



Gender Bias in Machine Translation: Improving Gendered Disambiguation of Pronouns Using Context-Aware Translation

Citation

Xu, Susan. 2019. Gender Bias in Machine Translation: Improving Gendered Disambiguation of Pronouns Using Context-Aware Translation. Bachelor's thesis, Harvard College.

Link

<https://nrs.harvard.edu/URN-3:HUL.INSTREPOS:37364657>

Terms of use

This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material (LAA), as set forth at

<https://harvardwiki.atlassian.net/wiki/external/NGY5NDE4ZjgzNTc5NDQzMGIzZWZhMGFIOWI2M2EwYTg>

Accessibility

<https://accessibility.huit.harvard.edu/digital-accessibility-policy>

Share Your Story

The Harvard community has made this article openly available.
Please share how this access benefits you. [Submit a story](#)

HARVARD UNIVERSITY

UNDERGRADUATE THESIS

**Gender Bias in Machine Translation:
Improving Gendered Disambiguation of
Pronouns using Context-Aware Translation**

Author:
Susan XU

Advisor:
Prof. Alexander M. RUSH

*A thesis submitted in partial fulfillment of the requirements
for the degree of A.B. in Computer Science with Honors*

March 29, 2019

Abstract

Susan XU

*Gender Bias in Machine Translation: Improving Gendered
Disambiguation of Pronouns using Context-Aware Translation*

We examine gender bias in machine translation in the context of disambiguating genderless pronouns to gendered ones. For example, Turkish uses the genderless pronoun ‘o’, which can be translated to ‘he’ or ‘she’. We argue that context-aware translation, a type of machine learning model that incorporates more context from the text than current models, is a possible method of improving this disambiguation.

We present a task to quantify how gender-biased one translation model is compared to another by applying a translation model in a classification setting to disambiguate a genderless pronoun into a gendered one. We define a ‘parity score’, given as the harmonic mean of the class-conditional accuracies.

We find that a baseline model is able to achieve relatively high translation quality (31 BLEU) but fails the classification task (parity = 0). We then experiment with using context-aware translation to improve performance on this task while still attaining a high level of translation quality. We compare the use of a context-aware translation model to standard debiasing methods in natural language processing and find that balancing the dataset to have an equal frequency of male and female pronouns in combination with a context-aware model yields the best results on our task (parity = .42).

Acknowledgements

First and foremost, thank you to my advisor, Professor Alexander M. Rush. This thesis would not exist without your continuous mentorship, support, and guidance. Thank you for introducing me to the world of NLP, pushing me to become a better scientist, and encouraging me throughout the ups and downs.

Thanks also to the Harvard Computer Science department, for continuously challenging me and providing me with a rigorous technical foundation without which this would not be possible. Special thanks are owed to Professors Stuart Shieber and Finale Doshi-Velez for reading this work and offering feedback. I am also grateful to my academic advisor, Professor Stephen Chong, for encouraging me to pursue research in our advising meetings and for letting me play with Annie when I was stressed. Thank you!

Finally, thank you to my friends and family for their never-ending support. In particular, thanks to Shira Li, for the many hours in MD spent coding together, and Katherine Binney, for many words of encouragement. Thesising didn't feel so lonely because of you both. My blockmates and best friends, Angeline Rao and Lily Xu. Thanks for the laughs, debates, and memories made over the years. Last but not least, Mom, Dad, and Lin. Thanks for encouraging me throughout all my endeavors throughout my life.

Contents

Abstract	i
Acknowledgements	ii
1 Introduction	1
2 Background	5
2.1 Definitions	5
2.2 Neural Machine Translation Models	6
2.2.1 RNNs	6
2.2.2 Attention	7
2.2.3 Transformer	8
2.3 Context-Aware Translation	12
2.4 Anaphora Resolution and Coreference Resolution	13
2.5 Bias and Fairness	15
3 Related Work	16
3.1 Bias and Fairness	16
3.2 Translation	17
3.2.1 Bias Quantification and Model Evaluation	17
3.2.2 Debiasing Methods	18
3.3 The Transformer: A Potential Discourse Solution	19
4 Data Analysis	20
4.1 Distribution of Training Examples	20
4.2 Frequencies of Pronouns Given Occupation	21
4.3 Can Context Help Disambiguate Pronouns?	23
5 Methods	27
5.1 Data	27
5.2 Models	28
5.3 Experiments	29
5.4 Evaluation Methods	31

5.4.1	BLEU	31
5.4.2	Pronoun Accuracy	31
6	Results and Discussion	33
6.1	Results	33
6.2	Discussion	33
6.2.1	Experiment 1: Randomly Sampled Data	33
6.2.2	Experiment 2: Gender Balanced Data	36
6.2.3	Experiment 3: Extra Coreference Dataset	37
6.2.4	Experiment 4: Impact of Debaised Word Embeddings	39
6.2.5	Summary of Results	41
7	Conclusion	42
7.1	Future Work	43
	Bibliography	45

List of Figures

1.1	Google Translate	2
1.2	Google Translate	3
2.1	Attention Mechanism	9
2.2	Context-aware Encoder	12
4.1	Distribution of translations of an ambiguous pronoun	21
4.2	Male Occupation Pronouns	23
4.3	Female Occupation Pronouns	24
4.4	Anaphora Examples: Pronoun Frequencies	25
4.5	Pronoun Frequencies in Anaphoric Turkish Examples with O	26
6.1	Examples generated from the baseline Transformer.	35
6.2	Comparison of examples from the baseline Transformer and the context-aware Transformer.	36
6.3	Comparison of examples from both context-aware and non-context aware models trained on gender-balanced data, along with examples from the baseline model.	37
6.4	Comparison of examples from both context-aware and non-context aware models trained on data with extra examples of coreference, along with examples from the baseline model.	38
6.5	Comparison of examples from a model that uses debiased word embeddings compared to one that does not.	40

List of Tables

4.1	The normalized fraction of examples that had a male pronoun compared to examples that had a female pronoun.	22
5.1	Hyperparameters	29
5.2	Comparison of the normalized fraction of examples that had a male pronoun compared to examples that had a female pronoun in both the baseline dataset and the ‘gender-balanced’ training dataset.	31
6.1	Test BLEU scores and parity scores on the pronoun accuracy task	34
6.2	Results from Experiment 4, which experimented with using debiased word embeddings.	39

Chapter 1

Introduction

In recent years, machine learning has achieved widespread success through the combination of deep learning on both models and datasets that are much larger than in the past. In the current age of ‘big’ data, data is measured not in megabytes or gigabytes, but terabytes and exabytes. On the surface, these methods seem to be a runaway success. They are capable of recommending us new videos (Covington, Adams, and Sargin, 2016), classifying different types of images (He et al., 2015), translating from English to other languages (Cho et al., 2014), and even beating the world’s best Go player at his own game (Silver et al., 2016). When we can now train on all the data in the world, the sky seems to truly be the limit.

This is the fantasy. However, as these systems take on greater responsibilities within society, an important question is whether or not their behavior is fair. Unfortunately, machine learning systems, especially the deep learning systems popular today, have been shown to perpetuate and even augment existing human biases present in the data. For example, recidivism prediction models disproportionately predict that black criminal defendants will reoffend (Chouldechova, 2017), and resume screening models give lower scores to female resumes (Dastin, 2018). Image recognition systems perform worse on black faces (Buolamwini and Gebru, 2018).

Within natural language processing (NLP), one might think that language is a field resistant to gender biases due to the unstructured nature of text, particularly with respect to language generation tasks. Yet, gender biases have been found in autocomplete suggestions (Lapowsky, 2018) and accuracy in speech recognition (Tatman, 2017). As it is common to use pre-trained NLP systems and components, the risk of embedded biases in systems spreads even more. Now, as machine learning systems are scaled to even larger datasets and claim to achieve even better results, the dangers of biased machine learning have moved from a theoretical risk discussed in academia to very real risks that can affect millions of users.

In fact, they already have. In late 2017, Google received considerable negative press for handling the translation of the Turkish pronoun ‘o’ in a decidedly biased way. ‘O’ is a third-person genderless pronoun and can mean ‘he’, ‘she’, or ‘it’ in English depending

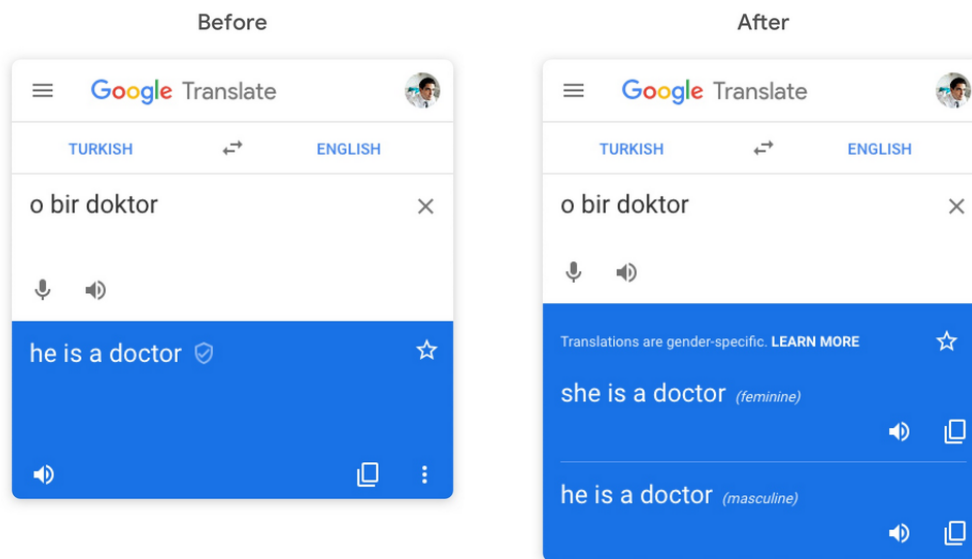


FIGURE 1.1: Screenshots of Google Translate from before and after the change, taken from the Google Blog (Kuczmarski, 2018). Before December 2018, only one translation was shown. After December 2018, both potential translations are given. However, if the input is sufficiently complex or contains multiple sentences, only one translation will be shown.

on the context. Without context, ‘o’ is completely ambiguous. In the presence of this uncertainty, Google Translate translated

O bir doktor. → He is a doctor.

but translated

O bir hemşire. → She is a nurse.

This phenomenon was not an isolated incident, nor even unique to Turkish. It also occurred when translating from Chinese and Malay, two other languages that exhibit a genderless personal pronoun. One year later, Google’s response was to add functionality that showed both translations of ‘he’ and ‘she’ when the input translation was sufficiently simple.

Google’s problem prompts a more careful examination of what is ‘fair’, as it relates to the resolution of an ambiguous pronoun. Is it ‘fair’ to reflect patterns in the training data, where male pronouns are statistically much more frequent? Or perhaps ‘gender-fairness’ is being probabilistically balanced across pronouns, so male and female pronouns are equally likely to occur?

Under the first definition, Google’s original behavior of predicting ‘he is a doctor’

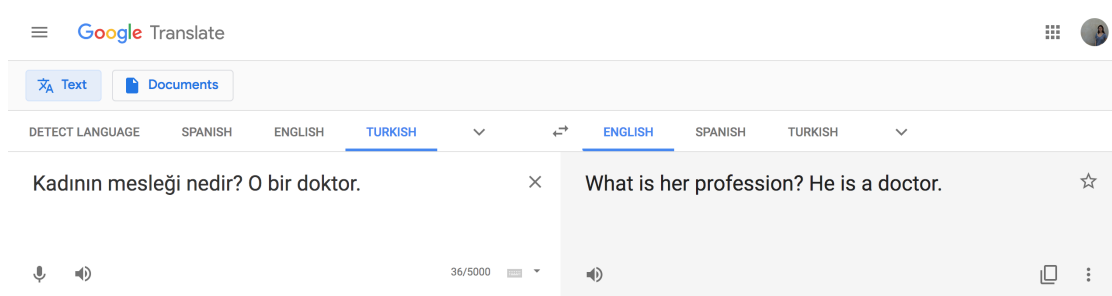


FIGURE 1.2: Screenshot of Google Translate incorrectly resolving a pronoun, despite the fact that the pronoun is able to be correctly resolved.

and ‘she is a nurse’ would be perfectly valid since those are patterns that likely exist in the training data. From an ethical standpoint, however, this seems like an inadequate definition of fairness. Not only are female perspectives statistically vastly under-represented in literature, but such a definition would encourage stereotypes that have damaging effects.

At first glance, the second definition certainly seems to be an improvement upon the first. In the face of true ambiguity, Google’s current solution is the best we can hope to do: show both translations and rely on the user to decide which one is best for them given their context for usage.

However, true ambiguity is not the condition we should focus on for language modeling because humans do not leave pronouns as truly ambiguous. Consider, for example, Figure 1.2. Here, because we know that ‘o’ refers to a woman, there is no ambiguity in the translation of ‘o’ to ‘she’. We should focus on cases such as these when it is possible for pronouns to be disambiguated correctly, but they are disambiguated incorrectly in a biased way. Depending on the particular dataset, ‘biased’ could mean that genderless pronouns are disproportionally disambiguated to be masculine or that they are disproportionally disambiguated in a stereotypic way.

In this setting, to be truly ‘gender fair’ is to be able to correctly resolve the pronoun at an equal rate across all subgroups. Ideally, this resolution would be perfect; but if not, we would like these pronouns to be disambiguated equally well regardless of if the pronoun refers to a male or female entity, and regardless of if gender stereotypes exist in the sentence.

Currently, this resolution is difficult to do correctly because most translation systems only consider one sentence at a time. However, research is being done into systems that consider more than one sentence, henceforth referred to as context-aware translation. In theory, one might hope that a context-aware translation system could utilize the information from prior sentences in order to resolve the correctly resolve the pronoun.

As a research question, a notion of ‘fairness’ for translation systems is difficult to

formalize. While there has been significant work in the sphere of classification tasks, there has been very little on the generation tasks such as translation. As such, part of the contribution of this work will be to propose a method for evaluating the gender fairness of a particular machine translation system.

Not only is the theoretical formalization of fairness in translation difficult, but translation provides a challenging use case for implementation as well. A common response to problems caused by unbalanced data is to introduce synthetic data to provide examples of anti-stereotypic cases. However, because of the unstructured nature of text, it is difficult to perform this introduction of synthetic data well without compromising translation quality. Another common response to gender bias in NLP is to use pre-trained components that are individually ‘debiased’, for some definition of debiased, with the hope that the overall system will also be less gender-biased. However, due to a particular implementation detail that is standard use to improve translation performance, these pre-trained components cannot be ‘substituted’ into translation model architectures without substantial degradation in performance. Nevertheless, we implement these options and evaluate their performance.

Moving forward in this thesis, we will first present background material on neural machine translation (NMT), context-aware translation, fairness, and pronoun resolution, with particular attention on the current state-of-the-art model in NMT, the Transformer. Next, we survey the current approaches researchers have taken to gender bias in NLP. In particular, we explain how the context-aware Transformer might be used to resolve translations of pronouns that can be disambiguated with additional context.

We then move to practical experimentation and implementation. First, we present and analyze the dataset of interest for patterns that may lead to a gender-biased model as a means of motivating our experiments. These analyses include the distribution of examples by pronoun, patterns of pronoun occurrence in the dataset, and the frequency with which contextual examples appear in the training set.

Next, we present the models, methods, and implementation details with which we run and evaluate our experiments. In particular, we present a method for analyzing the gender-bias of a machine learning model and a parity metric with which to judge our results. We experiment with a context-aware model, balancing the data to have equal frequency of male and female pronouns, increasing the amount of examples that take advantage of context, and using standard NLP debiasing techniques. We then discuss the results of these experiments.

Finally, we conclude by summarizing our results and presenting avenues for future work.

Chapter 2

Background

In this chapter, we will first discuss developments in neural machine translation, focusing on a model known as the Transformer and its extensions in context-aware translation. We will then discuss anaphora resolution and coreference resolution, crucial tasks in discourse understanding that we will take inspiration from in development of our own task. Finally, we will discuss current methods of assessing bias in NLP.

2.1 Definitions

Formally, we represent machine translation as a mapping between sequences of strings. Let $\mathbf{x} = \{x_1, \dots, x_\ell\}$ be the sequence of tokens in our input sentence, also referred to as a source sentence. Let $\mathbf{y} = \{y_1, \dots, y_{\ell'}\}$ be any candidate for the translation for $\mathbf{x}$ in our target language. Then, we attempt to find

$$\arg \max_{\mathbf{y}} p(\mathbf{y}|\mathbf{x}; \mathbf{w})$$

Within the neural framework, we attempt to model the conditional probability of $(\mathbf{y}|\mathbf{x})$ using a neural network directly, parameterized by a set of weight vectors $\mathbf{w}$. These weight vectors are generally learned through gradient descent.

When comparing machine translation systems, the BLEU (bilingual evaluation understudy) metric is commonly used to compare translation quality. This metric, which does not require human evaluation, is quick to calculate and claims to correlate with human judgments of quality. Generally speaking, a BLEU score of 40 is nearly indistinguishable from human translation, and a BLEU of 20 is understandable. When evaluating BLEU, we refer to the true translation as the ‘reference’, and the translation generated by the model as the ‘hypothesis’.

2.2 Neural Machine Translation Models

Most machine translation models follow an encoder-decoder architecture, first introduced by Cho et al. (2014) and formalized into an end-to-end approach for translation in Sutskever, Vinyals, and Le (2014). These models have two parts: the **encoder** is a neural network responsible for encoding the meaning of the variable-length source sentence into a vector or sequence of vectors $\mathbf{c}$ such that

$$f_E(\mathbf{x}) = \mathbf{c}$$

The **decoder** is then responsible for generating a variable-length translation based on this vector, so

$$f_D(\mathbf{c}) = \mathbf{y}$$

Together, they model the conditional probability of $\mathbf{y}$ given $\mathbf{x}$:

$$p_{\mathbf{w}}(\mathbf{y}_i | \mathbf{x}_i) = f_D(f_E(\mathbf{x}_i))$$

The two components are then jointly trained to maximize the conditional log-likelihood,

$$\max_{\mathbf{w}} \frac{1}{N} \sum_{i=1}^N \log(p_{\mathbf{w}}(\mathbf{y}_i | \mathbf{x}_i)),$$

where each $(\mathbf{x}_i, \mathbf{y}_i)$ is an example of the training set of N examples and $\mathbf{w}$ represents the matrices that parameterize the model.

We note that, as standard within NLP, we use a process known as word embedding to map each token in our vocabulary from a one-hot vector to a d -dimensional word vector. For example, if we are embedding the word “student”, we might have

$$\text{“student”} \rightarrow [0, 0, 1, \dots, 0] \in \mathbb{R}^{1 \times |V|} \rightarrow [.1, .01, \dots, .2] \in \mathbb{R}^{1 \times d},$$

where $|V|$ is the size of our vocabulary.

Here, we first lookup convert our token, ‘student’, into a one-hot vector of length $|V|$. Then, we map the one-hot vector into a dense vector of length d . The exact mapping is parameterized by a weight matrix of size $|V| \times d$, which is commonly learned during training.

2.2.1 RNNs

In Cho et al. (2014), each of the encoder and decoder is represented by a recurrent neural network (RNN). RNNs rely on updating a hidden state matrix $\mathbf{h}$. On the encoder side, instead of attempting to read the entire sentence $\mathbf{x} = \{x_1, \dots, x_\ell\}$ at once, the RNN

updates the hidden state $\mathbf{h}$ one input at a time, according to

$$\mathbf{h}_t = f_{\text{RNN}}(\mathbf{h}_{t-1}, x_t)$$

for each time step t . From a probabilistic perspective, the RNN is able to model the conditional distribution

$$P(x_t | x_{t-1}) = \text{softmax}(\mathbf{w}_{\text{RNN}}^T \mathbf{h}_t),$$

where $\mathbf{h}_t$ is the hidden state from time t , and $\mathbf{w}_{\text{RNN}}$ indicates the weight vector that parameterizes a final linear layer. This means that the RNN does not rely on the sentence exhibiting Markovian properties, as previous n -gram models did. Furthermore, the update performed by f_{RNN} is generally done using the LSTM (long short-term memory) cell, which has some success in learning long-range dependencies. However, this also means that inference is expensive, because it is inherently non-parallelizable for each example.

Once the encoder has looped over the entire input sentence, we are left with a single context vector which is the hidden state $\mathbf{h}_t$ from the final time step.

This means that we can apply this to translation in an end-to-end approach (Sutskever, Vinyals, and Le, 2014) by modeling the likelihood of a translation as

$$p(\mathbf{y} | \mathbf{x}) = \prod_{t=1}^{\ell'} p(y_t | c, y_1, \dots, y_{t-1})$$

Because translation inputs and outputs are of undefined length, a recurrent model makes it easy to account for differences input and output length of the vectors. This was the de facto model that many neural machine translation systems were built on until the introduction of the Transformer model, described below, in 2017.

2.2.2 Attention

Bahdanau, Cho, and Bengio (2014) conjectured that the fixed-length vector used previously did not encode enough information about the sentence. They introduced the idea of attention, loosely inspired by the idea that humans do not translate the entire sentence at once, but rather pay attention to specific parts when translating specific words.

Rather than retaining only the hidden state from the encoder's final time step, Bahdanau, Cho, and Bengio (2014) stack the hidden states from all time steps $1, \dots, \ell$ into a matrix, $\mathbf{H}_E \in R^{\ell \times d}$. When decoding at time step t , rather than using a fixed context vector c for all time steps, they use the attention vector

$$\text{SingleVectorAttention}(\mathbf{h}_D, \mathbf{H}_E) = (\mathbf{h}_D \mathbf{H}_E^T) \mathbf{H}_E \in R^{1 \times d}$$

where $\mathbf{h}_D \in R^{1 \times d}$ is the hidden state from the decoder at time step t .

The attention function can be thought of as producing a mapping from the set of all encoder hidden states, $\mathbf{H}_E$ and a particular decoder hidden state, $\mathbf{h}_D$ to a final new vector. This final vector is a weighted sum of the individual encoder hidden states, where the weight of each hidden state corresponds to its score with the decoder hidden state, $\mathbf{h}_D$. A simple formula for the score function is the dot product, shown above, which is quick to calculate using matrix operations.

During training, it is convenient and computationally faster to calculate the attention vector for all decoder hidden states simultaneously, rather than one at a time. We can stack the decoder hidden states into a matrix, $\mathbf{H}_D \in R^{\ell' \times d}$. and denote the attention function as

$$\text{Attention}(\mathbf{H}_D, \mathbf{H}_E) = (\mathbf{H}_D \mathbf{H}_E^T) \mathbf{H}_E \in R^{\ell' \times d}$$

It is useful to abstract this notion of attention beyond $\mathbf{H}_E$ and $\mathbf{H}_D$ into any matrices $\mathbf{Q}, \mathbf{K}, \mathbf{V}$, which represent keys, queries, and values respectively.

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = (\mathbf{Q} \mathbf{K}^T) \mathbf{V}$$

Intuitively, we are ‘querying’ each entry in the query matrix against ‘keys’ to obtain a score and using those scores to create a weighted sum of the ‘values’. We note that in this particular dot-product attention mechanism, we have that $\mathbf{Q} = \mathbf{H}_D$ and $\mathbf{K} = \mathbf{V} = \mathbf{H}_E$. However, we may choose to use different key, query, and value matrices, and we may not always have that $\mathbf{K} = \mathbf{V}$. For this reason, in the remainder of this thesis, we will use this matrix notation. For additional clarity, when applicable, we will also note what matrices are being used as $\mathbf{Q}, \mathbf{K}$, and $\mathbf{V}$.

The attention mechanism yielded significant improvements with respect to translation quality and has been included in nearly every state-of-the-art model since 2014.

2.2.3 Transformer

A limitation of recurrent models was that they were slower to train and perform inference. Because the recurrent model relies on $\mathbf{h}_{t-1}$ to generate $\mathbf{h}_t$, it is inherently non-parallelizable. Vaswani et al. (2017) moved away from the recurrent model to a completely non-recurrent model called the Transformer. The five main components of the Transformer are as follows:

1. Positional Encoding: Since there is no recurrent structure, there must be a way to encode the order of the sentence. Therefore, they incorporated positional encodings, which are a mapping $\mathcal{N} \rightarrow d_m$, where d_m is the dimension of the model

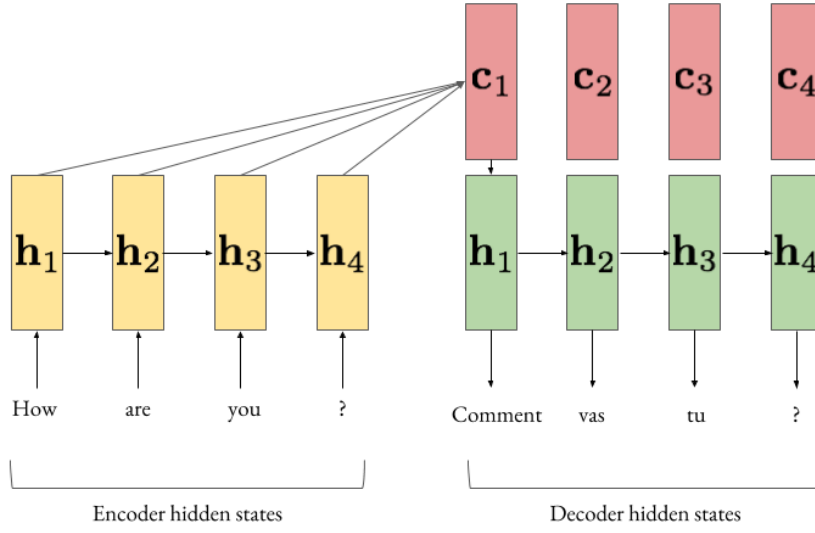


FIGURE 2.1: A visualization of the attention mechanism.

embeddings, that only depend on the positional index. These have the form

$$\text{PositionalEmbedding}(i, 2k) = \sin\left(\frac{i}{10000^{2k/d_m}}\right)$$

$$\text{PositionalEmbedding}(i, 2k + 1) = \cos\left(\frac{i}{10000^{2k/d_m}}\right)$$

where i is the positional index, and k is one particular dimension of the output vector d_m . The authors use sinusoidal functions for these encodings, motivated by the ability to extend to potentially longer sentences.

2. Scaled Dot Product Attention: Instead of just using the dot product attention mechanism described in Subsection 2.2.2, Vaswani et al. (2017) scaled the dot product by $1/\sqrt{d}$, where d is the dimension of the hidden state. The reason for this is that as the dimension grows, the attention weight may grow too large, pushing the softmax function into the region where it has a lower gradient. Thus, they scale it $1/\sqrt{d}$, to get

$$\text{Scaled Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \frac{(\mathbf{Q}\mathbf{K}^T)\mathbf{V}}{\sqrt{k}}$$

3. Self-attention: Self-attention, or intra-attention, is a mechanism to relate different positions within a single sequence. This is defined as

$$\text{Self Attention}(\mathbf{Q}) = \text{Scaled Attention}(\mathbf{Q}, \mathbf{Q}, \mathbf{Q}),$$

Vaswani et al. (2017) apply self-attention within both the encoder and decoder to calculate

$$\text{Self Attention}(\mathbf{H}_E)$$

and

$$\text{Self Attention}(\mathbf{H}_D)$$

where $\mathbf{H}_E \in R^{\ell \times d}$ is the hidden state stack from the encoder, and $\mathbf{H}_D \in R^{\ell' \times d}$ is the hidden state stack from the decoder.

4. Multi-Head Attention: Instead of just performing a single attention calculation, the authors project the hidden state matrices to separate lower-dimensional spaces using separate weights and run self-attention on each of those. Formally, we have

$$\text{Multi-Head Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{Concat}(\text{Head}_1, \text{Head}_2, \dots, \text{Head}_h)$$

where h is the total number of heads,

$$\text{Head}_i = \text{Scaled Attention}(\mathbf{Q}\mathbf{W}_q^i, \mathbf{K}\mathbf{W}_k^i, \mathbf{V}\mathbf{W}_v^i)$$

and $\mathbf{W}_q^i, \mathbf{W}_k^i, \mathbf{W}_v^i$ indicate the weight matrices for the projection in head i .

This attention mechanism is applied in both the encoder and decoder. In the encoder, we perform self-attention, so $\mathbf{Q} = \mathbf{K} = \mathbf{V} = \mathbf{H}_E$. In the decoder, we perform both self-attention, when $\mathbf{Q} = \mathbf{K} = \mathbf{V} = \mathbf{H}_D$, and attention between the encoder and decoder, so $\mathbf{Q} = \mathbf{H}_D$ and $\mathbf{K} = \mathbf{V} = \mathbf{H}_E$.

Vaswani et al. (2017) found that this multi-head attention mechanism allows the model to jointly attend to information from different representation subspaces at different positions. The intuition is that you might want to pay attention to different parts of a sentence for different purposes. For example, in the sentence, “the food eaten by the children was delicious”, if the word in question is ‘food’, you may want to attend to ‘eaten by’ to understand what happened to the food, and to ‘delicious’ to understand what the food was like.

Each of the encoder and decoder takes on the structure of a residual network (ResNet), first proposed by He et al. (2015). A residual network is composed of many layers, each of which defines a transformation. If we let x denote an input to a layer, and $f_i(x)$ the transformation defined by layer i , the residual connection will feed forward $f_i(x) + x$.

He et al. (2015) hypothesized that it would be easier to optimize the residual mapping, which is what all of the f_i jointly learn, rather than the original mapping.

On the encoder side, each residual layer within the residual network is composed of two sublayers:

- The multi-head self-attention mechanism described above,
- and a simple, two-layer feedforward network with a ReLU activation in between.

Each residual layer is repeated six times.

On the decoder side, each layer has three sub-layers instead of two.

- The first sublayer is a multi-head self-attention among the decoder embeddings, up to the time step of each embedding.
- The second sublayer applies multi-head attention between the decoder and encoder.
- The third sublayer is a simple two-layer feedforward network with a ReLU activation function in between.

Each layer is followed by the residual connection and a layer norm. The entire residual layer is repeated six times followed by a final linear layer to the size of the target vocabulary and a softmax.

In combination, these elements of the Transformer have allowed it to produce impressive results in translation, and it is considered the new state-of-the-art (Vaswani et al., 2017). Because of the lack of a recurrent structure, it is much faster to train and faster to perform inference from as well.

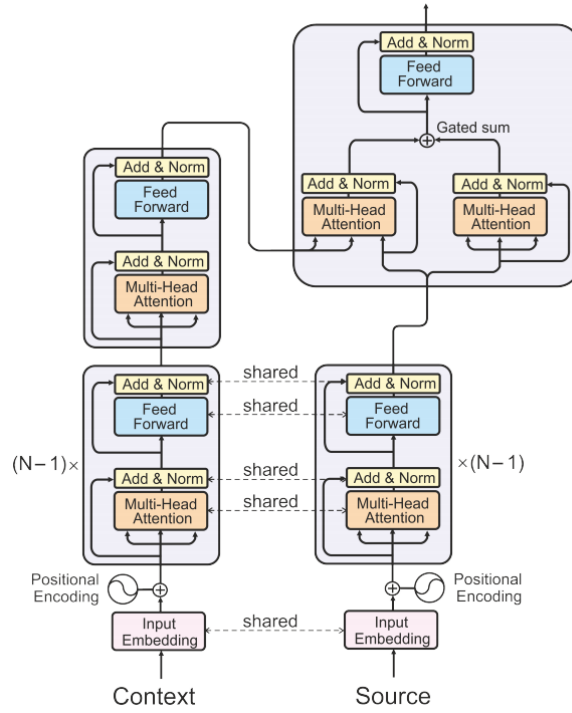


FIGURE 2.2: A context-aware extension of the Transformer encoder (Voita et al., 2018).

2.3 Context-Aware Translation

One limitation of machine translation is its limited ability in discourse, because past models only translate one sentence at a time. Recently, there has been development in context-aware translation, which attempts to increase the perceptive field of a model to improve translation quality by holding more in memory than just the current sentence. Suppose our current sentence is $\mathbf{x}_i = \{x_{i1}, x_{i2}, \dots, x_{i\ell}\}$, and the previous sentences in the document are $\{\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{i-1}\}$. Then, a context-aware translation model models the conditional probability,

$$P(\mathbf{y} | \mathbf{x}, \dots, \mathbf{x} - k)$$

for some choice of previous sentences k .

Current approaches to context-aware translation include using the prior sentence ($k = 1$), using multiple past sentences ($k > 1$), or using a future sentence. With regards to the Transformer, one way to incorporate additional context is to include a separate encoder for the context sentence, as done by Voita et al. (2018).

Voita et al. (2018) adds context only to the source sentence and not to the target sentence. Here, the context sentence is passed through a context encoder that is almost

identical to the Transformer’s source encoder in Vaswani et al. (2017). However, in the final layer, the outputs from the context encoder and the source encoder are combined using a gated sum controlled by a linear transformation followed by a softmax. In theory, the model should be robust to noisy context by learning to attend more to the context sentence only when it is relevant in decoding at a given time step.

We find the context-aware Transformer to be a possible method of incorporating the context necessary to correctly disambiguate a pronoun. There is additional evidence that shows that the context-aware Transformer is also capable of latently learning what the antecedent to a given pronoun is. More reasons for the use of a context-aware Transformer are explored in more depth in Chapter 3.

2.4 Anaphora Resolution and Coreference Resolution

Next, we present an overview of the tasks anaphora resolution and coreference resolution. These tasks are useful for understanding the performance of context-aware translation models and because we find that bias evaluation methods for these tasks can be adapted to the setting of correctly translating and disambiguating ambiguous pronouns with sufficient context. This last idea is explored in greater depth in Chapter 3.

First, we introduce the related task of pronoun resolution, also known as anaphora resolution. Given a sequence of sentences, and a pronoun present in the sequence, we want to identify which antecedent the pronoun is referring to. For example, given the sentence

$$s = \{ \text{Andy and I went to the store, and he left early.} \}$$

and a pronoun of interest,

$$s_9 = \text{he},$$

we want to identify the antecedent of the pronoun in the sentence. In this case, we would want to return

$$s_1 = \text{Andy}$$

Coreference resolution is a very similar problem to pronoun resolution but structures it as a clustering problem. Rather than finding an antecedent for some pronoun, we want to identify each coreference chain in the text, where a ‘coreference chain’ is a set of words or word phrases refer to the same object. Note that coreference is an undirected relation, whereas pronoun resolution is a directed relation from the pronoun to its antecedent. For example, given the

$$s = \{ \text{"Andy and I went to the store, and he left early"} \},$$

the goal is to identify the referents

$$\{s_1, s_9\} = \{"Andy", "he"\}$$

In the example above, it is clear that it is $\{"Andy", "he"\}$ is the coreference chain, rather than $\{"Andy", "I"\}$, however, sometimes the coreference chain in a sentence may be ambiguous without prior world knowledge. For example, consider the sentence:

The city councilmen refused the demonstrators a permit because they [feared/advocated] violence.

Knowing if ‘they’ should be a coreferent with ‘councilmen’ or ‘demonstrators’ requires world knowledge about the relationship of people with violence. Paired questions such as this form the basis of one of the most difficult coreference resolution tasks known as the Winograd schema challenge. Each entry presents a pair of sentences that only differ in one or two words, and that contains an ambiguity that requires the use of world knowledge for its resolution. Recently, however, there has been research demonstrating that coreference resolution systems exhibit gender bias. Zhao et al. (2018a) and Rudinger et al. (2018) independently curated Winograd-style challenges to test this assumption. Both of them found that state-of-the-art coreference systems exhibited gender biases, particular with regard to occupational stereotypes. They used sentences such as

[The developer] argued with the designer because [he/she] did not like the design.

and found that coreference systems disproportionately predicted pro-stereotypical results.

Separately, the Gendered Ambiguous Pronoun (GAP) (Webster et al., 2018) challenge has also been released by Google, built to address the lack of gender-balanced corpora of sufficiently large size in coreference resolution. In particular, they found that existing corpora were highly gender-biased and that present coreference systems favored masculine entities.

We find the WinoBias task proposed by Zhao et al. (2018a) to be adaptable to our setting, an idea which we explore in more depth in Chapter 3. We do not try to explicitly model anaphora resolution, but rather find these tasks to be relevant to our problem.

Anaphora resolution is also helpful in understanding what patterns of discourse the context-aware Transformer is capable of learning. In particular, the most promising aspect of the context-aware Transformer is that it has been shown to latently learn anaphora resolution. This suggests that perhaps the Transformer will be able to use the context in previous sentences to correctly disambiguate pronouns, an idea we will explore in further detail in Chapter 3.

2.5 Bias and Fairness

Recent research in bias and fairness in machine learning has mostly been limited to the topic of classification. Here, we focus not on the statistical ‘bias’ of a particular estimator, but rather ‘bias’ in the sense of providing ‘unfair’ predictions across different subgroups.

Define our population as X with distribution D , and suppose we have a protected subset $S \subset X$. Then we can define the bias of a classifier h with respect to S is

$$\text{bias}_h(X, S, D) = \Pr_D[h(x) = 1 | x \in S^C] - \Pr_D[h(x) = 1 | x \in S]$$

This lends itself to the idea of ‘statistical parity’: that the outcome of a classifier should be equal across groups defined by the protected attributes, or at least have only bias up to some value. A classifier h is said to have statistical parity up to bias ϵ if

$$|\text{bias}_h(X, S, D)| \leq \epsilon$$

Recall our motivation from Chapter 1 to focus on cases when it is possible for pronouns to be disambiguated correctly, but they are disambiguated at different rates across subgroups. Given this goal of equality over subgroups, we find that statistical parity is a useful condition to achieve fairness, and provide a ‘parity’ metric for doing so in Chapter 5.

Chapter 3

Related Work

In this section, we examine more closely the various approaches that researchers have taken to analyzing bias within NLP, with particular attention to ones that are applicable to the translation of pronouns. We revisit the work of Voita et al. (2018) in greater depth, and discuss why it appears promising for correctly disambiguating pronouns in translation.

3.1 Bias and Fairness

Within natural language processing, gender bias has been examined with respect to a variety of tasks. Bolukbasi et al. (2016) first found that word embeddings trained on Google News exhibited strong gender biases, citing the result ‘Man is to Programmer as Woman is to Homemaker’, and also presented a method for debiasing the embeddings, known as ‘Hard-Debias’. Zhao et al. (2018b) presented an alternative method to debiasing word embeddings, known as ‘Gender Neutral - Glove’. It is not clear, however, if debiased word embeddings alone are sufficient to debias the model which uses them, and whether or not using these debiased word embeddings will decrease the overall quality of the model. For instance, Park, Shin, and Fung (2018) examined gender bias in abusive language detection datasets, and experimented with debiased word embeddings and adding synthetic records to the training data. They found that debiased word embeddings alone were insufficient to significantly reduce bias. Dixon et al. (2018) also applied a data augmentation method to toxic language detection. They found that certain clearly non-toxic statements, such as ‘*I am a gay man*’ were given unreasonably high toxicity scores. As such, they added additional non-toxic training examples that included the identity words, for example ‘gay’, that had previously exhibited the most bias. The strategy of adding synthetic data was also found by Zhao et al. (2018a) to be most successful in combination with debiased word embeddings to improve performance on the WinoBias task.

Within coreference resolution, the WinoBias and Winogender datasets provide a task for coreference resolution that requires being able to identify the correct coreferent both

in the presence of and despite gender stereotypes (Rudinger et al., 2018; Zhao et al., 2018a). For example, a coreference system that would perform well would need to identify that in the sentence,

[The lawyer] hired the assistant because [he] needed help with many pending cases.

lawyer and *he* form the correct coreference chain, whereas in the sentence

The lawyer hired [the assistant] because [she] was unemployed.

the correct coreference chain is *assistant* and *she*. Finally, the Gendered Ambiguous Pronouns (GAP) challenge provides a gender-fair dataset of ambiguous pronouns (Webster et al., 2018). These examples do not ‘switch’ as the WinoBias examples do, but rather comprise as a whole a gender-balanced dataset.

We will extend the coreference work to translation by using the WinoBias schema as inspiration for a gender-bias testing framework for translation. Just as WinoBias and WinoGender require identify the correct coreferent both in the presence of and despite gender stereotypes, our schema requires translation systems to identify the correct pronoun, via the heuristic of higher likelihood, both in the presence of and despite gender stereotypes. Our schema will be comprised of examples of the form:

[Context]: Adamın / Kadının mesleği nedir? [Source]: O bir doktor. → [Target] He / She is a doctor.

This is also similar to the DiscoMT cross-lingual pronoun prediction task, which also attempts to test accurate translations of ambiguous pronouns through examples of the following form:

[Source] It's an idiotic debate. It has to stop. → [ce/elle/il]'st un debat idiot qui doit stopper.

However, we diverge from DiscoMT as our examples require the use of an external context sentence to disambiguate, whereas the examples in DiscoMT are able to be resolved through the source sentence alone. Furthermore, our task is explicitly designed to test the fairness in the resolution of gendered personal pronouns, whereas DiscoMT focuses on overall accuracy over many different pronouns.

3.2 Translation

3.2.1 Bias Quantification and Model Evaluation

To quantify the bias of a translation system, some have chosen to report BLEU scores of each subgroup as a proxy for the ‘fairness’ of the system overall. For example, Elaraby et

al. (2018) reports BLEU scores for both a validation set of male speakers and a validation set of female speakers, with the goal of making them more equitable. This is a similar approach taken as Voita et al. (2018), who report BLEU scores on sentences with an ambiguous pronoun that has either a feminine or masculine antecedent.

An alternative is to evaluate the model for text classification rather than text generation. Instead of directly evaluating the generated translation, we can use the model to score the likelihood of multiple contrastive translations which only differ in some protected attribute. For instance, we can compare the contrastive translations of “he is a doctor” and “she is a doctor” and look at the likelihood of each translation given the model. This has been used in Bawden et al., 2017 as the evaluation scheme to evaluate discourse phenomena, and offers more granularity than simply comparing BLEU scores.

We will use both methods. We favor the method of creating an evaluative test set for understanding the behavior of our model at a more fine-grained level. Nevertheless, we believe that reporting BLEU is also useful, and more accurately describes the performance of our model in a translation setting.

3.2.2 Debiasing Methods

To address the problem of reducing bias, Elaraby et al. (2018) and Vanmassenhove, Hardmeier, and Way (2018) chose to directly add a binary feature to the beginning of each sentence to represent if the speaker is male or female. Prabhumoye et al. (2018) applied a style transfer approach, where ‘men’ and ‘women’ are modeled as two different domains. Both Elaraby et al. (2018) and Vanmassenhove, Hardmeier, and Way (2018) address primarily cases where the bias arises due to a difference in the style of the speaker. However, this not address gender biases that arise due to ambiguity in translating from one language to another: for example, from a language like Turkish, with very few gender markings, to French or Russian, which has many more gender markings. To resolve the ambiguity well requires a model that is capable of handling discourse phenomena. Finally, these techniques are only applicable when each example can be labeled as ‘male’ or ‘female’. They do not extend to the majority of written language, where masculine and feminine referents and pronouns may be intermingled.

Another common approach is to use the debiased word embeddings found by Bolukbasi et al. (2016) within a larger machine learning model as opposed to learning the embeddings during training. This was the approach taken by Font and Costa-Jussà (2019) in debiasing Spanish - English translation.

However, within other NLP applications, using debiased word embeddings alone has been shown to be insufficient for reducing the bias of the overall model (Park, Shin,

and Fung, 2018). Furthermore, within translation specifically, words are frequently encoded as subwords using a byte-part encoding method, while most pre-‘debiased’ embeddings are word-level embeddings. Finally, recent translation systems share embeddings between the source and target languages. It is unclear how these ‘bilingual’ word embeddings should be debiased. For these reasons, we will not focus on word embeddings alone.

3.3 The Transformer: A Potential Discourse Solution

Within context-aware translation, the Transformer has gained popularity for implicitly learning anaphora resolution (Webster et al., 2018). They found that certain self-attention layers appeared to show stronger attention weights between coreferent words, and tested that heuristic on the GAP dataset. Similarly, Voita et al. (2018) found that a context-aware Transformer model was also able to learn a latent representation of anaphora resolution, which yielded better translation quality when an ambiguous pronoun was present in the source language.

First, Voita et al. (2018) observe that although the baseline Transformer model performed well on the test set overall (29.46 BLEU), they found that it performs much more poorly on the subgroup of cases where an ambiguous pronoun had a feminine antecedent as opposed to a masculine antecedent (21.8 BLEU vs 26.9 BLEU). When comparing the context-aware model to the baseline model, they notice that gains in BLEU are particularly significant in cases where the pronoun referred to a feminine antecedent compared to a masculine object (+4.8 BLEU vs +.3 BLEU), and suggest this is because the context-aware model is able to choose the correct pronoun based on the added context. To check this idea, they ran an experiment where they used the attention heuristic to test coreference resolution and compared that with CoreNLP, a leading system for coreference resolution. The attention heuristic agreed with CoreNLP 70% of the time, suggesting that the context-aware model was able to latently learn what it was referring to.

In contrast to Voita et al. (2018), Bawden et al. (2017) found that adding context to a recurrent model did not yield a latent anaphora representation. The combination of these results suggests that it might be the architecture of the Transformer itself, rather than simply adding context, that allow it to represent anaphora resolution across different sentences.

Based on these results, we will use the context-aware model of the Transformer to try to correctly disambiguate pronouns. We will compare these results to other methods of debiasing, including data balancing and using debiased word embeddings.

Chapter 4

Data Analysis

In this section, we will examine our training data to hypothesize what might cause problems in our trained models and to motivate our experiments in Chapter 5.

4.1 Distribution of Training Examples

For the remainder of this thesis, we will focus on Turkish to English translation. Turkish is of particular interest because it is one of the few languages that has a genderless third-person singular personal pronoun. This is the word ‘o’, which can be translated to ‘he’, ‘she’, ‘it’ or ‘that’, depending on context. In contrast, English has two gendered third person singular personal pronouns, ‘he’ and ‘she’. Thus, the word ‘o’ presents a source of ambiguity when translated into English. Therefore, we can examine the way a machine learning system resolves this ambiguity as a proxy for the gender biases it may have learned.

For training data, we use the OpenSubtitles 2018 corpus (Lison, Tiedemann, and Kouylekov, 2018) for Turkish and English. These are subtitles for popular movies and TV shows that were translated by volunteers and then made open source and aligned. The corpus consists of all movies and TV shows that have both Turkish and English subtitles. These movies range from about 1902 to the present, although there is substantially more training data available from the last ten to twenty years. We choose this particular corpus for the relatively large amount of Turkish training data compared to other tasks, such as WMT (Conference for Machine Translation). OpenSubtitles provides 45 million aligned sentence pairs, as opposed to about 200,000 aligned sentence pairs provided by WMT. However, we do note that the data quality is lower than other training data such as WMT Turkish data.

For the following data analysis, we randomly select 2 million examples, henceforth referred to as the ‘analysis dataset’. From the analysis dataset, we filter for the examples that contain the word ‘o’ to find sentences that have an ambiguous pronoun. As a proxy for the translation of ‘o’, we count the number of times the target sentence contains ‘he’, ‘she’, or ‘it’. The results are shown below.

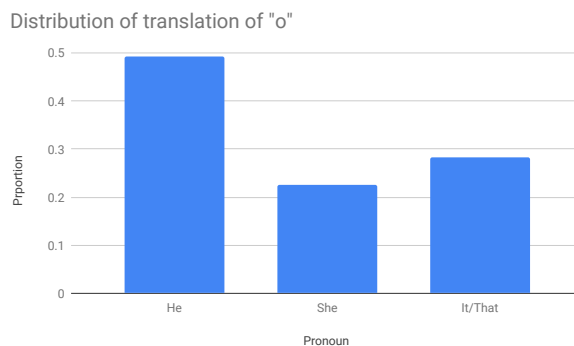


FIGURE 4.1: Approximately 50% of the time, 'o' was translated into 'he', 25% of the time, it was translated into 'she', and 25% into 'it' or 'that'.

This result is not surprising: there is a plethora of evidence that shows that female representation is significantly less than male representation in literature, both in terms of characters and authors. In our training set, as well, there is an imbalance between the number of examples that have a female subject compared to the number of examples that have a male subject. This is likely to cause problems because our model may learn to predict 'he' more often since it is the most frequent class.

4.2 Frequencies of Pronouns Given Occupation

An example often used to judge the bias of an NLP system is if it treats occupations in a stereotypic way. Before doing so, we should look at if the training data also portrays occupations in a stereotypic way. We acknowledge that gender representation in film media, especially popular Hollywood TV and movies, may be exacerbated or otherwise uncharacteristic of gender representation in other media. However, we believe that occupations are still an interesting analysis worth performing.

To perform this analysis, for all sentences that mention a given occupation, we count the number of those sentences that use a male pronoun and the number that use a female pronoun. We would expect the counts to be heavily biased towards the conventional stereotype for that occupation if it is indeed true that the training data reflects stereotypes in society.

First, we use the list of 'stereotypically female occupations' and 'stereotypically male occupations' defined by Zhao et al. (2018a). These occupations have the largest discrepancy in gender representation, according to the Bureau of Labor Statistics. We filter our analysis dataset for examples that mention these occupations into two datasets that contain only mentions of 'female occupations' and 'male occupations' respectively. Over each of these datasets, we count the number of examples that mention a male pronoun

Dataset	Prop. of Male Pronouns	Prop. of Female Pronouns
Full Dataset	0.65	0.35
Male Occupations	0.75	0.25
Female Occupations	0.63	0.37

TABLE 4.1: The normalized fraction of examples that had a male pronoun compared to examples that had a female pronoun. Found on (a) the entire dataset (b) the subset of examples which contain only mentions of stereotypically male occupations, and (c) the subset of examples that contain only mentions of stereotypically female occupations.

and the number of examples that mention a female pronoun. We report the normalized ratio between the two in Table 4.1.

We observe that there are significantly more male pronouns mentioned in the dataset of stereotypically male occupations, which is unsurprising. Interestingly, however, the proportion of male pronouns remains high even in the subcategory of stereotypically female occupations. This suggests that it will be difficult for the model to learn to generate ‘she’, even in the presence of a female stereotype.

We next take a closer look at the sentences in each of the sub-datasets and split them according to occupation. Then, for each occupation, we count the number of examples that mention that occupation and have a male pronoun, and the number of examples that mention that occupation and have a female pronoun. The results are shown in Figure 4.2.

Unsurprisingly, we find that for the vast majority of stereotypically male occupations, there are significantly more male pronouns mentioned than female pronouns. This suggests that the data does exhibit gender stereotypes with regards to stereotypically male occupations. The one possible exception to this is ‘cook’, which occurs in an almost equal ratio.

Next, we examine the frequencies of male and female pronouns with respect to stereotypically female occupations:

This result is somewhat surprising. We notice that for most of the stereotypically female occupations, the number of male pronouns mentioned far outnumbers the number of female pronouns mentioned, with the exception of ‘nurse’, which slightly favors female pronouns. This suggests that our analysis dataset does not have ‘stereotypically female’ occupations, and in fact considers nearly every occupation ‘stereotypically male’.

As such, we would expect a baseline model trained on this dataset to predict ‘he’ for all occupations, not just ones we might consider stereotypically female. This suggests that a more useful way of analyzing the pronoun translation accuracy will be to divide

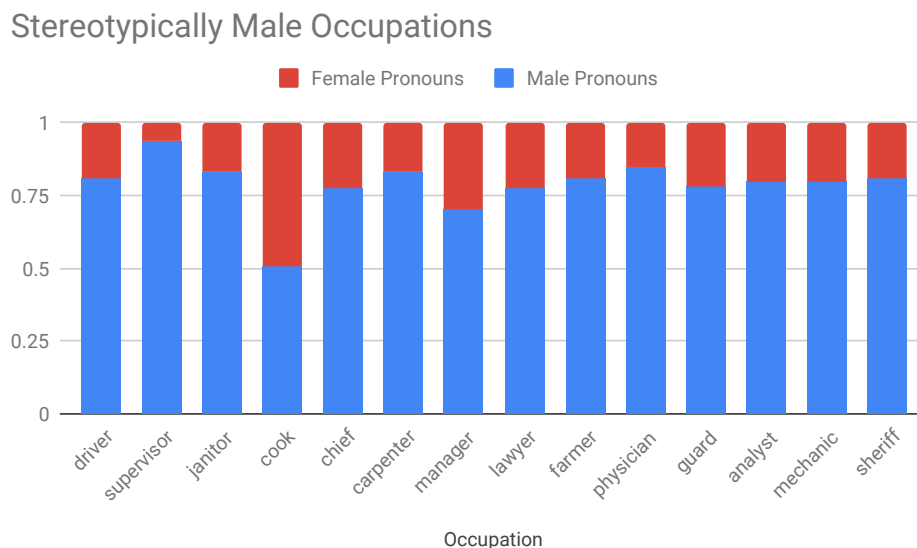


FIGURE 4.2: The normalized ratio of the number of male to female pronouns mentioned with each twenty stereotypically male occupations. Occupations that were mentioned with a pronoun less than five times are not displayed.

the test set by the gender identity of the pronoun rather than the presence or absence of a stereotype.

4.3 Can Context Help Disambiguate Pronouns?

Recall that we hope to use a context-aware model to disambiguate pronouns in a manner that performs equally well across different subgroups of the data. However, to learn patterns that rely on context requires a sufficient frequency of training cases that exhibit those patterns. Specifically, we are interested in the frequency of training cases that exhibit a pronoun in the source sentence and its antecedent in the context sentence. For example, the training case

[Target Context] John, it's about Mom. [Target] She's had a heart attack.

is one such example, because the pronoun *she* has an antecedent, *Mom*, in the context sentence. Note that our model does not utilize the target context, but we provide it for the sake of example rather than the Turkish context sentences.

Because of a lack of a high-quality coreference annotator for Turkish, we instead use

Stereotypically Female Occupations

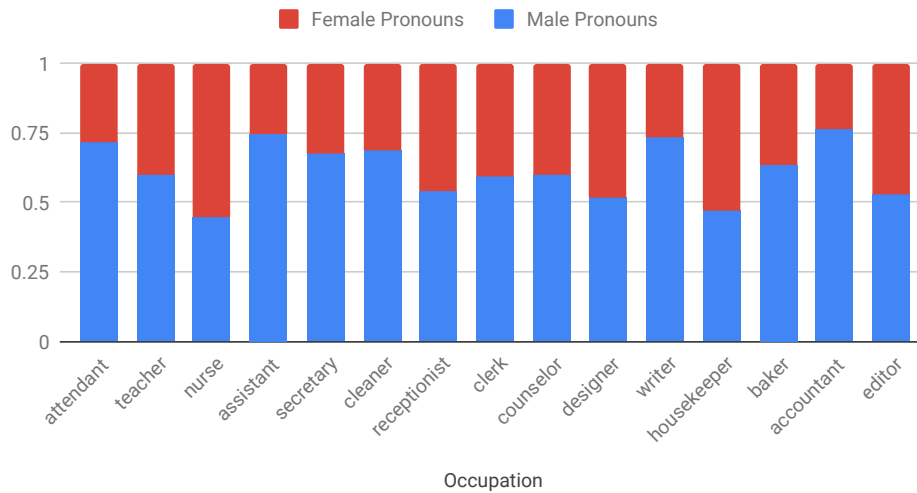


FIGURE 4.3: The normalized ratio of the number of male to female pronouns mentioned with each twenty stereotypically female occupations. Occupations that were mentioned with a pronoun less than five times are not displayed.

Stanford CoreNLP’s coreference annotator on the target sentence and its context (Manning et al., 2014). The Stanford CoreNLP coreference annotator is a well-established system that is able to detect coreference chains, and consequently, find cases where there is a pronoun in the source sentence and an antecedent in the context sentence.

Using this approach, we find that only 6.4% of training examples fall into this category. This could make it difficult for the context-aware model to learn inter-sentence phenomena from a simple random sample of the data, because of the rarity with which it occurs. This suggests that we should try to increase the percentage of examples that exhibit coreference to see if the performance of the context-aware Transformer improves. Note that we do not limit our training examples to only those which exhibit coreference, because we would not have enough examples.

Next, we report the distribution of pronouns in all cases where a coreference chain occurs in Figure 4.4. We observe that in nearly 60% of cases, the pronoun in a coreference chain is ‘it’. ‘He’ also outnumbers ‘she’, at a ratio of about 2:1, but each pronoun only makes up about 7% and 3.5% of cases respectively. Here we note a point of departure of our work from Voita et al. (2018), which looks at modeling the ambiguity in ‘it’ from English to Russian since the Russian ‘it’ is gendered but the English ‘it’ is not. However, whereas ‘it’ comprises the vast majority of pronouns in a coreference chain, ‘he’ and ‘she’ are quite rare. This could potentially make it even more difficult for our model to learn inter-sentence dependencies for the personal pronouns ‘he’ and ‘she’, since they

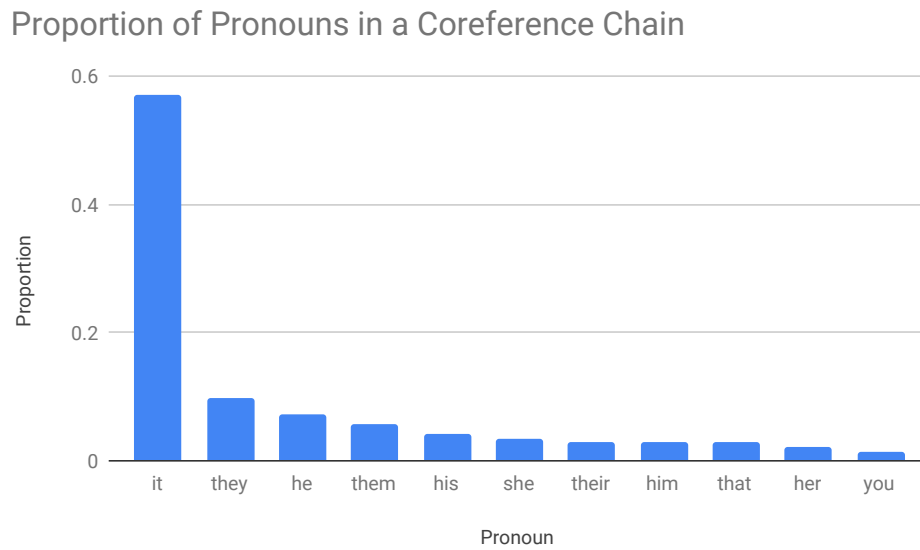


FIGURE 4.4: The distribution of pronoun frequencies in training examples that show anaphora resolution between the context sentence and the source sentence.

occur with relatively lower frequencies.

Within these cases, we filter for examples where the Turkish sentence includes the word ‘o’. We should be careful to note that this does not necessarily mean that the pronoun in the referent chain was translated to ‘o’, because ‘o’ may have occurred in the sentence ambiguously and not appeared within the referent chain. Nevertheless, we then show the distribution of the most common pronouns in Figure 4.5.

We observe that, as expected, the majority of pronouns found are a potential translation of ‘o’: ‘it’, ‘he’, and ‘she’. However, ‘it’ still comprises the plurality of pronouns found in a coreference chain. At first glance, this might seem contradictory to the results found in Figure 4.1, where ‘he’ was the majority pronoun mentioned. However, recall that Figure 4.5 only shows pronouns involved in a coreference chain between the previous sentence and the current sentence, whereas Figure 4.1 simply counts pronoun references. It seems quite likely that viewers are expected to remember the humans involved in a given discourse for much longer than a single sentence, which is why there are fewer coreferent chains for personal pronouns when the context is a single sentence away. In contrast, because ‘it’ can refer to so many different referents, viewers are not expected to hold the antecedent in memory for nearly as long, and so there are more coreferent chains for ‘it’.

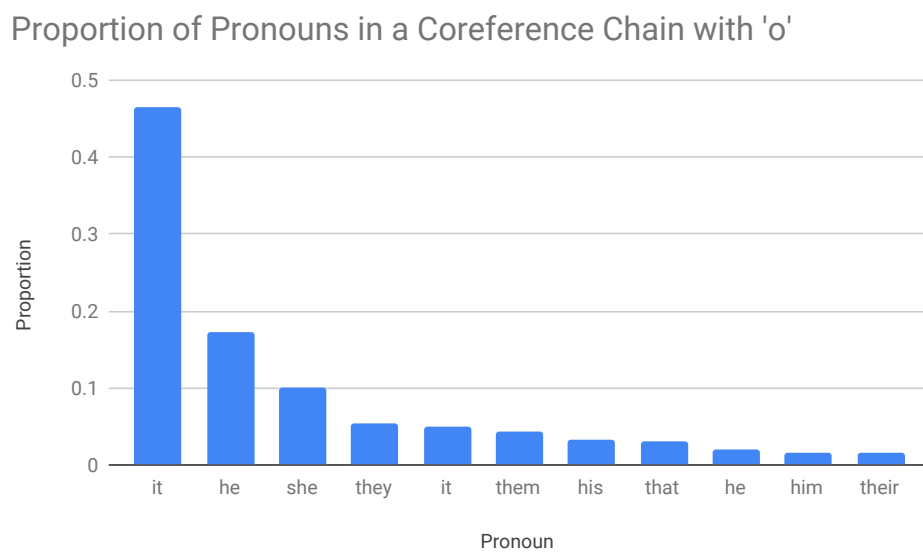


FIGURE 4.5: The distribution of pronoun frequencies in training examples that show anaphora resolution between the context sentence and the source sentence and the Turkish sentence contains the word 'o'.

Chapter 5

Methods

In this section, we describe the experimental methodology: specifically, the models, evaluation methods, and experiments run. Based on the analysis in the previous section, we present four experiments: using a context-aware Transformer, modifying the training data to be more ‘gender-balanced’, modifying the training data to more examples that exhibit coreference, and using debiased word embeddings.

5.1 Data

For training data, we use the OpenSubtitles 2018 corpus (Lison, Tiedemann, and Kouylekov, 2018) for Turkish and English. As outlined in the previous chapter, these are subtitles for popular movies and TV shows that were translated by volunteers and then made open source and aligned. Again, this dataset was chosen for the relatively large amount of Turkish training data compared to other tasks, although we acknowledge that the data quality is lower than other training data corpora.

We first split the set of movies into training, validation, and test sets with a 60% / 20% / 20% split. For training, from the set of training movies, we randomly select 2 million sentences, out of a total of approximately 9 million sentences. The validation and test sets are each comprised of 10,000 sentences, selected from movies which were not in the training set.

Movie subtitles can be particularly noisy to align compared to translations of written articles, so Lison, Tiedemann, and Kouylekov (2018) provide alignment overlap values between sentence pairs, used to judge how likely the two sentences are to correctly align. These values range from 0 to 1. As done in Voita et al. (2018), we select sentence pairs that have an alignment overlap of at least .9 to reduce noise. When selecting context sentences, we take the previous sentence if its timestamp is no more than 7 seconds prior to the current sentence. This is to avoid noise from cases where there might be a long pause in dialogue and the ‘previous sentence’ is in fact from an entirely different discourse and is also done in preprocessing by Voita et al. (2018).

Sentences were pre-processed to be lowercase, and superfluous leading and trailing punctuation (such as dashes or ellipses) were stripped. Unless otherwise noted, sentences were encoded with byte-pair encoding, a common strategy in machine translation. Byte-pair encoding is a method of segmenting sentences into ‘subwords’, rather than full words, and has been shown to substantially improve performance in translation. The particular subwords are chosen based on the frequency of subwords in the dataset. For example, an English sentence encoded using byte-pair encoding might look like:

"This is anarchism" $\rightarrow$ ‘_this’, ‘_is’, ‘_an’, ‘arch’, ‘ism’

We used the BPEMb library, which is a collection of pre-trained byte-pair encoding segmentation methods and encoding embeddings (Heinzerling and Strube, 2018). We only used their segmentation and did not use the actual subword embeddings.

The only time we do not use byte-pair encoding is when experimenting with debiased word embeddings when we use full word encoding so as to be able to use pre-trained word embeddings. It is necessary to use full word encoding here because the ‘debiasing’ methods presented by Bolukbasi et al. (2016) and Zhao et al. (2018b) rely on ‘definitional pairs’ of words, e.g. man/woman, in order to perform the debiasing.

Finally, the token ‘<s>’ is prepended to the beginning of each target sentence, and the ‘</s>’ is appended to the end of each target sentence to indicate the beginning and end of a sentence respectively. These are necessary if we hope to use our model for text generation. As done by Voita et al. (2018), we also use the preceding sentence as context for each source sentence and precede each context sentence with a ‘<boc>’ token to differentiate it from a source sentence.

5.2 Models

As a baseline, we will use the Transformer (Vaswani et al., 2017), as detailed in Chapter 2 with standard hyperparameters. We will compare it to the context-aware model proposed by Voita et al. (2018), the architecture of which was described in Chapter 2 and implications with regard to pronoun resolution described in Chapter 3. All models are implemented in PyTorch, a machine learning framework in Python.

Unless otherwise specified, we also employ the common strategy of sharing embedding weights between the source and target language. Sharing weights has been shown to improve performance while reducing the size of the model to be trained (Press and Wolf, 2016). The only time we do not use the shared weight strategy is to experiment with debiased word embeddings.

Parameter	Value	Description
Num_layers	6	Number of residual layers in the Transformer
d_{model}	512	Dimension of non-feed-forward layers
d_{ff}	2048	Dimension of feed-forward layers
h	8	Number of ‘heads’ in Multi-Headed Attention
Dropout	0.1	Regularization parameter
Tokenization Method	BPE	How the sentences are tokenized
Source Vocab Size	10000	Hyperparameter for BPE
Target Vocab Size	10000	Hyperparameter for BPE
Num_epochs	50 - 200	Number of epochs trained
Batch size	3000 - 5000	Number of observations processed at once

TABLE 5.1: Model hyperparameters for both the baseline Transformer model and context-aware models.

During inference, because the number of potential sentences is $O(|\text{vocab}|^{\text{len}})$, we follow the standard practice of performing beam search with a beam size of 5. We use a maximum length of 30. Beam search is a heuristic search algorithm where at each time step, we maintain the top five hypotheses for translation that have been generated thus far. We stop expanding a hypothesis when the token ‘</s>’ is output since that indicates the end of a sentence. After running beam search for a fixed number of steps, in this case, 30, we report the top hypothesis.

The hyperparameters used to train the model are standard Transformer hyperparameters used by Vaswani et al. (2017) and Voita et al. (2018). There are six residual layers in each of the encoder and decoder. Within the Transformer, most layers have a dimension of 512, with the exception of feed-forward layers which have a dimension of 2048. We use a value of 8 for the number of heads in multi-headed attention, as outlined in Chapter 2. As is commonly done for regularization of neural networks, we use a dropout probability of 0.1.

All models were trained using NVIDIA V100 Tensor Core Graphic Processing Units (GPUs). Different models required settings to converge. In general, because the context-aware models processed more data as input, they were much slower to train and required more computing power. Non-contextual models with BPE were able to trained in 100 - 200 epochs on a single GPU, with a batch size of 3000. Models with context required about 50 epochs on 4 GPUs, with a batch size of 5000.

5.3 Experiments

Recall that we are interested in seeing if adding context to a Transformer will improve its performance in translating ambiguous pronouns. As such, **Experiment 1** will compare the baseline Transformer with a context-aware Transformer trained on a simple random

sample of 2 million sentences from the training set. Is simply changing the architecture sufficient to improve performance in pronoun translation?

However, by the analysis in the previous chapter, we know that examples that exhibit coreference are quite rare, comprising only 6.5% of the training data. If we had more examples with coreference, would the performance of the context-aware Transformer improve? As such, in **Experiment 2**, we will sample more examples that exhibit coreference to see if that improves the performance of the context-aware Transformer relative to the baseline Transformer. Our modified dataset has 12.1% examples that exhibit coreference out of 2 million, which is about twice as many examples as the baseline dataset.

We also observed in the previous chapter that ‘o’ is translated to ‘he’ twice as often as ‘she’, as seen in Figure 4.1. What would happen if these frequencies were equal, in a perfect world with perfectly balanced representation? This suggests a natural approach to select training data with more balanced frequencies between ‘he’ examples and ‘she’ examples, which we do in **Experiment 3**. This strategy is also a commonly used ‘debiasing method’, and provides a good point of comparison to our other experiments. We take a fairly naive approach, which is to simply ‘gender-balance’ the training data, such that the fraction of examples with a female pronoun is equal to the number of examples with a male pronoun. After balancing the data, we confirm that the normalized fraction of examples with a male pronoun as compared to a female pronoun are equal, as seen in Table 5.2.

Finally, a common strategy for debiasing tasks is to use debiased word embeddings, so we compare the performance of using debiased word embeddings in **Experiment 4**. We use the ‘Hard-Debias’ algorithm proposed by Bolukbasi et al. (2016), but note that other researchers have found comparable results from using ‘Hard-Debias’ and ‘Gender-Neutral Glove’ embeddings in a translation setting (Font and Costa-Jussà, 2019). However, we discussed in Chapter 3 how it is often unclear if simply debiasing the word embeddings is sufficient to debias the resulting task in which they are used. Furthermore, if we are to use debiased word embeddings, we cannot use byte-pair encoding, which we described earlier as a common technique to improve translation quality. Therefore, we will use debiased word embeddings on both the baseline Transformer compare its performance to models that do not use debiased word embeddings. Will the models with debiased word embeddings be able to achieve the same translation quality as models that do not? Will using the debiased word embeddings improve performance on the pronoun translation task?

Dataset	Prop. Male Pronouns	Prop. Female Pronouns
Random Training Dataset	0.66	0.34
Balanced Training Dataset	0.50	0.50

TABLE 5.2: Comparison of the normalized fraction of examples that had a male pronoun compared to examples that had a female pronoun in both the baseline dataset and the ‘gender-balanced’ training dataset.

5.4 Evaluation Methods

5.4.1 BLEU

As a means of measuring general translation performance, for each experiment, we report BLEU on a test set of 10,000 sentences from the OpenSubtitles corpus. BLEU is calculated using the `multi-bleu.perl` script, which is standard practice within the machine translation research community.

5.4.2 Pronoun Accuracy

As a means of measuring the gender bias of a particular model, we create a test set consisting of ‘pro-stereotype’ and ‘anti-stereotype’ examples in the style of Winobias to judge if ambiguous pronouns can be resolved correctly in the presence of gender stereotypes (Zhao et al., 2018a). For example, the source-target sentence pair with context,

[Context]: Adamın mesleği nedir? [Source]: O bir doktor. → [Target] He is a doctor.

is a pro-stereotypical translation, whereas

[Context]: Kadının mesleği nedir? [Source]: O bir doktor → [Target] She is a doctor.

is an ‘anti-stereotypical’ translation. Stereotypical occupations are taken from the Winobias list of gendered occupations. (Zhao et al., 2018a).

All examples follow the template:

[Context] Adamın / Kadının mesleği nedir? [Source] O bir {occupation}.

→ [Target] He / she is a {occupation}.

The occupations were chosen from the list of occupations with the largest gender gaps, as reported by the Bureau of Labor Statistics. We take the top forty occupations and create a male and female subject example for each. In total, our test set has eighty examples.

We are now able to apply our translation model to a classification task. For each sentence, the task is to correctly choose the correct pronoun, as measured by the likelihood of each pronoun as given by the model. For example, consider the following:

[Context]: *Adamın mesleği nedir?* [Source]: *O bir doktor.* → [Target] *He is a doctor.*

Since the correct translation for *o* in this example is *he*, if the model scores *he* with a higher probability than *she*, then we count this example as correct.

In Chapter 4, we found that sentences with female pronouns were vastly under-represented compared to male pronouns. Because our baseline model predicts ‘he’ for every example, we find it useful to divide our test set into examples that have a female pronoun and examples that have a male pronoun. Then, our goal can be to achieve statistical parity between the class-conditional accuracies. To quantify this measure, we report the parity score on this classification task as previewed in Chapter 2. Inspired by the F1 score, we can measure this parity by taking the harmonic mean between the class conditional accuracies. We refer to this quantity as the **parity** of a classifier. This measure is then given by

$$\begin{aligned} \text{Parity} &= \left(\frac{\text{Sensitivity}^{-1} + \text{Specificity}^{-1}}{2} \right)^{-1} \\ &= \left(\frac{\text{Accuracy on Male Pronouns}^{-1} + \text{Accuracy on Female Pronouns}^{-1}}{2} \right)^{-1} \end{aligned}$$

A perfectly unbiased translation model, at least in the context of disambiguating pronouns correctly, would have a parity score of 1. A baseline model that predicts the same pronoun for every example will achieve a parity score of 0. A model that is able to correctly classify every example will achieve a parity score of 1. A model that is able to correctly switch the pronoun from ‘he’ to ‘she’ based on the context sentence will achieve a parity score that is greater than 0.5.

We note that this metric differs from the F1 score because the F1 score does not take true negatives into account. That is, if we were to only calculate the F1 score on by considering female pronouns as a ‘true positive’, then we would neglect the goal of also classifying male pronouns correctly. Therefore, we find a balanced metric such as the parity metric defined above to be necessary.

Chapter 6

Results and Discussion

In this section, we present the results of our experiments and provide discussion and analysis as to why they might have occurred.

6.1 Results

Table 6.1 shows our results from experimenting with model architecture and modifying the training set. Table 6.2 shows our results from experimenting with using debiased word embeddings.

In summary, we find that although the baseline Transformer model has a high translation quality, it predicts ‘he’ for every example in the pronoun accuracy task, as we suspected based on the analysis performed in Chapter 4. We find that gender-balancing the data improves the performance the most and that the best parity score is achieved with both gender-balanced data and a context-aware Transformer. Neither adding extra examples that exhibit coreference nor using debiased word embeddings significantly improve performance on the pronoun accuracy task.

We will begin by discussing the results on the training set that is comprised of a simple random sample, followed by the results on the gender-balanced dataset and the dataset with extra coreference. Finally, we will discuss the impact of using debiased word embeddings.

6.2 Discussion

6.2.1 Experiment 1: Randomly Sampled Data

We find that the baseline Transformer model is able to generate fairly high-quality translations with a BLEU score of 31.46. Indeed, an empirical examination of some of the sentences generated shows that the translations generated by the baseline model have high fluency.

However, in cases where there is an ambiguous pronoun or another word which is genderless in Turkish but gendered in English, the baseline Transformer consistently

Experiment	Context	Training Set	Test BLEU	Parity
1	No	Random	31.46	0.0
1	Yes	Random	31.38	0.18
2	No	Gender Balanced	29.92	0.29
2	Yes	Gender Balanced	29.78	0.42
3	No	Extra Coreference	29.46	0.05
3	Yes	Extra Coreference	30.11	0.05

TABLE 6.1: Test BLEU scores and parity scores on the pronoun accuracy task described above. All experiments use the Transformer as the basic architecture. ‘Context’ indicates whether the model used is the context-aware Transformer, as proposed by Voita et al. (2018), or the non-context-aware Transformer, as originally presented by Vaswani et al. (2017). ‘Training Set’ indicates what data the model was trained on. All training datasets are samples of 2 million sentences from the total training dataset, where ‘Random’ is a simple random sample, ‘gender-balanced’ indicates a sample that has an equal proportion of sentences with male pronouns as female pronouns, and ‘extra coreference’ indicates a sample that has additional samples that exhibit coreference between the context sentence and the source sentence.

predicts the male version of the word. Examples of this phenomenon are shown in Figure 6.1 and are consistent with the results in Table 6.1, where the baseline Transformer model achieved a parity score of 0 on the pronoun accuracy task by predicting ‘he’ for every example and never predicting ‘she’.

The context-aware Transformer had a similar BLEU score of 31.38 but was able to achieve a higher score on the pronoun accuracy task, with a parity score of 0.18. From the examples in Figure 6.2, we can see that in addition to gains in pronoun correctness, the context-aware Transformer exhibits comparable fluency in its translations compared to the baseline Transformer. The improvements in pronoun correctness support the results in Table 6.1, which show an improvement in parity score by 0.18 on the pronoun accuracy task by the context-aware Transformer relative to the baseline.

However, we should note that the context-aware model was still unable to learn to change which pronoun it predicted based on context. That is, it was unable to realize that because the sentences

[Context]: *Adamın mesleği nedir?* [Source]: *O bir doktor.* → [Target] *He is a doctor.*

and

[Context]: *Kadının mesleği nedir?* [Source]: *O bir doktor.* → [Target] *She is a doctor.*

differed in context, it should predict a different pronoun in each of the cases. Instead, it predicted the same pronoun regardless of context. There are several possible reasons for

Hypothesis: each of you was chosen by your own free will to spend the night in prison to earn \$00,000 a head.

Reference: all of you have chosen, of your own free will, to spend the night in this prison for the chance to win \$20,000 each.

Hypothesis: but these small fish spend most of their time in the depths of the ocean where dolphins can't reach.

Reference: but these small fish spend most of their time down in the deep ocean, way beyond the reach of dolphins.

Hypothesis: i need to watch my little brother.

Reference: i have to look after my baby sister.

Hypothesis: he had a heart attack.

Reference: she's had a heart attack.

FIGURE 6.1: Examples generated from the baseline Transformer. Note that although the sentences have high fluency, the model tends to predict the male form of ambiguous words. Incorrectly gendered words are highlighted in red, with their correct forms in blue.

this, the most likely reason being that there were simply insufficient examples of both cases. This could potentially be improved by data augmentation, where we include examples of both pro-stereotypical and anti-stereotypical cases.

Baseline Hypothesis: you know, i wanted to do something special to show **him** that i care about **him**.

Context-Aware Hypothesis: you know, i wanted to do something special to show **her** that i care about **her**.

Reference: you know, i just wanted to make **her** something special to show that i care.

Baseline Hypothesis: despite their enormous size blue whales are one of the most aerodynamic and energy savings within all swimming animals.

Context-Aware Hypothesis: despite their great size, the blue whales are the most aerodynamic and energy-minded in all the floating animals.

Reference: despite their immense size blue whales are one of the most streamlined and energy-efficient of all swimmers.

FIGURE 6.2: Comparison of examples from the baseline Transformer and the context-aware Transformer. The context-aware Transformer is able to translate pronouns correctly at a higher rate than the baseline Transformer, and the fluency of more complicated translations is comparable. Incorrectly gendered words are highlighted in red, with their correct forms in blue.

6.2.2 Experiment 2: Gender Balanced Data

Next, we experimented with increasing the count of sentences with female pronouns in the training set such that the number of sentences with female pronouns and the number of sentences with male pronouns was equal, based on the analysis in Chapter 4 showing the substantially higher count of male pronouns in the original dataset.

We note that we only balanced the marginal counts of each pronoun, so the contexts for the occurrences of masculine and feminine pronouns are likely still very stereotypical. A stronger definition of ‘balancing’ would ensure that every instance of a gendered pronoun was ‘balanced’ with a possibly synthetic example of an identical situation with the opposite pronoun. Although this is beyond the scope of this work, we explore possible methods of doing so in Chapter 7.

Using this training set, we found that both the Transformer and the context-aware Transformer were able to substantially improve on the pronoun translation task, as reported in Table 6.1. The non-context-aware Transformer was able to achieve a parity score of 0.29 at a BLEU score of 29.92, and the context-aware Transformer was able to achieve a parity score of 0.42 at a BLEU score of 29.78.

Some example translations are shown in Figure 6.3. Both the context-aware and non-context-aware Transformers trained on the gender-balanced dataset predict ‘she’ at a higher rate than the baseline trained on the non-gender-balanced dataset. Occasionally, the context-aware Transformer trained on the gender-balanced dataset will predict ‘she’ even when the correct translation should be ‘he’.

<i>Baseline Hypothesis:</i> try to remember everything he said about me, okay? <i>Gender-Balanced Non-Context-Aware Hypothesis:</i> try to remember everything she said about me, okay? <i>Gender Balanced Context-Aware Hypothesis:</i> try to remember everything she said about me, okay? <i>Reference:</i> try to remember everything she said about me, okay?
<i>Baseline Hypothesis:</i> yeah, i did that for him. <i>Gender-Balanced Non-Context-Aware Hypothesis:</i> yeah, i did that for him. <i>Gender-Balanced Context-Aware Hypothesis:</i> yeah, i did it for her. <i>Reference:</i> yeah, i just made this for her.
<i>Baseline Hypothesis:</i> and you sent him after me? <i>Gender-Balanced Non-Context-Aware Hypothesis:</i> and you sent him after me? <i>Gender-Balanced Context-Aware Hypothesis:</i> and you sent her after me? <i>Reference:</i> and still, you sent her after me?
<i>Baseline Hypothesis:</i> he got me a flight. <i>Gender-Balanced Non-Context-Aware Hypothesis:</i> he's booked me on a flight. <i>Gender-Balanced Context-Aware Hypothesis:</i> she's booked me on a flight. <i>Reference:</i> he's booked me on a flight.

FIGURE 6.3: Comparison of examples from both context-aware and non-context aware models trained on gender-balanced data, along with examples from the baseline model. Correctly gendered words are in blue, and incorrectly gendered words are in red.

6.2.3 Experiment 3: Extra Coreference Dataset

We then experimented with increasing the percent of training samples that exhibited coreference from the context sentence to the source sentence, referred to as the 'Extra Coreference' training set. From Table 6.1, we notice that both the context-aware Transformer and the non-context-aware Transformer exhibit very little improvement in parity score on the pronoun accuracy task compared to the baseline. Therefore, we conclude that simply providing examples of coreference is insufficient for the model to generalize when the examples contain very few of the underrepresented pronoun in question.

From a qualitative observation of some of the translations shown, we observe that the models trained on the data with extra coreference tend to disagree on pronouns compared to the baseline model. However, there is not a consistent pattern as to which one is correct. The context-aware model trained on the Extra Coreference data does tend to predict feminine pronouns slightly more often than the other models, but sometimes it also predicts the feminine pronouns even when the correct pronoun is masculine. We find it difficult to make definitive conclusions about the performance of our models after

Baseline Hypothesis: **him** too.

Extra Coreference Non-Context-Aware Hypothesis: that too.

*Extra Coreference Context-Aware Hypothesis:*that too.

Reference: **her**, too.

Baseline Hypothesis: you didn't ask them to kill you for them.

Extra Coreference Non-Context-Aware Hypothesis: he did not ask you to speak for them, to kill for them.

Extra Coreference Context-Aware Hypothesis: no one asked you to speak for them, to kill for them.

Reference: nobody asked you to speak for them, to kill for them.

Baseline Hypothesis: **it** wasn't caused by problems.

Extra Coreference Non-Context-Aware Hypothesis: **that** didn't cause problems.

Extra Coreference Context-Aware Hypothesis: **this** wasn't caused by problems.

Reference: **this** wasn't caused by troubles.

Baseline Hypothesis: you told his wife you'd find **him**.

*Extra Coreference Non-Context-Aware Hypothesis:*you told his wife you'd find **him**.

Extra Coreference Context-Aware Hypothesis: you told his wife you'd find **her**.

Reference: you told his wife that you would find **him**.

FIGURE 6.4: Comparison of examples from both context-aware and non-context aware models trained on data with extra examples of coreference, along with examples from the baseline model. Although these models frequently disagreed with respect to pronouns, there was not one particular model which consistently translated pronouns correctly. Correct forms of pronouns are highlighted in blue, and incorrectly translated forms are highlighted in red.

adding examples with extra coreference.

Model	Epochs	BPE	Src-Trg	Embeddings	Test BLEU	Parity
Baseline	100	Y	Shared	Learned	31.46	0.0
Transformer	500	N	Separate	Learned	28.23	0.0
Transformer	300	N	Separate	Debiased	28.46	0.0

TABLE 6.2: Results from Experiment 4. All models use a non-context-aware Transformer trained on the same random sample of 2 million sentences from the training set. Debiased word embeddings does not improve performance on the pronoun accuracy task compared to others models trained on the same data. Furthermore, not using BPE causes a substantial decrease in BLEU and significantly longer training time.

6.2.4 Experiment 4: Impact of Debiased Word Embeddings

The results from using debiased word embeddings are shown in Table 6.2. First, we observe that by not using byte-pair encoding and not sharing the source-target embeddings, which is required in order to use debiased word embeddings, the translation quality decreases substantially. Interestingly, we note that using the debiased embeddings yielded a higher BLEU score (28.46) as opposed to not using pre-trained word embeddings (28.23), but we attribute this to having substantially fewer parameters to optimize. Since the pre-trained embeddings were trained on the same data as the overall model was, it makes sense that the pre-trained embeddings would be fairly effective. However, we note that there was no improvement in the accuracy of pronoun translations between using the debiased embeddings and not using them, as all three models predicted ‘he’ in all cases. We do note that in a qualitative comparison of the translations, there were some improvements in pronoun translation, as shown in Figure 6.5. These results mostly support those found by Park, Shin, and Fung (2018), which found that debiased word embeddings alone were insufficient to reduce bias.

A practical note we think is worthwhile to mention is that by not using BPE and not sharing source-target embeddings, the training time increased substantially. While the baseline was trained in 100 epochs, taking about 1 day on an NVIDIA V100 Tensor Core GPU, the models that did not use byte pair encoding or source-target embeddings took 300 - 500 epochs, taking about 3 - 5 days on the same GPU. We believe that such a substantial increase in training time and degradation in performance, without any noticeable improvements in pronoun accuracy, mean that debiased embeddings are not a feasible solution to pronoun translation.

Baseline Hypothesis: **he** had a heart attack.

Separate Embeddings Hypothesis: **he** had a heart attack.

Debiased Embeddings Hypothesis: **he** had a heart attack.

Reference: **she**'s had a heart attack.

Baseline Hypothesis: **he**'s okay now, but **he**'s serious.

Separate Embeddings Hypothesis: **he**'s all right now, but we'll figure it out.

Debiased Embeddings Hypothesis: **she**'s fine now, but she's okay.

Reference: **she**'s okay, but it's pretty serious.

Baseline Hypothesis: but these small fish spend most of their time in the depths of the ocean where dolphins can't reach.

Separate Embeddings Hypothesis: but most of these little <unk> have been <unk> in the middle of the ocean where the ocean had been <unk>

Debiased Embeddings Hypothesis: but these little <unk> <unk> spend most of their time in the middle of the ocean.

Reference: but these small fish spend most of their time down in the deep ocean, way beyond the reach of dolphins.

FIGURE 6.5: Comparison of examples from a model that uses debiased word embeddings compared to one that does not. The model with debiased word embeddings uses the female pronoun correctly slightly more often than the baseline does, but very rarely. Both models that do not use BPE suffer from inability to process unknown tokens apart from as <unk>, as well as lower fluency than the baseline model.

6.2.5 Summary of Results

We introduce our own pronoun translation task, meant to gauge if a model can translate an ambiguous pronoun correctly in both pro-stereotypical and anti-stereotypical cases. For each experiment, we report both BLEU as well as performance on the pronoun translation task.

From Experiment 1, we find that the baseline Transformer generates high-quality translations, but performs the worst on the pronoun translation task, as it consistently predicts ‘he’ for all ambiguous pronoun. We find that the context-aware Transformer consistently performs slightly better at the pronoun translation task than its non-context-aware counterpart in all settings while achieving comparable BLEU.

From Experiment 2, we find that the most substantial gains in pronoun prediction accuracy resulted from modifying the training data to be more gender-balanced. Both non-context-aware and context-aware Transformer models benefited from this modification of the training data; however, the context-aware Transformer achieved the best results on the pronoun accuracy task.

From Experiment 3, we find that adding additional examples exhibiting coreference does not result in substantive changes in performance in either the context-aware Transformer or the non-context-aware Transformer.

Finally, in Experiment 4, we experiment with using debiased word embeddings. We find that they are insufficient to significantly improve performance on the pronoun prediction task, and lead to substantial losses in BLEU and increases in training time.

Chapter 7

Conclusion

In this thesis, we began by motivating research in gender bias in machine translation through the example of Google Translate’s biased resolution of ambiguous pronouns. We then presented background material on state-of-the-art developments in neural machine translation: in particular, a model known as the Transformer and its extensions in context-aware translation. We also introduced the tasks anaphora resolution and coreference resolution, related tasks that are crucial to improving the translation of ambiguous pronouns. Additionally, we presented different definitions of bias and fairness as it relates to classification.

We then delved more deeply into current approaches taken to analyzing gender bias in NLP, including using debiased word embeddings, data augmentation, and the Wino-Bias / WinoGender tasks. We discuss different approaches researchers have taken to quantifying the gender bias of a machine translation system. Finally, we closely examined the Transformer and its context-aware extensions as a potential solution to resolving ambiguous pronouns.

We then move on to the experimental portion of this thesis, beginning with a detailed data analysis of the training dataset to motivate the experiments in the next section. Based on our analysis, we present three main experiments: using a context-aware Transformer, modifying the training data to be more ‘gender-balanced’, increasing the number of examples that exhibit coreference, and using debiased word embeddings.

We found that the baseline Transformer generates relatively high-quality translations, but performs the worst on the pronoun translation task, predicting ‘he’ in every case. The context-aware Transformer improved upon this performance in the pronoun translation task and was able to generate translations of comparable quality.

However, the most significant improvements on the pronoun translation task were obtained by modifying the training data to be more gender-balanced, such that the fraction of examples that had a female pronoun was equal to the fraction of examples that had a male pronoun.

We also experimented with modifying the data to have more examples of coreference. However, we did not find any significant improvement on the pronoun translation

task.

Finally, we found that using debiased word embeddings to be insufficient to significantly improve ambiguous pronoun translation, while also leading to worse BLEU scores.

7.1 Future Work

We conclude that while context-aware translation does seem to be a promising avenue to improve the disambiguation of genderless pronouns, there are several limitations. First, we were unable to find an example where the translation of an ambiguous pronoun switches purely based on context. This is most likely because the training data did not have examples of such a switch, as we found in Chapter 4 that the majority of examples that exhibited coreference did not use personal pronouns. However, it is infeasible to assume that training data in the future will have both pro-stereotypical and anti-stereotypical examples. As such, it seems that data augmentation with synthetic examples will be an important extension to include both pro-stereotypical and anti-stereotypical examples. However, data augmentation is challenging to perform in text settings because of the unstructured nature of text compared to tabular data. A naive approach could be to add synthetic examples either by adding templated examples, as done by (Dixon et al., 2018), or even simply by replacing masculine pronouns with feminine pronouns. A more sophisticated approach might call upon conditional text generation to generate more complex examples that exhibit some desired pattern such as being pro-stereotypical and anti-stereotypical.

Another reason the context-aware Transformer might have struggled is that a discourse involving personal pronouns is unlikely to continuously redefine the antecedent throughout the discourse. Rather, humans tend to maintain knowledge of the antecedent a personal pronoun for a longer length of time than, for instance, a non-personal pronoun like ‘it’. This is especially in the case of movies when visual cues provide a reference for disambiguating context that we cannot use in translation. As such, another avenue could be to extend the amount of context used, though this approach will necessitate increased computing power due to more data. To decrease the amount of context that needs to be processed, we might consider using a hierarchical model instead. These have seen success in document classification by utilizing a structure that incorporates word-level and sentence-level information separately in a hierarchical manner that mimics the structure of documents (Yang et al., 2016). We could also try to find or create a different dataset that exhibits more examples of coreference among personal pronouns.

Finally, a limitation to the bias quantification task presented in this thesis is that it is modified for each new pair of languages that one wishes to analyze or for other

dimensions of bias apart from gender. We chose Turkish specifically because it has genderless pronouns, but what about the gender bias of languages that do not have genderless pronouns? For example, the translation from English to Spanish of

My friend is a doctor → Mi amigo es un doctor

but

My friend is a nurse → Mi amiga es enfermera

echoes the same behavior as the Google Translate Turkish example. We can imagine creating an analogous pronoun task as the one presented in this thesis, except instead of classifying the disambiguation of a pronoun, we would classify the disambiguation of a gendered word (in this case ‘amigo’/‘amiga’). Still, this requires the creation of a new test set for each pair of languages or dimension of bias through manual compilation. As such, a more systematic way of generating examples to test the bias of a system is necessary.

Bibliography

- Bahdanau, Dzmitry, Kyunghyun Cho, and Yoshua Bengio (2014). “Neural Machine Translation by Jointly Learning to Align and Translate”. In: *CoRR* abs/1409.0473.
- Bawden, Rachel et al. (2017). “Evaluating Discourse Phenomena in Neural Machine Translation”. In: *CoRR* abs/1711.00513. arXiv: 1711.00513. URL: <http://arxiv.org/abs/1711.00513>.
- Bolukbasi, Tolga et al. (2016). “Man is to Computer Programmer as Woman is to Home-maker? Debiasing Word Embeddings”. In: *CoRR* abs/1607.06520.
- Buolamwini, Joy and Timnit Gebru (2018). “Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification”. In: *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*. Ed. by Sorelle A. Friedler and Christo Wilson. Vol. 81. Proceedings of Machine Learning Research. New York, NY, USA: PMLR, pp. 77–91. URL: <http://proceedings.mlr.press/v81/buolamwini18a.html>.
- Cho, Kyunghyun et al. (2014). “Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation”. In: *CoRR* abs/1406.1078.
- Chouldechova, Alexandra (2017). “Fair prediction with disparate impact: A study of bias in recidivism prediction instruments”. In: *Big data* 5 2, pp. 153–163.
- Covington, Paul, Jay Adams, and Emre Sargin (2016). “Deep Neural Networks for YouTube Recommendations”. In: *Proceedings of the 10th ACM Conference on Recommender Systems*. RecSys ’16. Boston, Massachusetts, USA: ACM, pp. 191–198. ISBN: 978-1-4503-4035-9. DOI: 10.1145/2959100.2959190. URL: <http://doi.acm.org/10.1145/2959100.2959190>.
- Dastin, Jeffrey (2018). “Amazon scraps secret AI recruiting tool that showed bias against women”. In: *Reuters*. URL: <https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G>.
- Dixon, Lucas et al. (2018). “Measuring and Mitigating Unintended Bias in Text Classification”. In: pp. 67–73. DOI: 10.1145/3278721.3278729.
- Elaraby, Mostafa et al. (2018). “Gender Aware Spoken Language Translation Applied to English-Arabic”. In: *CoRR* abs/1802.09287. arXiv: 1802.09287. URL: <http://arxiv.org/abs/1802.09287>.
- Font, Joel Escudé and Marta R. Costa-Jussà (2019). “Equalizing Gender Biases in Neural Machine Translation with Word Embeddings Techniques”. In: *CoRR* abs/1901.03116.

- He, Kaiming et al. (2015). “Deep Residual Learning for Image Recognition”. In: *CoRR* abs/1512.03385.
- Heinzerling, Benjamin and Michael Strube (2018). “BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages”. In: *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Ed. by Nicoletta Calzolari (Conference chair) et al. Miyazaki, Japan: European Language Resources Association (ELRA). ISBN: 979-10-95546-00-9.
- Kuczmarski, James (2018). *Gender-specific translations on the Google Translate website*. Google. URL: blog.google.com.
- Lapowsky, Issie (2018). “Google Autocomplete Still Makes Vile Suggestions”. In: *Wired*. URL: GoogleAutocompleteStillMakesVileSuggestions.
- Lison, Pierre, Jörg Tiedemann, and Milen Kouylekov (2018). “OpenSubtitles2018: Statistical Rescoring of Sentence Alignments in Large, Noisy Parallel Corpora”. In: *LREC*. European Language Resources Association (ELRA).
- Manning, Christopher D. et al. (2014). “The Stanford CoreNLP Natural Language Processing Toolkit”. In: *Association for Computational Linguistics (ACL) System Demonstrations*, pp. 55–60. URL: <http://www.aclweb.org/anthology/P/P14/P14-5010>.
- Park, Ji Ho, Jamin Shin, and Pascale Fung (2018). “Reducing Gender Bias in Abusive Language Detection”. In: *CoRR* abs/1808.07231.
- Prabhumoye, Shrimai et al. (2018). “Style Transfer Through Back-Translation”. In: *CoRR* abs/1804.09000. arXiv: 1804.09000. URL: <http://arxiv.org/abs/1804.09000>.
- Press, Ofir and Lior Wolf (2016). “Using the Output Embedding to Improve Language Models”. In: *CoRR* abs/1608.05859.
- Rudinger, Rachel et al. (2018). “Gender Bias in Coreference Resolution”. In: *CoRR* abs/1804.09301. arXiv: 1804.09301. URL: <http://arxiv.org/abs/1804.09301>.
- Silver, David et al. (2016). “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529, pp. 484–503. URL: <http://www.nature.com/nature/journal/v529/n7587/full/nature16961.html>.
- Sutskever, Ilya, Oriol Vinyals, and Quoc V. Le (2014). “Sequence to Sequence Learning with Neural Networks”. In: *CoRR* abs/1409.3215.
- Tatman, Rachael (2017). “Gender and Dialect Bias in YouTube’s Automatic Captions”. In: pp. 53–59. DOI: 10.18653/v1/W17-1606.
- Vanmassenhove, Eva, Christian Hardmeier, and Andy Way (2018). “Getting Gender Right in Neural Machine Translation”. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, pp. 3003–3008. URL: <http://aclweb.org/anthology/D18-1334>.
- Vaswani, Ashish et al. (2017). “Attention Is All You Need”. In: *CoRR* abs/1706.03762.

- Voita, Elena et al. (2018). "Context-Aware Neural Machine Translation Learns Anaphora Resolution". In: *CoRR* abs/1805.10163. arXiv: 1805.10163. URL: <http://arxiv.org/abs/1805.10163>.
- Webster, Kellie et al. (2018). "Mind the GAP: A Balanced Corpus of Gendered Ambiguous Pronouns". In: *CoRR* abs/1810.05201. arXiv: 1810.05201. URL: <http://arxiv.org/abs/1810.05201>.
- Yang, Zichao et al. (2016). "Hierarchical Attention Networks for Document Classification". In: *HLT-NAACL*.
- Zhao, Jieyu et al. (2018a). "Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods". In: *CoRR* abs/1804.06876. arXiv: 1804.06876. URL: <http://arxiv.org/abs/1804.06876>.
- Zhao, Jieyu et al. (2018b). "Learning Gender-Neutral Word Embeddings". In: *CoRR* abs/1809.01496.