the OpenNMT prediction score and out of interest, the sacreBLEU score was calculated.

Example 1:

English source sentence:

“I voted to approve the Accession Treaties only under protest and out of solidarity with my future colleagues.”

German target sentence:

„Ich habe den Beitrittsverträgen unter Protest und nur aus Solidarität mit meinen zukünftigen Kolleginnen und Kollegen zugestimmt.“

OpenNMT prediction score: -0.6384

For example 1, the reference sentence from the German Europarl dataset is exactly the same as the German translation:

„Ich habe den Beitrittsverträgen unter Protest und nur aus Solidarität mit meinen zukünftigen Kolleginnen und Kollegen zugestimmt.“

sacreBLEU score: 100

Example 2:

English source sentence:

“We will not allow this European Community to be divided up, with citizens and states allotted to first or second class within it; on the contrary, we are one single common European Union, and, as such, we practise solidarity.”

German target sentence:

„Wir werden dies nicht zulassen und gemeinsam mit den Bürgerinnen und Bürgern und Staaten, die sich in erster oder jener zweiter Klasse befinden, im Gegenteil, wir gehören zu einer einheitlichen gemeinsamen Europäischen Union.“

OpenNMT prediction score: -14.2236

For example 2, the reference sentence from the German Europarl dataset is quite different:

„Wir lassen diese europäische Gemeinschaft nicht ausgrenzen, nicht in Bürgerinnen und Bürger, nicht in Staaten erster und zweiter Klasse einteilen, sondern wir sind eine gemeinsame Europäische Union und als solche solidarisch.“

sacreBLEU score: 4.10

The above two examples each show an English source sentence and the German translation of that sentence both using Model nr. 4 (*the seventies*). The OpenNMT provided prediction scores have not been analysed as part of this research, but they do provide a quick indication of the translation quality. As can be seen in the above examples, the first translation prediction scored a prediction score of -0.6384, which is a very good prediction score, whereas the bottom translation only scored a prediction of -14.2236. These prediction scores built in the translation model by OpenNMT provide a quick overview of the quality of the translation output.

For sacreBLEU scores, the maximum and best quality that a translation can achieve is 100 (like 100%), and the lowest being 0. The sacreBLEU score for the first example naturally is 100 since the hypothesis and reference sentences are exactly the same. For example 2, however, the sacreBLEU score lies only at 4.10. These sacreBLEU scores underline the OpenNMT prediction scores.

5 Analysis

This section presents results and analysis of the research conducted for this dissertation. The first outputs of this research are the German translations of the English test set for each of the models trained using OpenNMT-py. Next to the German translation obtained from the model that was trained on the complete Europarl dataset (of 1,527,168 sentences in the train data and 189,206 sentences in the validation data), four German translations were obtained of the same English source text (test set) from each of the four smaller models (*the hunnies, the nineties, the eighties,* and *the seventies*), with 78,083 sentences in the train data and 9,370 sentences in the validation data.

The German translation obtained from the model that was trained on the complete Europarl dataset serves as a reference point, not to directly compare, but to show what automatic evaluation scores can be evaluated by training a basic NMT model on the Europarl corpus, and how much gender (female plural terms) is present in that output. The direct comparisons will be done between the four equally sized smaller model outputs.

This section covers two aspects of analysis: gender bias and automatic evaluation metrics. Section 5.1 covers an analysis of gender-bias in the German translation outputs and how the different types of models, depending on what type of data they've been trained with, lead to differently gender-biased translation outputs. Section 5.2 covers an analysis of automatic evaluation metrics of the German translation outputs in comparison to the German reference test text for each of the four models. The automatic evaluation metrics calculated for this section are sacreBLEU, COMET and the recall, precision and F-measure BERTscores.

Important to note here is that the data used to train and validate the different models was not manipulated: individual female plural terms ending in 'innen' were not simply filtered out of a sentence to leave only the male plural term. However, and this is very important, the resulting compiled datasets for training each model may vary in context. The smaller datasets all stem from the overall English-German Europarl corpus but different lines (different sentences) were randomly selected to comprise the four smaller datasets. That may mean that very different content and therefore very different vocabulary was used to train these four models. The overall language is similar for all datasets since they stem from the same corpus, it is formal language from the European Parliament. Yet, the individual sentences compiled to form the smaller datasets, may contain very different information – with topics ranging from policies to assassinations to athletes. That entails that some of these four datasets contain more information about a certain topic than others, and vice versa.

Particularly when looking at Models nr 3. (*the eighties*) and nr. 4 (*the seventies*) that respectively contain 80% and 70% of sentences not including and 20% and 30% including female plural terms, the diversity of topics may be very different to Model nr. 1 (*the hunnies*) which does not include any female plural terms. For example, Models nr. 3 and nr. 4 may cover more social content such as topics of female athletes taking anabolic substances, whereas Model nr. 1 may cover more neutral content such as budgets. This is just an example but something to bear in mind particularly when looking at automatic evaluation scores of how well the German translations score in comparison to the reference text.

5.1 Gender-Bias in Translation

This section covers the analysis of gender-bias in the four German translation outputs and is the most important part of analysis for this dissertation since it measures gender-bias in translation outputs depending on different training data. All four German translations stem from the same English source text. The aim was to show that training an NMT model on much smaller but more gender-balanced datasets reduces gender-bias in the output translation (although unfortunately at the expense of translation quality). The idea followed was that of Saunders and Byrne (2020) that a "well-formed small dataset may give better results than attempts at debiasing the entire original dataset" (7728).

One translation was yielded by Model nr. 1 (*the hunnies*), which was trained on data **not containing** any female plural term ending in 'innen' at all. The second translation was yielded by Model nr. 2 (*the nineties*), which was trained on 90% data **not containing** female plural terms ending in 'innen' and 10% of data **containing** female plural terms ending in 'innen'. The third translation was yielded by Model nr. 3 (*the eighties*), which was trained on 80% data **not containing** female plural terms ending in 'innen' and 20% of data **containing** female plural terms ending in 'innen'. The fourth translation was yielded by Model nr. 4 (*the seventies*), which was trained on 70% data **not containing** female plural terms ending in 'innen' and 30% of data **containing** female plural terms ending in 'innen'. Important to remember here is that those sentences not containing female plural terms may often be gender-neutral sentences such as: "It is also an important country economically".

To measure gender and with that determine whether a text might reflect more or less gender-bias, the number of sentences containing female plural words ending in 'innen' was counted in each German translation. This was done by filtering the same words using a regex function as was outlined in sections 4.3.3 and 4.3.4 of preparing the data. A manual count was then done to confirm or deny this computational count leading to an error rate of maximum 10. The number and percentage of sentences containing female plural words such as "Bürgerinnen" or "Europäerinnen" for each German translation for each model tested are listed in table 6 below. All German translations stem from the same English source text.

	Model trained on complete Europarl EN-DE corpus	**Model nr. 1** *the hunnies*	**Model nr. 2** *the nineties*	**Model nr. 3** *the eighties*	**Model nr. 4** *the seventies*
No. of sentences containing female	99	0	143	199	235

plural terms in DE translations					
Percentage of sentences of the DE translations containing female plural terms	1.06%	0	1.53%	2.12%	2.51%

Table 6. Number and percentage of sentences containing female plural terms ending in ‘innen’ in German translation of the same English source text

The result of the model trained on the complete English-German Europarl corpus is merely displayed as an example, the results to be directly compared are those from Models 1 to 4. The total number of female plural terms counted (computationally and manually) in the German translation using the model trained on the complete (large) dataset is 99. The German translations of the English test set each contain 9,370 sentences. This means that 99 out of 9,370 sentences, so 1.06%, contain a female plural term. This number is very low because, as already explained, the dataset is not very gender-laden in the first place, with the language being rather formal, focussed on topics surrounding the European Parliament such as resolutions, member states and budgets. Nevertheless, numerous sentences in this German translation could have been gendered but the male plural term was chosen such as in the following sentence, where ‘citizens’ was translated to ‘Bürger’ instead of ‘Bürgerinnen und Bürger’ for example: “Es gibt eine große Gruppe von **Bürgern**, die in der Diskussion über die Zusammensetzung des Europäischen Parlaments ignoriert wurden.“

The German translation resulting from the model trained on the complete EN-DE dataset will not be considered for the further gender-analysis. The four smaller models, since they were trained on equally sized datasets, will be analysed more closely. Depicted below in figure 23 is a bar chart of the number of sentences containing female plural terms ending in ‘innen’ in the four German translations – the same values as shown in table 6 above but depicted more visually. This bar chart also includes error bars of value 10. Figure 23 clearly shows that the number of sentences including female plural terms increases from Model nr. 1 to Model nr. 4 as more female-biased datasets are used to train the relevant model.

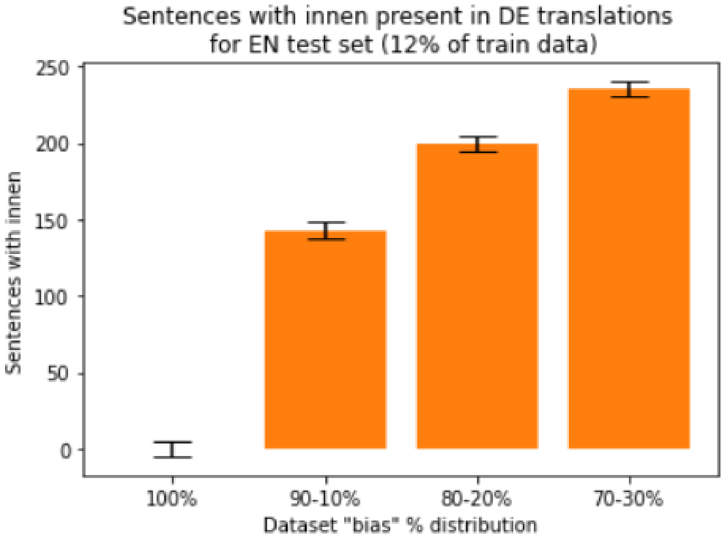


Figure 23. Bar chart of the number of sentences (with error bars) including female plural terms (words that end in ‘innen’) present in the German translations of the English test set, for all four models.

On the left, the model was trained with zero female plural terms in the training data and not surprisingly, the German translation resulting from that model also contains zero female plural terms. Moving on to Model nr. 2, 10% of the training data were sentences containing at least one female plural term. In the German translation resulting from this model, a total of 143 sentences can be counted containing a female plural term. That is 1.53% of sentences of the total German translation. There is a clear spike and the biggest difference from Model nr. 1 to Model nr. 2 regarding their German translations containing female plural terms – stemming from the same English translation.

The two bars on the right show the results for Models nr. 3 and nr. 4. The German translation obtained from Model nr. 3, whose training data included 20% of sentences containing female plural terms, counts a total of 199 sentences with female plural terms, which is 2.12% of sentences of the total German translation. This is an increase of 56 sentences in comparison to Model nr. 2. Last but not least, Model nr. 4, whose training data included 30% of sentences containing female plural terms, counts a total of 235 sentences with female plural terms, which is 2.51% of sentences of the total German translation.

This result beautifully shows that if the training data used to train and validate an NMT model is less ‘male-biased’ and contains more female terms such as female plural words ending in ‘innen’, then translations done using such a NMT model will respectively show more female presence by yielding a higher number of female plural terms.

Figure 24 below depicts the same as figure 23 above but includes a blue line that marks the number of sentences with female plural terms ending in ‘innen’ present in the German reference text. This bar chart shows how each German translation obtained by each of the four models ranks in comparison to the German reference text when looking at the number of female plural terms present in the text.

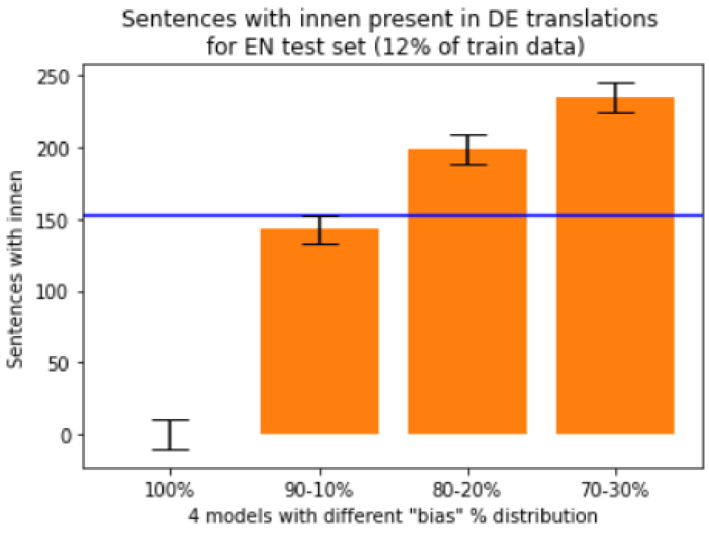


Figure 24. Bar chart of the number of sentences (with error bars) including female plural terms (words that end in ‘innen’) present in the German translations of the English test set, for all four models. The blue line is for reference: that is how many female plural terms (words that end in ‘innen’) are present in the German reference test text: 152.

This shows that the German reference, taken from the Europarl corpus, is also slightly male-biased and does not translate all gender ambiguous terms such as ‘Europeans’ into the male and female plural, as can be seen in the following example: „Mit der irischen Ablehnung wurde wieder einmal der gemeinsame Wille der **Europäer** auf die Probe gestellt, die Herausforderungen der Globalisierung Hand in Hand zu bewältigen”, where ‘Europeans’ was translated to ‘Europäer’ instead of ‘Europäerinnen und Europäer’. This German Europarl dataset contains a combination of ‘Kollegen’, ‘KollegInnen’, ‘Kollegen und Kolleginnen’ and ‘Kolleg/innen‘, showing that it was authored or translated by different individuals. This makes it rather difficult to calculate adequate automatic evaluation metrics, as explained in section 5.2.

The below tables 7 and 8 further show how each of the four models translated certain English source sentences into German. Looking at table 7, the English source sentence “I believe that this is an important message for the voters and the citizens of Europe” was translated differently by each model. The focus here lies on the English gender-ambiguous plural terms ‘voters’ and ‘citizens’. The first row shows the German reference sentence, which, in this case, includes both “Wählerinnen und Wähler” as well as “Bürgerinnen und Bürger”.

As can be expected from the results outlined above, Model nr. 1’s translation only includes the male plural term ‘Wähler’ and ‘Bürger’. The translation done by Model nr. 2 shows the same gender-result. Models nr. 3 and nr. 4, however, which had a higher percentage of training data containing female plural terms both yielded translations of ‘Wählerinnen und Wähler’. Unfortunately, the female-centred percentage in the training data of these two models was not yet enough to also translate ‘Bürgerinnen und Bürger’ in this specific sentence.

Model	Englisch source sentence	German target translation
German reference sentence	*“I believe that this is an important message for the **voters** and the **citizens** of Europe.”*	*„Ich glaube, dass das eine wichtige Botschaft ist, auch für die **Wählerinnen und Wähler**, für die **Bürgerinnen und Bürger**.“*
Model nr. 1 (*the hunnies*)		*„Meiner Ansicht nach ist dies eine wichtige Botschaft für die **Wähler** und die europäischen **Bürger**.“* sacreBLEU: 8.58
Model nr. 2 (*the nineties*)		*„Ich glaube, dass dies eine wichtige Botschaft für die **Wähler** und die **Bürger** Europas ist.“* sacreBLEU: 17.79
Model nr. 3 (*the eighties*)		*„Ich glaube, dass dies eine wichtige Botschaft für die **Wählerinnen und Wähler** und die **Bürger** Europas ist.“* sacreBLEU: 32.83
Model nr. 4 (*the seventies*)		*„Ich glaube, dies ist eine wichtige Botschaft für die **Wählerinnen und Wähler** und für die europäischen **Bürger**.“* sacreBLEU: 28.34

Table 7. An example of an English source sentence translated into a German sentence by the four different models. The German reference sentence and a sacreBLEU score for each reference are also included.

Out of interest, the sacreBLEU for each translation in comparison to the German reference was calculated. The translation from Model nr. 1 shows the lowest score of 8.58 (remembering that the best sacreBLEU score is 100), and the translations for Models nr. 2, 3 and 4 respectively are 17.79, 32,83 and 28.34. The translations by Models nr. 3 and nr. 4 clearly score the highest sacreBLEU, both of which contain the female plural term for ‘voters’. The score would have been even higher if they had also contained the female plural term of ‘citizens’.

Table 8 equally shows an example of how each of the four models translated certain English source sentences into German. The English source sentence “on behalf of the UEN Group. - (PL) Mr President, the development of the ‘Union's’ internal market is very significant in terms of forging links between the countries and the citizens of the Union.” was translated differently by each model. The focus here lies on the English ambiguous plural term ‘citizens’. All four German translations include an <unk> meaning an ‘unknown’ term that the model did not have in its vocabulary and so could not translate from English into German. Considering that these models were each trained on a very small datasets of less than 100,000 sentences (with MT models usually being trained on many millions of sentences), it is not surprising that the training vocabulary does not cover all vocabulary present in a test dataset. The more data a model can train on, the more vocabulary it will have available and the better the resulting machine translations. Naturally, such unknowns lower the overall translation qualities and therefore also automatic evaluation metrics, as shown in section 5.2 but what is important for this dissertation – how much gender is present in the final translation – can still be seen.

In table 8 for example, the first three models all translated ‘citizens’ to ‘Bürger’ and only Model nr. 4 that was trained with a dataset containing most female plural terms translated it to ‘Bürgerinnen und Bürger’. This increase in female presence was exactly the aim of training a model on a dataset that contained more female-gender terms in comparison to models trained on datasets that contained less female-gendered terms, percentage-wise even the complete Europarl parallel corpus.

Model	**Englisch source sentence**	**German target translation**
Model nr. 1 (*the hunnies*)	*“on behalf of the UEN Group. - (PL) Mr President, the development of the "Union's" internal market is very significant in terms of forging links between the countries and the* ***citizens*** *of the Union.”*	*„im Namen der UEN-Fraktion. - (PL) Herr Präsident! Die Entwicklung des <unk> der Gemeinschaft ist für uns sehr <unk> was die Beziehungen zwischen den Ländern und* ***den Bürgern*** *der Union betrifft.“*
Model nr. 2 (*the nineties*)		*„im Namen der UEN-Fraktion. - (PL) Herr Präsident! Die Entwicklung des Binnenmarktes der Union ist bei <unk> Verbindungen zwischen den Ländern und* ***den Bürgern*** *der Union sehr wichtig.“*
Model nr. 3 (*the eighties*)		*„im Namen der UEN-Fraktion. - (PL) Herr Präsident! Die Entwicklung des Binnenmarktes der Union ist im Hinblick auf <unk> Beziehungen zwischen den Ländern und* ***den Bürgern*** *der Union sehr wichtig.“*
Model nr. 4 (*the seventies*)		*„im Namen der UEN-Fraktion. - (PL) Herr Präsident! Die Entwicklung des Binnenmarktes ist im Hinblick auf die <unk> von Verbindungen zwischen den Ländern und* ***Bürgerinnen und Bürgern*** *der Union sehr wichtig.“*

Table 8. An example of an English source sentence translated into a German sentence by the four different models.

The take-away from this is that it can be measured and depicted that if training data is ‘male-biased’, the translations will also be ‘male-biased’. Whereas for every little bit that the training data is less biased but skewed towards being more gender-balanced, the translations will also reflect less bias, showing more female-gender. Measuring gender present in the output translations of each of the four models is the crucial part for this dissertation. This analysis shows that gender-bias in training data is clearly reflected in machine translations. Nevertheless, automatic evaluation metrics are the go-to metrics to assess the quality of a machine translation. A select number of automatic evaluation metrics were calculated as part of this research and the results are outlined in section 5.2 below.

5.2 Automatic Evaluation (string-based and embedding-based) Metrics

This section compares and evaluates the following automatic evaluation metrics: sacreBLEU, COMET and BERTScores (Recall, Precision, F-Measure). As explained in section 1.6, sacreBLEU is an adapted automatic evaluation metric of the string-matching BLEU metric. COMET and BERTScores are embedding-based metrics. There are various automatic evaluation metrics and even more variations and adaptions of these were developed over time. These three main metrics were chosen because they have been developed or adapted in the last few years and are under the top metrics to measure machine translation (semantic) quality.

In the realm of this research, these metrics are less relevant as the analysis in section 5.1 as the focus of this dissertation lies specifically on the presence of gender (bias) in machine translations. One thing that needs to be said before discussing the metric results is that the overall average of these metrics is relatively low due to the very small datasets (of 78,083 sentences) that the four models were trained on. The small size of the datasets directly impacts the amount of vocabulary a model has available for training, as outlined above. This means that there may be quite a lot of vocabulary in the English test set that the model had not yet been trained for, leading to a high number of 'unknowns' present in the final translations for each of the four smaller models. This could be counteracted, at least partially, by training the model on pre-tokenized datasets, by using SentencePiece for example, but this was not done in this research as outlined in section 6 on limitations.

5.2.1 sacreBLEU

As done by Saunders and Byrne (2020: 7730), sacreBLEU scores were calculated for the output translations of this research. The scores are both shown in table 9 below and plotted on a graph in figure 25. All metrics were also calculated for the model trained on the complete English-German Europarl corpus as a sort of baseline, to show what a model trained on this data is capable of achieving. Since the size of the datasets these models were trained on, however, vary massively, only the metric scores for the four smaller models are plotted and analysed in detail.

	Model trained on complete Europarl EN-DE corpus	**Model nr. 1** *the hunnies*	**Model nr. 2** *the nineties*	**Model nr. 3** *the eighties*	**Model nr. 4** *the seventies*
sacreBLEU	19.4819	13.1202	13.0504	12.4791	12.4583

Table 9. sacreBLEU scores for the German translations compared to the German reference text of all four smaller models and the model trained on the complete Europarl corpus.

These sacreBLEU scores were automatically calculated by comparing the German translations of the same English test set with the German reference text (directly taken from the German Europarl corpus). Table 9 shows that the sacreBLEU score reached by the model trained on the complete parallel corpus is 19.4819. sacreBLEU scores from 20 above can be considered quite acceptable so this value scratches the surface. Perfect sacreBLEU scores lie at 100 but when looking at BLEU scores (note: BLEU not sacreBLEU, even though these metrics can be roughly compared) in current research, values often lie around the 40s, not in the 80s or 90s.

When having a closer look at the sacreBLEU scores for Models nr.1 to 4, it is interesting to see that the value of the scores drops noticeably. It can be clearly seen that these scores are well below the score reached by the model trained on the complete parallel corpus. The sacreBLEU scores drop by at least 6.3 points. This is understandable considering that the datasets the four smaller models were trained on were only 5% the size of the dataset the big model was trained on. Being aware of that, it is actually remarkable that the metric scores are not even lower.

For better visualisation, the sacreBLEU scores of the four smaller models are plotted in figure 25.

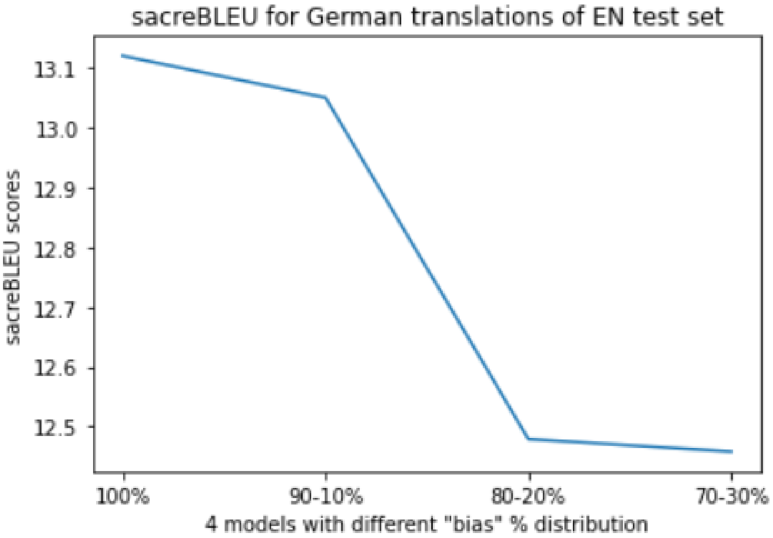


Figure 25. sacreBLEU scores for the German translation (of the English test set) of each of the four models in comparison with the German reference test text. Note: the y-axis does not start at 0 for better visualisation. The range between the minimum and maximum y-value (sacreBLEU score) is 0.6619.

The translation of Model nr. 1 yields a sacreBLEU score of 13.1202 with this score being slightly lower for Model nr. 2 (13.0504) and noticeably lower for Models nr. 3 (12.4791) and nr. 4 (12.4583). The reason for the difference of these scores are manifold. First and foremost, the German reference text, against which these translations are measured, does not contain a very high number of gendered-terms (female plural terms ending in ‘innen’) itself. In the German reference text of 9,370 sentences, only 1.62% of those sentences contain a female-gendered (‘innen’) term. This means that a higher number of female-gendered terms in a translation will not necessarily yield a better translation evaluation or a higher metric score when compared to a reference text, which does not contain a relatively high number of female-gendered terms either. Therefore, the automatic evaluation metrics analysed in this section do not refer to the topic of gender-bias in machine translation based on gender-bias in training data.

What can be seen here is that the training data might also be skewed by content. As mentioned in sections 4.3.2, 4.3.3 and 4.3.4 on data preparation, skewing the datasets to include more sentences with female gender (female plural terms) might also skew the content and vocabulary a model trained on that dataset will learn. Since the Europarl corpus is mainly formal speech and its content covers information discussed at the European Parliament, those datasets that contain more of this information will likely yield a higher metric score. The German reference text itself includes more formal information on European Parliament content rather than gender-focussed discussions. Therefore, Models nr. 1 and nr. 2 actually score higher sacreBLEU scores than Models nr. 3 and nr. 4 whose data was more skewed to include female-gender.

5.2.2 BERTScores

As described in section 1.6.6, BERTScore calculates a similarity score using BERT pre-trained contextual embeddings, based on word embeddings, for every token in a MT sentence compared with every token in the human reference sentence (Zhang et al., 2020: 4). Contextual embeddings can generate different vector representations for one word in different sentences depending on the surrounding words, the surrounding context (Zhang et al., 2020: 3). BERTScore calculates the traditional metrics recall (R_{BERT}), precision (P_{BERT}), and F-Measure (F_{BERT}) scores, where the latter is a combination of the first two. These three metrics are evaluated in this section. BERTScore is an embedding-based metric, which more effectively detects paraphrasing and distant dependencies and ordering, compared to string-based metrics that match *n*-grams from the surface of a sentence (Zhang et al., 2020: 1, 9). BERTScores are calculated between 0 and 1, with 1 being the highest achievable score, referring to a high-quality MT.

Table 10 below depicts the three calculated BERTScores R_{BERT}, P_{BERT} and F_{BERT} for each model. For each model and each BERTScore the average scores for the German translations are calculated, with all three scores ranging between the maximum of 1 and the minimum of 0. On average, however, the scores below have been calculated for the German translation of each model. All three scores, and specifically F_{BERT}, being a combination of R_{BERT} and P_{BERT}, are highest for the model trained on the complete parallel corpus. Since this model was trained on 95% more data, it only makes sense that the resulting translation scores a much higher value, as already seen when looking at sacreBLEU above. sacreBLEU and BERTScores do not specifically focus on gendererd terms and as the entire Europarl dataset in general does not contain much gender, it is only natural for the big model to score much higher in automatic evaluation metrics.

	Model trained on complete Europarl EN-DE corpus	**Model nr. 1** *the hunnies*	**Model nr. 2** *the nineties*	**Model nr. 3** *the eighties*	**Model nr. 4** *the seventies*
R_{BERT}	0.7907	0.7536	0.7512	0.7447	0.7461
P_{BERT}	0.7778	0.7385	0.7341	0.7263	0.7280
F_{BERT}	0.7839	0.7456	0.7422	0.7350	0.7366

Table 10. The average values of R_{BERT} (Recall) and P_{BERT} (Precision) and F_{BERT} (F-Measure) scores for the German translations compared to the German reference text and the English source text of all four smaller models, and the model trained on the complete Europarl corpus.

What is most interesting that for the big dataset, the best BERTscores lie at 1 and the lowest merely at 0.4490, whereas for the four smaller models, the best BERTScores all lie at 1 and the lowest all lie at 0, meaning that those translations scoring 0 are completely off or simply filled with ‘unknowns’. The four smaller models are the ones that can be be compared. Interestingly, these do not vary as strongly with the biggest difference being 0.0106. Curiously, Model nr. 3 is an outlier by breaking the linear trend of decreasing values from Model nr. 1 to Model nr. 4. This can also be clearly seen for the COMET scores evaluated in section 5.2.3.

Figure 26 below depicts the F_{BERT} (the combination of R_{BERT} and P_{BERT}) average scores for the four smaller models for better visualisation.

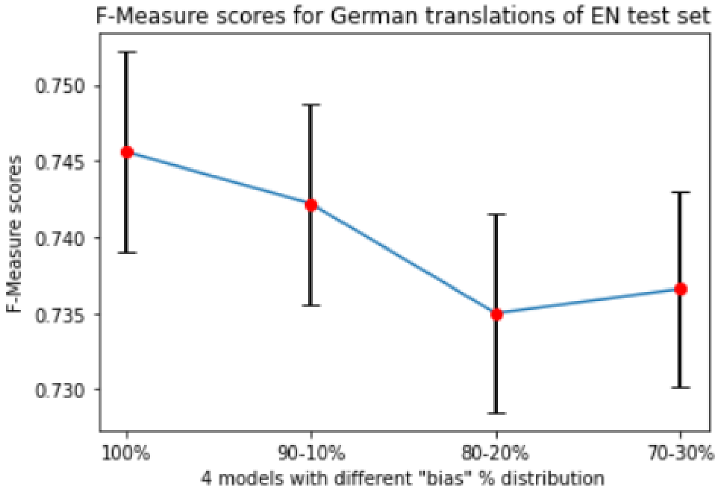


Figure 26. The average values F_{BERT} (F-Measure) scores with error bars for the German translation (of the English test set) of each of the four models in comparison with the German reference test text. Note: the y-axis does not start at 0 for better visualisation. The range between the minimum and maximum y-value (BERTScores) is around 0.0489.

Next to the plot of the average value, the variance is plotted as error bars (in black). The standard deviation is quite large with the actual values ranging between 0 and 1. When looking at the overall variance, however, this lies at a maximum of 0.0075. Apart from Model nr. 3 being a slight outlier to the linear decrease, there is a clear trend of the BERTScores decreasing from Model nr. 1, which does not contain any female-gendered plural terms, to Model nr. 4, which does include 30% of sentences with female-gendered plural terms. This is similar to the sacreBLEU scores above. BERTScores are calculated differently to sacreBLEU scores but in this case yield the same pattern of results for the four models.

The reasons for the BERTScores to drop from Model nr. 1 to nr. 4 might be the same as mentioned above: a higher number of female-gendered terms in a translation will not necessarily yield a higher metric score (if the German reference does not contain a high number of female-gendered terms itself), and the datasets that the four models are each trained on might contain very different context and vocabulary that are either more or less reflected in the German reference. For both the sacreBLEU scores and BERTScores, it can be said that the automatic evaluation metrics analysed in this section do not reflect the presence of gender in machine translation based in 'gender-bias' in training data.

5.2.3 COMET

The third calculated automatic evaluation metric is COMET. In comparison to the traditional BLEU score, COMET is a rather new metric that has been finetuned to assess the quality of machine translation. COMET is also an embedding-based metric. Embedding-based metrics create soft-alignments between a human reference sentence and a MT (hypothesis) sentence in an embedding space and then compute a score reflecting the semantic similarity between said reference and hypothesis segments (Rei et al., 2020: 2692).

COMET scores are calculated by using available models to evaluate translations. The model used for calculating COMET for this research was 'wmt-comet-da' (Rei et al., 2020). COMET scores are calculated as values between 0 and 1 with one being the highest value and 0 being the lowest. On average, a score between 0.4 and 0.5 is considered quite a good measure of a qualitative translation. Naturally, the higher the score, the higher the translation quality.

Once again, the COMET score was also calculated for the complete parallel corpus as a 'baseline' to show what score a model trained on the entire dataset can yield. The result is a score of 0.4613 as shown in table 11 below. Considering this score lies between 0.4 and 0.5, it can once again be said, that the 'baseline' model produces an acceptable translation. Again, however, only the four smaller models can be directly compared.

	Model trained on complete Europarl EN-DE corpus	**Model nr. 1** *the hunnies*	**Model nr. 2** *the nineties*	**Model nr. 3** *the eighties*	**Model nr. 4** *the seventies*
COMET	0.4613	0.4700	0.4878	0.4511	0.5173

Table 11. COMET scores for the German translations compared to the German reference text and the English source text of all four smaller models, and the model trained on the complete Europarl corpus.

Interestingly, these COMET scores yield the opposite results of what sacreBLEU and the BERTscores presented, with Model nr. 4 reaching the highest score. Once again it can be seen that Model nr. 3 is an outlier of the linearly increasing trend for the COMET scores as well as for the BERTScores discussed in section 5.2.2. This becomes more visible when looking at the graph in figure 27 below.

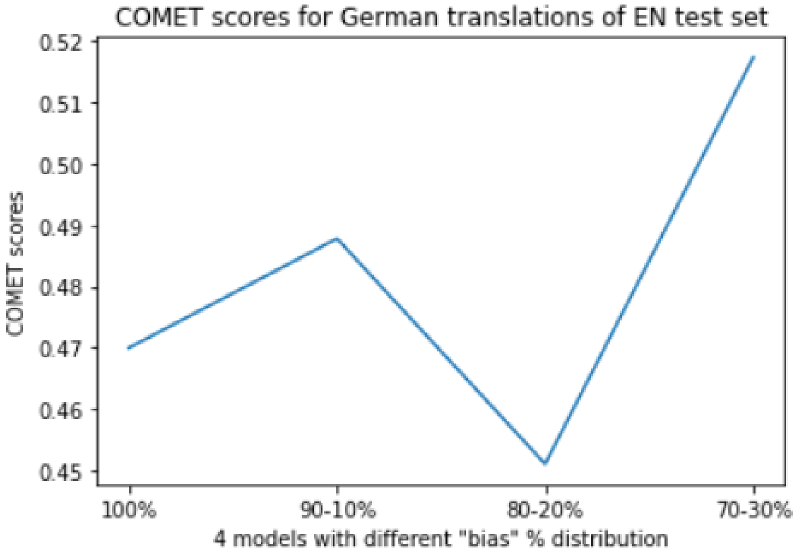


Figure 27. COMET scores for the German translation (of the English test set) of each of the four models in comparison with the German reference test text. Note: the y-axis does not start at 0 for better visualisation. The range between the minimum and maximum y-value (COMET score) is 0.0662.

COMET is said to capture semantic similarity and is therefore closer to human judgement. As described in section 1.6.5 in experiments conducted by Rei et al. (2020) COMET scores outperform string-based metrics such as BLEU but also the embedding-based metric BERTScore. The COMET models tested achieve advanced results for segment-level correlation with human judgements (Rei et al., 2020: 2685). This is very interesting for this research because the COMET scores are opposite to the sacreBLEU and BERTScores where Model nr. 1 (*the hunnies*) reached the highest scores, with Model nr. 4 (*the seventies*) actually reaching the highest COMET score. Theoretically it means that Model nr. 4 reaches the highest semantic similarity and MT quality. Practically, however, it is difficult to fully evaluate the models and their respective translations with metrics considering that the datasets used to train them were very small and a number of 'unknowns' are present in the final translation, which interfere with the calculation of metrics.

Nevertheless, it is interesting to evaluate the models using automatic evaluation metrics, which are the go-to for any research evaluation MT, and to have a look at the outcomes.

5.3 What if the German Reference Contained More Female Plural Terms?

The above analysis was done without purposefully manipulating data and the English test set that each model was tested on by creating a German translation was a small percentage randomly taken from the large Europarl corpus, as per MT practice. As explained above, the Europarl corpus in its nature is very formal and not very gendered, resulting in the English test-set also not containing a lot of possibility for gender. What if, however, the English test-set contained many more gender-ambiguous words that could be translated into German either by opting for the plural male generic or by including the plural female term?

This has been done as an example this section, which goes against the usual MT practice. The overarching aim of this dissertation is to show that when a NMT system is trained on gender-biased data, the output translations will also be gender-biased. With limitations, this was evaluated and shown in section 5.1. The approach taken in this section is slightly different. Each of the four models was trained with the same data and same percentage of gender-bias in the data as done above (so the four models were the same ones as for the previous analysis). However, the translation analysed in this section results from a different English test set.

For this separate evaluation, an English test set of 434 sentences was compiled. All 434 sentences contain at least one gender-ambiguous word in English such as 'colleagues', 'consumers' or 'rapporteur' that could be translated differently into German, either just stating the plural male generic or including a type of female-gendered version like 'VerbraucherInnen' or 'Verbraucherinnen und Verbraucher'. The 434 sentences were taken directly from the English Europarl corpus without any further adjustment. This was done by filtering the English corpus filtered for sentences containing such gender-ambiguous words. The German reference (Europarl corpus) of these 434 has been translated or authored so that each of these 434 sentences include a female plural term. Since the aim in MT is to create such gendered translations to help create a more gender-fair society, this German reference is considered (in the realm of this dissertation and specifically this section 5.3) a 'golden standard' because that is what we ultimately want to achieve.

The English test-set and the German reference taken from the Europarl parallel corpus thus contain sentences such as the following:

English test-set sentence:

”We have always given strong support to animal protection and **consumer** protection in this House.”

German reference sentence:

„Wir haben uns in diesem Haus immer sehr für den Tierschutz und den Schutz der **Verbraucherinnen und Verbraucher** ausgesprochen.“

This research aims to show that if data used to train an NMT system contains more or contains less female-gendered terms, this will be directly reflected in the output translation. To focus more on gender, a small sample of the Europarl corpus was taken as a test sample. All four models created a German translation of this gender-ambiguous English test set and the results are quite remarkable and very different to the ones above.

These results are not discussed in detail since the test-set is meant to reflect the overall Europarl corpus and since the gender-ambiguous test-set does not reflect the overall Europarl corpus, the results do not reflect MT practice. The results do, however, show that gender-bias in data used to train an NMT model is clearly reflected in the translation outputs.

Just to show what a difference it makes if the English test-set contains gender-ambiguous terms that can be translated differently, the results of the four translations of the four models are presented here. As in section 5.1 above, the number of sentences in the German translation containing a plural female-gendered term ending in ‘innen’ was counted for each model translation. The results are shown in figures 28 and 29 below. Figure 28 is a bar chart depicting the number of sentences containing a female-gendered plural term (ending in ‘innen) in the German translations of each model. In the German reference, there is a female-gendered term in every sentence, so 434 in total. This bar chart clearly shows that, once again, if there are more female-gendered terms are included in the training data, there are also more female-gendered terms included in the German translation. If the training data is male-biased, then the output translation will also be male-biased. For each percentage increase of more female-gendered data in the training data, the number of female-gendered terms in the translations increases as well.

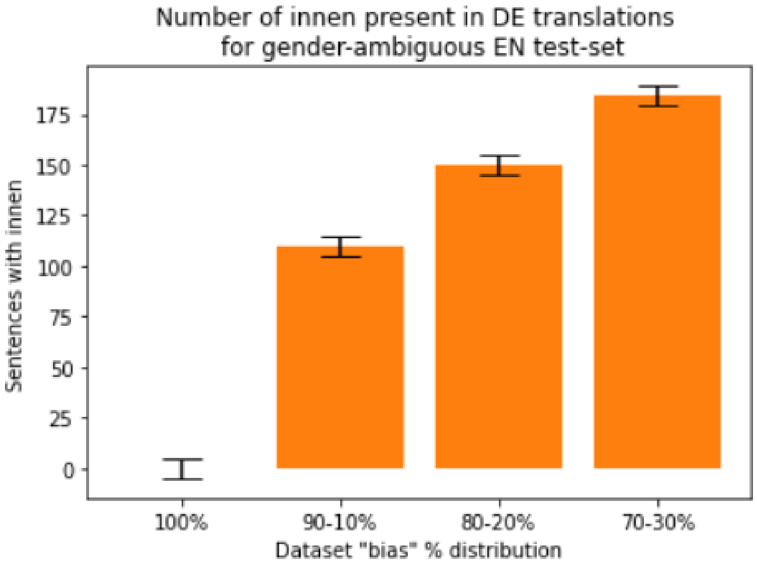


Figure 28. Bar chart of the number of sentences (with error bars) including female plural terms (words that end in ‘innen’) present in the German translations of the English test set, for all four models.

Figure 29 below shows the same value of sentences including female-gendered terms in the German translations but as a percentage. As previously, Model nr. 1 does not yield any female-gendered terms in the translation as it was trained on zero female-gendered plural terms. The German translation from Model nr. 2 contains 110 female-gendered plural terms (ending in ‘innen’), which is 25.35%. The German translation from Model nr. 3 contains 150 female-gendered plural terms (ending in ‘innen’), which is 34.56%. The German translation from Model nr. 4 contains 184 female-gendered plural terms (ending in ‘innen’), which is 42.4%.

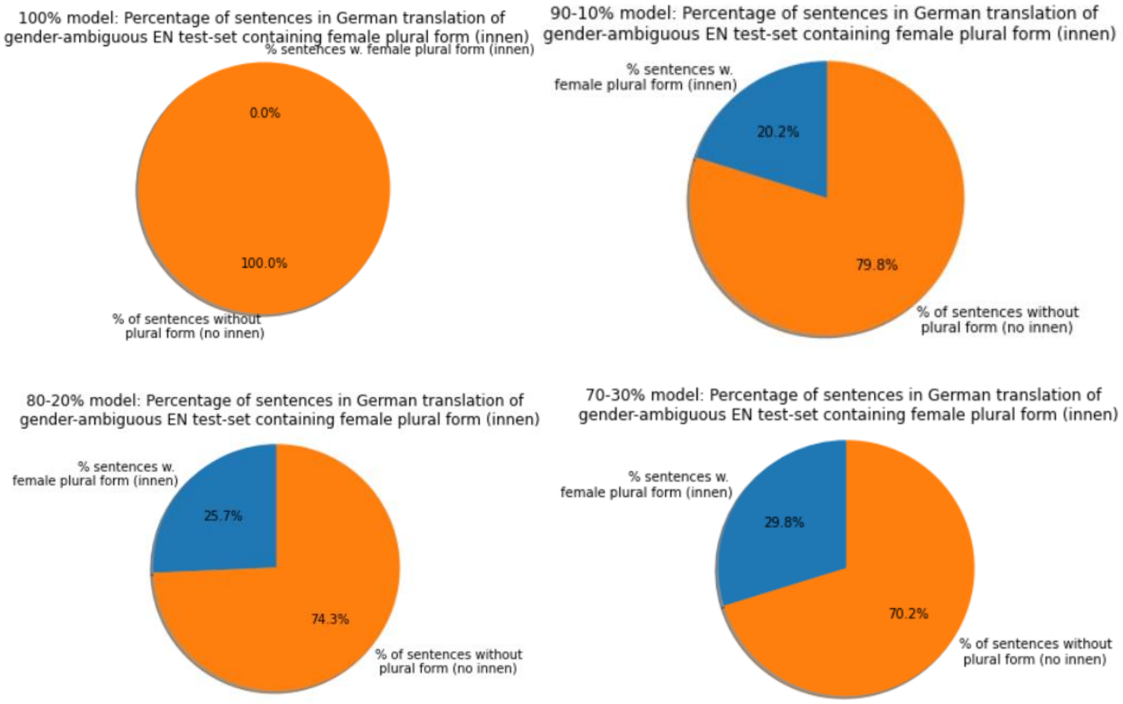


Figure 29. Pie charts of the number of sentences including female plural terms (words that end in 'innen') present in the German translations of the English test set, for all four models.

The results visualised in figure 29 once again clearly show that as the number of female-gendered terms in the training data increases, the number of female-gendered terms in the translation also increases, leading to a more gender-fair translation. Therefore, it can clearly be seen that if a NMT system is trained on a gender-biased dataset, the output translations will also be gender-biased. For each measure taken to decrease such gender-bias, the gender-bias will also decrease in the output translations, leading to a gender-fairer translation.

Just to demonstrate the different results achieved in this section, the sacreBLEU scores were calculated for each of the four models. They are quite different to the ones calculated in section 5.2.1. Translations from the gender-ambiguous English test set yield very different results, with Model nr. 1 scoring the lowest sacreBLEU score of 15.360 and Model nr. 4 scoring the highest of 19.4142, as shown in table 12 and visualised in figure 30 below.

	Model nr. 1 *the hunnies*	**Model nr. 2** *the nineties*	**Model nr. 3** *the eighties*	**Model nr. 4** *the seventies*
sacreBLEU	15.360	17.4467	18.8381	19.4142

Table 12. sacreBLEU scores for the German translations of the small ambiguous English test set-compared to the German reference text of all four smaller models.

The sacreBLEU scores are now in favour of the more female-gendered models in comparison to the more male-biased models since the German reference text (against which the translations are compared) is also more female-gendered, with each single sentence containing at least one female-gendered plural term (ending in ‘innen’).

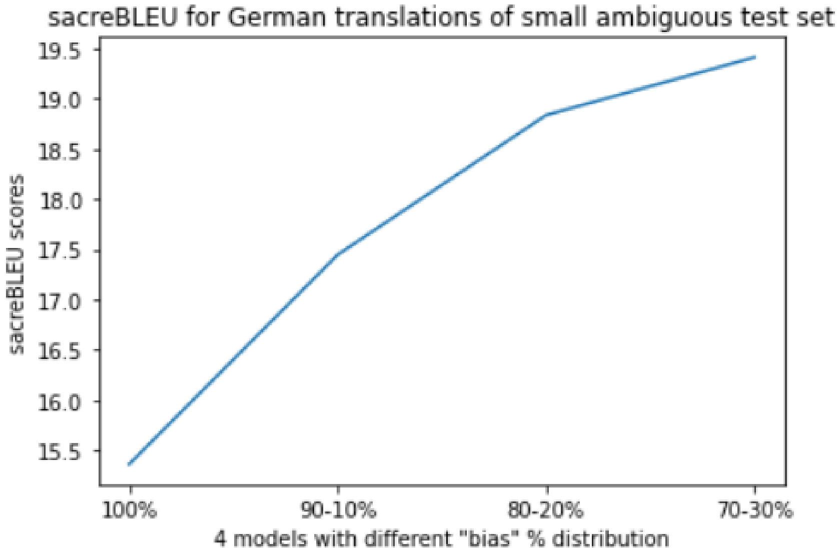


Figure 30. sacreBLEU scores for the German translation (of the small gender-ambiguous English test set) of each of the four models in comparison with the German (female) reference test text. Note: the y-axis does not start at 0 for better visualisation. The range between the minimum and maximum y-value (sacreBLEU score) is 4.0542.

These results are very interesting. As described above, if an NMT system is trained on a gender-biased dataset, the output translations will also be gender-biased. These results show that if the reference datasets are also female-gendered, as they theoretically should be if one aims for gender fairness in MT, then those translations containing more female-gendered terms and are less male-biased, reach a noticeably higher automatic evaluation metric – at least when looking at sacreBLEU scores – than those translations that are output by models trained on a male-biased dataset. The big difference in these sacreBLEU scores show how big of a difference a few percentage points can make of having a training data that is more or less gender-biased.

For the German translations of this smaller English test-set yielded by each of the four models, COMET and BERTScores were also calculated. Table 13 below shows the calculated COMET and F_{BERT} (as an average of P_{BERT} and R_{BERT}). The F-Measure values are also visualised in Figure 31 below.

	Model nr. 1 *the hunnies*	**Model nr. 2** *the nineties*	**Model nr. 3** *the eighties*	**Model nr. 4** *the seventies*
COMET	0.4409	0.3665	0.3314	0.4225
F_{BERT}	0.7519	0.7604	0.7664	0.7699

Table 13. COMET and F_{BERT} (F-Measure) scores for the German translations of the small gender-ambiguous English test set compared to the German reference text of all four smaller models.

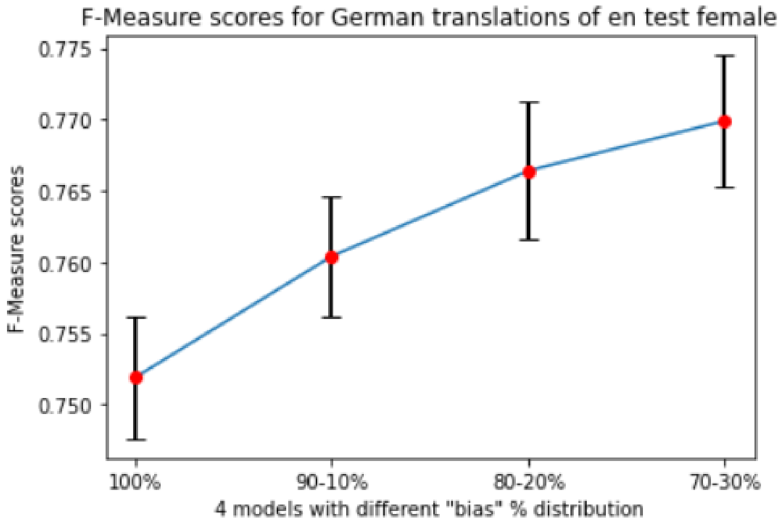


Figure 31. F_{BERT} scores for the German translations of the small gender-ambiguous English test set-compared to the German reference text of all four smaller models. Plotted with error bars.

Noticeable, here the F-Measure error bars (variance) are slightly lower at an average value of 0.0044, whereas the F-Measure error bars were noticeably larger at an average of 0.0066 in section 5.2.2 for the 12%-English test-set above. Similar to the sacreBLEU scores, the BERTScores follow a different pattern for the German translations calculated in this section in comparison to the German translations in sections 5.2.1 and 5.2.2. For the German translations yielded by the four models of the 12%-English test-set above, Model nr. 1 scores the highest BERTScore, with the values decreasing for each following model. In this section, however, for the German translations of the four models of the small ambiguous English test-set, Model nr. 1 scores the lowest BERTScores, with the values increasing for each model.

These results show that, when looking at an English test-set that contains a higher number of gender-ambiguous terms with these terms being gendered (with both male and female plural form) in the German reference dataset, then the four models score very differently. The 10% increment of sentences in the training dataset including plural female terms makes a measurable gender-difference in the translations, and those models trained with more female-gendered content score higher sacreBLEU and BERTScores. The COMET scores yielded in this section differ noticeably for each of the four models. Since no clear pattern emerged here, these values have not been plotted for visualisation.

This analysis section 5 visually and measurably shows that if a NMT model is trained on a male-biased dataset, the translation output will also be male-biased. If, however, the training data becomes less male-biased by including more female-gendered terms, then the translation output will also contain more female-gendered terms and be less gender-biased. Therefore, it can be expected that if NMT systems are trained on data that equally include 50% male and 50% female gendered terms, then the translation outputs should also equally include 50-50% male and female gendered terms, meaning it would be unbiased. This all refers to binary gender as explained in the beginning of this dissertation.

5.4 Gender-Evaluation of Commercial MT Tools in Comparison

The above sections show that gender-bias in a dataset used to train a MT system is directly reflected in translations done by these MT systems. The more gender-biased, usually male-skewed, a training dataset is, the more gender-biased, male-skewed, the translation will be. Usually, datasets used to train commercial MT systems are rather male-skewed and therefore gender-biased due to the data available being inherently biased.

To compare results (specifically focussing on gender) of how the MT models trained in the realm of this research translated the smaller ambiguous English test set, this smaller English test set discussed in section 5.3 was translated on DeepL and Google Translate. In the German translation yielded by DeepL, a total of 120 female plural terms ending in 'innen' were manually counted. In the German translation yielded by Google Translate, however, a total of 148 female plural terms ending in 'innen' were manually counted.

Both of these translations translated more English ambiguous terms into both the male and female term, like 'Kolleginnen und Kollegen', than Model nr. 2 (*the nineties*) tested in this research, which counts 110 plural female terms in the German translation, but less than Model nr. 3 (*the eighties*) trained in this research, which counts 150 plural female terms in the German translation. Model nr. 2 was trained on a dataset of which only 10% of sentences included a plural female term, and 90% didn't. Model nr. 3 was trained on a dataset of which 20% of sentences included a plural female term, and 80% didn't.

Interestingly, there is a rather noticeable gender-gap between the German translation yielded by DeepL and the one yielded by Google Translate. No automatic evaluation score was evaluated and compared for these online translations so the quality of these translations, besides focussing on the gender-aspect, was not evaluated and cannot be compared at this stage. Since the Europarl German reference dataset for this small English test-set counts 448 plural female terms ending in 'innen', it is very interesting to see how online available commercial MT systems such as DeepL and Google Translate, translate this test-set and how gender-biased they are in comparison to each other and in comparison to the models trained for this research. Figure 32 below depicts the four models in comparison with DeepL and Google Translate when counting the number of plural female terms ending in 'innen' present in the German translation of the small gender-ambiguous English test-set discussed in section 5.3.

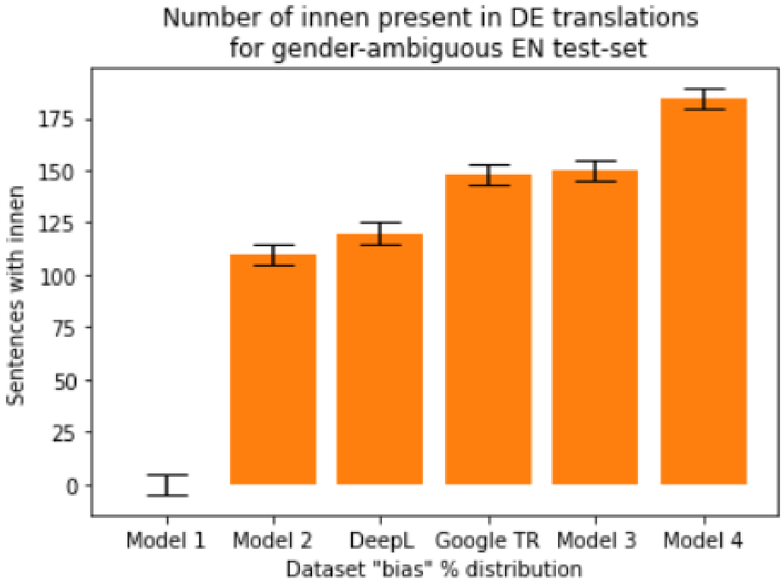


Figure 32. Bar chart of the number of sentences (with error bars) including female plural terms (words that end in 'innen') present in the German translations of the English test set for all four models as well as for DeepL and Google Translate

This shows that Models nr. 3 and nr. 4, whose datasets respectively contain 20% and 30% of sentences including plural female terms ending in 'innen', yield the highest output of plural female terms in the German translations. Disregarding the automatic evaluation of the quality of these translations (since the datasets used to train the four models above were very small and the translations include a number of 'unknowns'), it is very interesting to see what influence the gender-bias (male-skewedness) in the training data has on the output translations. It is also interesting to also compare the gender-results of the four models above with translations done by commercial MT systems DeepL and Google Translate.

6 Limitations and Further Research

6.1 Shortcomings of NMT

NMT harbours numerous challenges that need to be addressed. These challenges include gender-bias in translation outputs based on gender-bias in training corpora, limited availability of parallel corpora for researchers to train and evaluate NMT systems, and automatic evaluation metrics not being as accurate as human evaluation. Particularly the widely used and compared traditional BLEU metric is considered to be slightly outdated and cannot simply be used to overarchingly compare MT quality in different research approaches.

As societal needs develop, NMT will also need to adapt. Today, the topic of gender equality in society and gender-bias in MT is very prominent and much researched. In a few years, or even decades, other topics might be more pressing in society and will therefore need to be implemented in NMT. Shah, Schwartz and Hovy (2020) refer to this as ‘The Known and Unknown Unknowns’, stating that “[o]ne of the challenges in addressing selection bias is that we can not know *a priori* what sort of (demographic) attribute will be important to control” (5253). Today, gender-bias defines research in MT but tomorrow, it might be a different topic.

Large corpora are needed to train a NMT system: they “do not show robust behaviour when confronted with conditions that differ significantly from training conditions [such as] limited exposure to training data” (Koehn, 2017: 101). When training data does not cover a certain domain, a NMT system trained on that data will be limited to translate texts in said domain. As a result, poor translations from texts out of domain will be produced. Figure 33 below shows how much BLEU scores for MT models increase as the size of parallel training data increases.

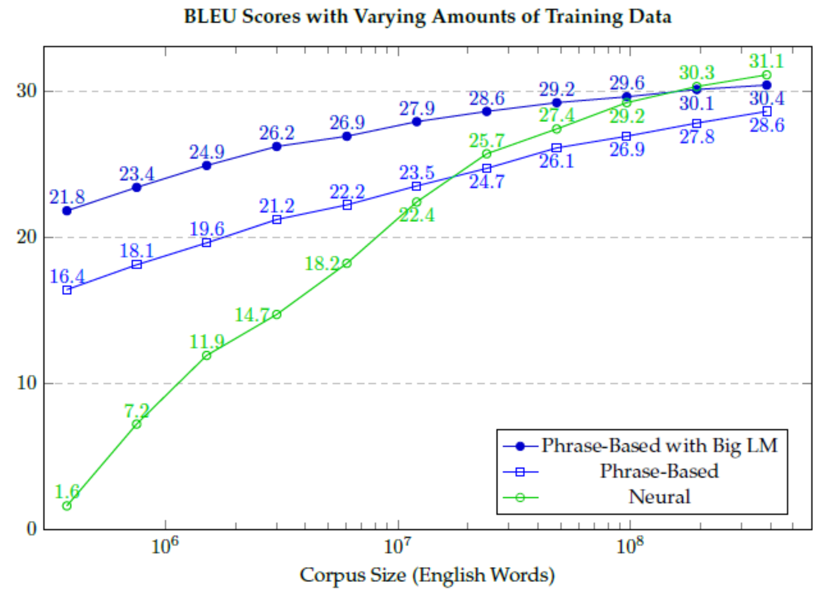


Figure 33. BLEU scores for three English-Spanish MT systems trained on training parallel data ranging from 0.4 million to 385.7 million words.

As depicted in figure 33, particularly the NMT system (green) shows an astonishing increase in BLEU scores in line with an increase in the size of parallel training data.

Automatic evaluation metrics serve as rather good indicators of quality in MT, but “do not provide any [actual] insight into the type of errors that occur in NMT model translations” (Jooste, Haque and Way, 2021: 296). Automatic evaluation metrics assess the overall quality of a machine translation, but a more detailed analysis would be necessary to understand the exact errors in the output translation. Jooste, Haque and Way (2021) agree that “[h]uman assessments are a lot more accurate than automatic evaluation methods [but are] slow and costly so cannot be used as frequently as one might like” (291). A continued improvement of automatic evaluation metrics, and specifically a standardisation of these metrics, for direct comparison is needed.

6.2 Limitations of this Research

Limitations of the research conducted will be outlined in this section. Most importantly, the hypothesis initially defined could be confirmed by this research, measurably showing that the amount of gender-bias in MT training data directly correlates with gender-bias in output translations. By increasing or decreasing the amount of gender-bias, the resulting translations will respectively reflect more or less gender-bias. However, there are a number of limitations to this research that can be adapted for continued research to provide even clearer and qualitatively higher results.

The limitation most directly affecting the qualitative results of this research is the small size of the datasets. Training a MT model on a dataset of less than 80,000 sentences per language is insufficient. In practice, MT systems are, and should be, trained on millions of sentence pairs. The translation quality noticeably suffered due to the small size of the datasets, with an overwhelming count of ‘unknown’ terms in the final translation. The models simply didn’t have access to enough vocabulary to translate new terminology present in the English test-sets. By training a MT model on larger corpora, the models can learn more extensive vocabulary and as a result translate test-sets at a much higher quality.

A second limitation, which could have helped counter the limitation encountered above, is that of pre-tokenization. Common practice in training MT systems is to tokenize the dataset in advance, for example by using Byte-Pair Encoding or SentencePiece (Kudo and Richardson, 2018). Tokenization is a necessary step to prepare natural language text computationally, as found in the parallel corpora, so that a computer can better process the input. A computer learns natural language text as encoded vectors. If datasets are pre-tokenized, words can be split up into smaller sections that the computer can better learn. The model can adapt these split word sections to other words when facing new terminology. As part of this research, a SentencePiece model was trained using the code available on GitHub[14]. When tokenizing “This is an English sentence.” And “Dies ist ein Deutscher Satz.”, the following was a resulting tokenized output:

['▁Th', 'is', '▁is', '▁an', '▁En', 'gl', 'ish', '▁', 's', 'en', 'ten', 'ce', '.']
['▁Dies', '▁ist', '▁ein', '▁Deutsch', 'er', '▁Satz', '.']

The entire dataset was equally tokenized. In practice, a NMT model then trains on this tokenized data and outputs a translation. The resulting translation is then detokenized (to natural language). Tokenization enables a NMT system to cope much better with new vocabulary. In this research, certain ‘unknowns’ were present post detokenization and prevented the calculation of automatic evaluation metrics. For that reason, the NMT models trained on pre-tokenized datasets were not analysed as part of this research. For future research, however, pre-tokenization will be the next step to achieve translations of higher quality.

Another limitation – not specifically a limitation to this research but something that could be worked on to further improve the quality of the translation outputs – is to train the four NMT models by applying the Transformer architecture. In this research, the NMT systems were trained by an RNN-model with attention made available by OpenNMT-py. This model had been developed for NMT for many years and is capable of training models well to provide good translation qualities. However, the newest model is the Transformer with self-attention as developed by Vaswani et al. (2017). Recent research has been conducted into training NMT models by applying the Transformer architecture. Models trained on this architecture provide measurably better qualitative translations. Using the Transformer architecture is not a necessity, but it can help achieve translation results of higher quality. Therefore, training NMT models on a Transformer architecture would be the next step of this research. However, more computational knowledge is required to understand and employ the provided, pre-written code of a Transformer, which is why this is an aspect to be developed for future research.

The low visibility of gendered terms in the Europarl training dataset is another limitation. As previously mentioned in section 4.3.3, the Europarl English-German parallel corpus incorporates very formal language, resulting from the proceedings of the European Parliament. Therefore, the corpus does not include many gendered terms. In the German dataset, primarily the generic masculine, particularly for plural terms, is used. The amount of possible gender present in the parallel corpus overall is rather small, but those terms that could be gendered in German (like ‘colleagues’) are mostly represented as plural male generic terms. A lack of publicly available large parallel corpora makes MT research difficult. With a higher availability of parallel corpora containing more natural speech, the effect of gender-bias in training data reflected in translations could be better evaluated and demonstrated. Repeating this research approach by training MT models on corpora containing more natural speech will also be a next step to yielding qualitatively higher results.

[14] https://github.com/google/sentencepiece

If similar research were conducted by training a MT model on a dataset containing more natural speech and therefore also more phrases containing gender, Morphy could be applied for data analysis. Morphy is “an integrated tool for German morphology, part-of-speech tagging and context-sensitive lemmatization” (Lezius, Rapp and Wettler, 1998: 743), including gender. An updated version DE Morphy is available on Github[15]. Applying DE Morphy could help analyse the presence of gender in a dataset before training a NMT model as well as in the translated datasets.

6.3 Future NMT Research

The aim of this research was to measurably show that the amount of gender-bias present in datasets used to train MT systems is directly reflected in resulting translations. The broader question is how to prevent gender-bias naturally occurring in training datasets to be reflected in machine translations. One solution shown in this research is to create a more gender-balanced dataset to include more female-gendered terms, instead of the generic masculine. To achieve gender-neutrality in machine translations, the training data would need to be gender-balanced 50-50%. The German Europarl corpus, one of the most commonly used corpora to train MT systems in research, naturally only contains around 1.5% of female-gendered plural terms. Since it would not be a viable solution to massively cut parallel corpora to such small sizes for them to contain an equal and balanced amount of male and female gender, a different approach will most likely need to be taken. The method evaluated in this research could potentially be done in combination with other approaches.

Furthermore, the methodology followed in this research focussed on the plural generic masculine. In section 3, where different linguistic phenomena of gender-bias occurring in MT are outlined, other patterns, such as professions, adjectives and trigger words majorly affecting gender-bias in machine translation have not been the focus of this research. A different approach would be needed to eliminate gender-bias in those situations, to reflect society more fairly.

It is essential to create or adapt datasets to be gender-balanced, both for binary and non-binary individuals, to then train NMT systems on these datasets. To do so, both a computational aspect of balancing existing datasets is required as well as a shift in society’s mentality and evolution of professions in society. The existing imbalance in society of certain professions (doctors, nurses etc.) or associations (pretty, smart etc.) that stereotypically relate to a male or female gender is the underlying issue. MT systems learn this inherent gender-imbalance and reflect it in their translation outputs. Therefore, also a shift in society, which leads to the creation of datasets, is required to help remove stereotypes of professions and certain adjectives or associations.

A shift in society will not occur over night, which is why a computational solution is also needed to help create more gender-fair datasets. Approaches could be to skew datasets

[15] https://github.com/DuyguA/DEMorphy

to contain a higher percentage of female-gendered terms, as done as part of this research, or to give female-gendered terms more weight than male-gendered terms during training. Another example would be to always provide a female and a male (and then also a non-binary) option in the target translations for gender-ambiguous terms in the source language. This is a doable option for the translation of a single sentence but less realistic for the translation of a paragraph or an entire document. These are just some examples to underline a few possibilities of creating more gender-fair machine translation. Clearly, much brainstorming and research is needed to find appropriate and feasible solutions to eliminate gender-bias in MT.

7 Conclusion

Within AI and NLP, MT is a rapidly evolving field with MT systems relying on large amounts of data. Unfortunately, natural language data, such as datasets used to train MT systems, inevitably reflect biases present in our society (Saunders and Byrne, 2020: 7724). Since the data fed into a MT system directly affects the resulting translations, these translations reflect the biases contained within the data. Such biases include, but are not limited to, racial and gender bias. Gender-bias in everyday speech, academics and on the German news is increasingly debated, and therefore it becomes more important to update MT-outputs appropriately to not reflect gender bias.

The experimental research conducted for this dissertation shows that the amount of gender-bias present in datasets used to train MT systems directly affects the amount of gender-bias reflected in machine translations. The hypothesis defined for this dissertation could be confirmed. It could be measurably shown that if a MT system is trained on a small dataset that is more gender balanced, the output translations will reflect much less gender-bias in comparison to when a MT system is trained on an equally small dataset, which, however, contains more gender-bias. To summarise, it can be measured that the amount of bias present in the output translations directly correlate with the amount of bias present in the datasets.

The result of this research highlights the necessity for unbiased or gender-balanced datasets to train NMT systems. Datasets need to become more gender-balanced to fairly reflect the diversity in society. Only when MT systems are trained on more gender-balanced datasets, these MT systems can yield more gender-balanced translations. Gender-fair NMT can positively influence individuals that are negatively impacted by biased MT-texts and can aim to create a more gender-fair society.

Bibiliography

Bahdanau, Dzmitry; Cho, KyungHyun; Bengio, Yoshua (2014): Neural Machine Translation by Jointly Learning to Align and Translate. *ICLR 2015 Conference*. Published in CoRR: 1–15.

Banarjee, Satanjeev; Lavie, Alon (2005): METEOR: An Automatic Metric for MT Evaluation with Improved Correlation with Human Judgements. In: Goldstein, Jade; Lavie, Alon ;Lin, Chin-Yew; Voss, Clare (eds): *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Association for Computational Linguistics: 65–72. https://aclanthology.org/W05-0909

Basta, Christine; Costa-Jussà, Marta R.; Follonosa, José A. R. (2020): Towards Mitigating Gender Bias in a decoder-based Neural Machine Translation model by Adding Contextual Information. In: Cunha, Rossana; Shaikh, Samira; Varis, Erika; Georgi, Ryan; Tsai, Alicia; Anastasopoulos, Antonios; Raghavi Chandu, Khyathi (eds): *Proceedings of the The Fourth Widening Natural Language Processing Workshop*. Association for Computational Linguistics: 99–102. https://aclanthology.org/2020.winlp-1.25/

Bender, Emily M.; Gebru, Timnit; McMillan-Major, Angelina; Shmitchell, Shmargaret (2021): On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? In: *FAccT'21: Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*: 610–623. https://doi.org/10.1145/3442188.3445922

Bolukbasi, Tolga; Chang, Kai-Wei; Zou, James Y; Saligrama, Venkatesh; Kalai, Adam T. (2016): Man is to computer programmer as woman is to homemaker? In: Lee, Daniel D.; von Luxburg, Ulrike; Garnett, Roman; Sugiyama, Masashi; Guyon, Isabelle (eds): *NIPS'16: Proceedings of the 30th International Conference on Neural Information Processing Systems*: 4356–4364. https://dl.acm.org/doi/10.5555/3157382.3157584

Brownlee, Jason (2020): A Gentle Introduction to Cross-Entropy for Machine Learning, Machine Learning Mastery. https://machinelearningmastery.com/cross-entropy-for-machine-learning

Caliskan, Aylin; Bryson, Joanna J; Narayanan, Arvind (2017): Semantics derived automatically from language corpora contain human-like biases. In: *Science* 356(6334): 183–186. DOI: 10.1126/science.aal4230

Sharma, Alek (2022): What is YAML? A beginner's guide, CircleCI. https://circleci.com/blog/what-is-yaml-a-beginner-s-guide

Costa-Jussà, Marta R.; de Jorge, Adrià (2020): Fine-tuning Neural Machine Translation on Gender-Balanced Datasets. In: Costa-jussà, Marta R.; Hardmeier, Christian;

Radford, Will; Webster, Kellie (eds): *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*. Association for Computational Linguistics: 26–34. https://aclanthology.org/2020.gebnlp-1.3

Dudenredaktion (2022a): Die geschlechtsübergreifende Verwendung mas-ku-liner Formen. In: Duden. https://www.duden.de/sprachwissen/sprachratgeber/generische-verwendungsweise-maskuliner-formen

Dudenredaktion (2022b): Geschlechtergerechter Sprachgebrauch. In: Duden. https://www.duden.de/sprachwissen/sprachratgeber/Geschlechtergerechter-Sprachgebrauch

Font, Joel Escudé; Costa-Jussà, Marta R. (2019): Equalizing Gender Bias in Neural Machine Translation with Word Embedding Techniques.In: Costa-jussà, Marta R.; Hardmeier, Christian; Radford, Will; Webster, Kellie (eds): *Proceedings of the First Workshop on Gender Bias in Natural Language Processing.* Association for Computational Linguistics: 147–154. https://aclanthology.org/W19-3821

Forcada, Mikel L. (2017): Making sense of neural machine translation. In: Translation Spaces 6:2 (2017) 291–309. https://doi.org/10.1075/ts.6.2.06for

Google Colab (n.d.a): https://colab.research.google.com/notebooks/basic_features_overview.ipynb

Google Colab (n.d.b): https://colab.research.google.com/signup

Gonen, Hila; Goldberg, Yoav (2019): Lipstick on a Pig: Debiasing Methods Cover up Systematic Gender Bias in Word Embeddings But do not Remove Them. In: Burstein, Jill; Doran, Christy; Solorio, Thamar (eds): *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)*. Association for Computational Linguistics: 609–614. https://aclanthology.org/N19-1061

Hahn, Michael (2020): Theoretical Limitations of Self-Attention in Neural Sequence Models. In: Johnson, Mark; Roark, Brian; Nenkova, Ani (eds): *Transactions of the Association for Computational Linguistics, Volume 8*. MIT Press: 156–171. https://aclanthology.org/2020.tacl-1.11

Hovy, Dirk; Bianchi, Federico; Fornaciari, Tommaso (2020): “You Sound Just Like Your Father” Commercial Machine Translation Systems Include Stylistic Biases. In: Jurafsky, Dan; Chai, Joyce; Schluter, Natalie; Tetreault, Joel (eds): *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics: 1686–1690. https://aclanthology.org/2020.acl-main.154

Hovy, Dirk; Spruit, Shannon L. (2016): The Social Impact of Natural Language Processing. In: Erk, Katrin; Smith, Noah A. (eds): *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*.

Association for Computational Linguistics: 591–598. https://aclanthology.org/P16-2096

Jooste, Wandri; Haque, Rejwanul; Way, Andy (2021): Philipp Koehn: Neural Machine Translation. In: *Machine Translation* 35: 289–299. https://doi.org/10.1007/s10590-021-09277-x

Klein, Guillaume; Hernandez, Francois; Nguyen, Vincent; Senellart, Jean (2020a): The OpenNMT Neural Machine Translation Toolkit: 2020 Edition. In: Denkowski, Michael; Federmann, Christian (eds): *Proceedings of the 14th Conference of the Association for Machine Translation in the Americas (Volume 1: Research Track)*. Association for Machine Translation in the Americas: 102–109. https://aclanthology.org/2020.amta-research.9

Klein, Guillaume; Kim, Yoon; Deng, Yuntian; Senellart, Jean; Rush, Alexander M. (2017): OpenNMT: Open-Source Toolkit for Neural Machine Translation. In: Bansal, Mohit; Ji, Heng (eds): *Proceedings of ACL 2017, System Demonstrations*. Association for Computational Linguistics: 67–72. https://aclanthology.org/P17-4012

Klein, Guillaume; Zhang, Dakun; Chouteau Clement; Crego, Josep; Senellart, Jean (2020b): Efficient and High-Quality Neural Machine Translation with OpenNMT. In: Birch, Alexandra; Finch, Andrew, Hayashi, Hiroaki; Heafield, Kenneth; Junczys-Dowmunt, Marcin; Konstas, Ioannis; Li, Xian; Neubig, Graham; Oda, Yusuke (eds): *Proceedings of the Fourth Workshop on Neural Generation and Translation*. Association for Computational Linguistics: 211–217. https://aclanthology.org/2020.ngt-1.25

Koehn, Philip (2005): Europarl: A Parallel Corpus for Statistical Machine Translation. In: *Proceedings of Machine Translation Summit X: Papers*: 79–86. https://aclanthology.org/2005.mtsummit-papers.11

Koehn, Philip (2017): Draft of Chapter 13: Neural Machine Translation. In: *Statistical Machine Translation*. Cambridge: University Press: 1–117. https://doi.org/10.1017/CBO9780511815829

Koehn, Philip (2020): *Neural Machine Translation*. Cambridge: University Press.

Krüger, Ralph (2019): Lenkende Einflüsse von Übersetzungstechnologie auf den Fachübersetzungsprozess. In: Ahrens, Barbara; Hansen-Schirra, Silvia; Krein-Kühle, Monika; Schreiber, Michael; Wienen, Ursula (eds.): *Translation – Fachkommunikation – Fachübersetzung*. Berlin: Frank & Timme: 29-65. https://www.frank-timme.de/verlag/verlagsprogramm/buch/page/3/verlagsprogramm/barbara-ahrens-silvia-hansen-schirramonika-krein-kuehlemichael-schreiberursula-wienen-hg-tran-4/backPID/separate-titel-2.html

Krüger, Ralph (2020a): Expliciation in Neural Machine Translation. In: *Across Languages and Cultures* 21(2): 195–216. http://dx.doi.org/10.1556/084.2020.00012

Krüger, Ralph (2020b): Propositional opaqueness as a potential problem for neural machine translation. In: Ahrens, Barbara; Krein-Kühle, Monika; Wienen, Ursula; Hansen-Schirra, Silvia; Schreiber, Michael (eds.): *Translation – Kunstkommunikation – Museum / Translation – Art Communication*. Berlin: Frank & Timme: 261–278. https://www.frank-timme.de/verlag/verlagsprogramm/buch/page/2/verlagsprogramm/barbara-ahrensmorven-beaton-thomemonika-krein-kuehleralph-krueger-lisa-linkursula-wienen-hged/backPID/separate-titel-17.html

Krüger, Ralph (2021): Die Transformer-Architektur für Systeme zur neuronalen maschinellen Übersetzung – eine popularisierende Darstellung. In: *trans-kom* 14(2): 278–324. https://www.trans-kom.eu/bd14nr02/trans-kom_14_02_05_Krueger_NMUe.20211202.pdf

Kudo, Taku; Richardson, John (2018): SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing. In: Blanco, Eduardo; Lu, Wei (eds): *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics: 66–71. https://aclanthology.org/D18-2012

Lezius, Wolfgang; Rapp, Reinhard; Wettler, Manfred (1998): A Freely Available Morphological Analyzer, Disambiguator and Context Sensitive Lemmatizer for German. In: *COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics*: 743–748. https://aclanthology.org/C98-2118

Mirkin, Shachar; Nowson, Scott; Brun, Caroline; Perez, Julien (2015): Motivating Personality-aware Machine Translation. In: Màrquez, Lluís; Callison-Burch, Chris; Su, Jian (eds): *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics: 1102–1108. https://aclanthology.org/D15-1

Papineni, Kishore; Roukos, Salim; Ward, Todd; Zhu, Wei-Jing (2002): BLEU: A Method for Automatic Evaluation of Machine Translation. In: *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*: Association for Computational Linguistics: 311–318. https://doi.org/10.3115/1073083.1073135

Popović, Maja (2015): CHRF: character *n*-gram F-score for automatic MT evalution. In: Bojar, Ondrej; Chatterjee, Rajan; Federmann, Christian; Haddow, Barry; Hokamp, Chris; Huck, Matthias; Logacheva, Varvara; Pecina, Pavel (eds): *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Association for Computational Linguistics: 392–395. https://aclanthology.org/W15-3049

Popović, Maja (2017): Comparing Language Related Issues for NMT and PBMT between German and English. In: *The Prague Bulletin of Mathematical Linguistics, Number 108*: 209–220.

Post, Matt (2018): A Call for Clarity in Reporting BLEU Scores. In: Bojar, Ondrej; Chatterjee, Rajen; Federmann, Christian; Fishel, Mark; Graham, Yvette; Haddow, Barry; Huck, Matthias; Jimeno Yepes, Antonio; Koehn, Philipp; Monz, Cristof; Negri, Matteo; Névéol, Aurélie; Neves, Mariana; Post, Matt; Specia, Lucia; Turchi, Marco; Verspoor, Karin (eds): *Proceedings of the Third Conference on Machine Translation: Research Papers*. Association for Computational Linguistics: 186–191. https://aclanthology.org/W18-6319

Rabinovich, Ella; Mirkin, Shachar; Patel, Raj Nath; Specia, Lucia; Wintner, Shuly (2017): Personalized Machine Translation: Preserving Original Author Traits. In: Lapata, Mirella; Blunsom, Phil; Koller, Alexander (eds): *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Association for Computational Linguistics: 1074–1084. https://aclanthology.org/E17-1101

Rei, Ricardo; Stewart, Craig; Farinha, Ana C; Lavie, Alon (2020): COMET: A Neural Framework for MT Evaluation. In: Webber, Bonnie; Cohn, Trevor; He, Yulan; Liu, Yang (eds): *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics: 2685–2702. https://aclanthology.org/2020.emnlp-main.213

Romanov, Alexey; De-Arteaga, Maria; Wallach, Hanna; Chayes, Jennifer; Borgs, Christian; Chouldechova, Alexandra; Geyik, Sahin; Kenthapadi, Krishnaram; Rumshisky, Anna; Kalai, Adam Tauman (2019) What's in a name? reducing bias in bios without access to protected attributes. In: Burstein, Jill; Doran, Christy; Solorio, Thamar (eds): *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)*. Association for Computational Linguistics: 4187–4195. https://aclanthology.org/N19-1424

Saunders, Danielle; Byrne, Bill (2020): Reducing Gender Bias in Neural Machine Translation as a Domain Adaptation Problem. In: Jurafsky, Dan; Chai, Joyce; Schluter, Natalie; Tetreault, Joel (eds): *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics: 7724–7736. https://aclanthology.org/2020.acl-main.690

Shah, Deven; Schwartz, H. Andrew; Hovy, Dirk (2020): Predictive Biases in Natural Language Processing Models: A Conceptual Framework and Overview. In: Jurafsky, Dan; Chai, Joyce; Schluter, Natalie; Tetreault, Joel (eds): *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics: 5248–5284. https://aclanthology.org/2020.acl-main.468

Stanovsky, Gabriel; Smith, Noah A.; Zettlemoyer, Luke (2019): Evaluating Gender Bias in Machine Translation. In: Korhonen, Anna; Traum, David; Màrquez, Lluís (eds): *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics: 1679–1684. https://aclanthology.org/P19-1164

Tilburg University (2021): Dr. Eva Vanmassenhove on gender bias in Machine Translation Systems. On: YouTube. https://www.youtube.com/watch?v=Mlja7xYKn1U

Way, Andy (2019): Chapter 14. Machine Translation. Where are we at today? In: Angelone, Erik; Ehrensberger-Dow, Maureen; Massey, Gary (eds): *The Bloomsbury Companion to Language Industry Services*: 311–332. Bloomsbury Companions. Bloomsbury Academic Publishing, NY, USA. ISBN 9781350024939

Way, Andy (2018): Quality Expectations of Machine Translation. In: Moorkens, Joss; Castilho, Sheila; Gaspari, Federico; Doherty, Stephen; (eds): *Translation Quality Assessment*. Springer International Publishing. ISBN: 978-3-319-91241-7

Vanmassenhove, Eva; Emmery, Chris; Shterionov, Dimitar (2021a): NeuTral Rewriter: A Rule-Based and Neural Approach to Automatic Rewriting into Gender-Neutral Alternatives. In: Moens, Marie-Francine; Huang, Xuanjing; Specia, Lucia; Yi, Scott Wen—tau (eds): *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics: 8940–8948. https://aclanthology.org/2021.emnlp-main.704

Vanmassenhove, Eva; Hardmeier, Christian; Way, Andy (2018): Getting Gender Right in Neural Machine Translation. In: Riloff, Ellen; Chiang, David; Hockenmaier, Julia; Tsujii, Jun'ichi (eds): *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics: 3003–3008. https://aclanthology.org/D18-1334

Vanmassenhove, Eva; Shterionov, Dimitar; Gwilliam, Matthew (2021): Machine Translationese: Effects of Algorithmic Bias on Linguistic Complexity in Machine Translation. In: Merlo, Paola; Tiedemann, Jorg; Tsarfaty, Reut (eds): *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics: 2203–2213. https://aclanthology.org/2021.eacl-main.188

Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Lilion; Gomez, Adrian N., Kaiser, Lukasz; Polosukhin, Illia (2017): Attention is All You Need. In: von Luxburg, Ulrike; Guyon, Isabelle; Bengio, Samy; Wallach, Hanna; Fergus, Rob (eds.): NIPS'17: *Proceedings of the 31st International Conference on Neural Information Processing Systems*: 6000–6010. ISBN: 9781510860964

Wang, Ruibo; Li, Jihong (2019): Bayes Test of Precision, Recall, and F1 Measure for Comparison of Two Natural Language Processing Models. In: Korhonen, Anna; Traum, David; Màrquez, Lluís (eds.): *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics: 4135–4145. https://aclanthology.org/P19-1405

Zhang, Tianyi; Kishore, Varsha; Wu, Felix; Weinberger, Kilian Q.; Artzi, Yoav (2020): BERTScore: Evaluating Text Generation With BERT. ICLR 2020 Conference Blind Submission

Zhao, Jieyu; Mukherjee, Subhabrata; Hosseini, Saghar; Chang, Kai-Wei; Awadallah, Ahmed (2020): Gender Bias in Multilingual Embeddings and Cross-Lingual Transfer. In: Jurafsky, Dan; Chai, Joyce, Schluter, Natalie, Tetreault, Joel (eds): *Proceedings* *of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics: 2896–2907. https://aclanthology.org/2020.acl-main.260

Zhao, Jieyu; Zhou, Yichao; Li, Zeyu; Wang, Wie; Chang, Kai-Wie (2018): Learning gender-neutral word embeddings. In: Riloff, Ellen; Chiang, David, Hockenmaier, Julia; Tsujii, Jun'ichi (eds): *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics: 4847–4853. https://aclanthology.org/D18-1521

Appendix

The appendix of this dissertation can be found in a Google Drive Folder. The link to this folder is the following:

https://drive.google.com/drive/folders/123WmbHk2qgT1lurNz8NYdlNBO5jXDH7l?usp=sharing

In this folder you‘ll find:

- 4 Copies of my Google Colab Notebooks
  - Master’s Thesis: NMT-Models
  - Data Preparation
  - Output & Analysis
  - A test version of trying to tokenize and detokenize data applying SentencePiece
- 2 English Test-Sets
  - 1. Of 12% of the training data
  - 2. Smaller gender-ambiguous test-set
- 2 German References
  - 1. Of the 12% English test-set
  - 2. Of the smaller gender-ambiguous English test-set
- 7 German Translations
  - Model nr. 4’s German translation of 12% English test-set
  - Model nr. 4’s German translation of gender-ambiguous English test-set
  - Model nr. 3’s German translation of 12% English test-set
  - Model nr. 3’s German translation of gender-ambiguous English test-set
  - Model nr. 2’s German translation of 12% English test-set
  - Model nr. 2’s German translation of gender-ambiguous English test-set
  - Model nr. 1’s German translation of 12% English test-set
  - Model nr. 1’s German translation of gender-ambiguous English test-set
  - Complete Europarl corpus’ German translation of 12% English test-set
- Complete Europarl Dataset
  - Complete English corpus
  - Complete German corpus