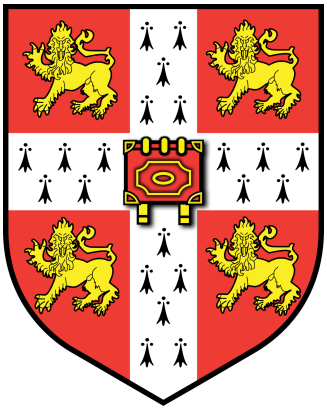

Reducing Gender Bias in Neural Machine Translation

MPhil MLMI

MLMI 8 Neural Machine Translation

Candidate Number
J902G

Date:
April 17, 2022

1 Overview

Whilst state-of-the-art machine translation (MT) systems are able to perform text translation at high accuracy, they suffer from major gender bias effects due to biases in the data that they have been trained on [7]. This bias not only accentuates social stereotypes[1] but also degrades from the quality of the machine translation, making addressing gender bias a topic of interest within the Neural Machine Translation (NMT) community.

Historical approaches have involved re-training MT systems from scratch on non-biased word embeddings or fine-tuning them on smaller debiased data-sets [8]. Whilst these methods have improved gender accuracy (the percentage of translation sentences correctly gendered), they have the drawback that catastrophic forgetting causes overall translation accuracy to decrease.

This work aims to address this, and looks at a lattice rescoring scheme, where a single gendered word in a translation is mapped to a lattice of multiple gender inflected alternatives via Finite-State Transducers (FSTs), as introduced in [7]. The objective of this project is then to carry out rescoring on these multi-gendered lattices to produce translations that are both accurate translations (as the basic structures of the MT system translations have not changed), whilst boasting improved gender accuracy by selecting better gender inflections of each translated word.

To test the translation and gender accuracies of these systems, two data-sets are used. The first is the WMT18 data-set consisting of 2,998 translated phrases, which are used to assess the translation accuracy before and after lattice rescoring (with translation accuracy gauged via the BLEU score). The second is the WinoMT data-set [5] containing 3,888 phrases, each containing a statement about a profession with an associated gender. This data-set is used to calculate gender accuracy and the difference in F1 scores between male and female translations.

The FST architectures used in this practical follow the AT&T FSM format, and are compiled and manipulated using the OpenFST module.

[1] For example, in 2018 Amazon had to abandon plans to use a machine learning based CV ranker after realising that biased training data had made it favour men for technical jobs over women [4].

2 Exercise GDBNMT.1

Word to Class Transducer:

The word to class flower transducer `wtoc.fst` is produced by first writing an arc format file (using Python) of the form "`0 0 <word> <class>`" for each word-class mapping in `$GDBNMTBDIR/wordclasses`, and then compiling that using the `fstcompile` command, where the input and output symbols are both given by `$GDBNMTBDIR/fsts/w+l.map.de`.

It is important to note that there are input strings in the translation files that are not included in the `$GDBNMTBDIR/wordclasses` file but are in the symbols file. Therefore, any string (that is not a class) in the symbols file that was not already in the FST was also added and mapped to itself. Care had to be taken to exclude any class-to-class mappings from the symbol file here, as there are no class inputs in the translations so they are not necessary, and including them would produce problems when inverting for the class to word transducer later.

When composed (via `fstcompose`) with the input translations, the word to class flower transducer performed as required, giving outputs of the form (WMT18's 1.fst is used as an example here):

```
<s> München 1856 :  Vize.l.1 Kaue.l.12 , diese.l.3 diese.l.3 Blick auf diese.l.3 Stade.l.4 verändern
</s>
<s> München 1856 :  Vierte.l.4 Kaue.l.12 , diese.l.3 diese.l.3 Blick auf diese.l.3 Stade.l.4 verändern
</s>
```

Gender Mapping Transducer:

With a transducer that maps single words to multple classes, it is now possible to compose it with its inverse to produce one that maps single words to multiple words of the same class but different gender, as shown in the toy example in Figure 1.

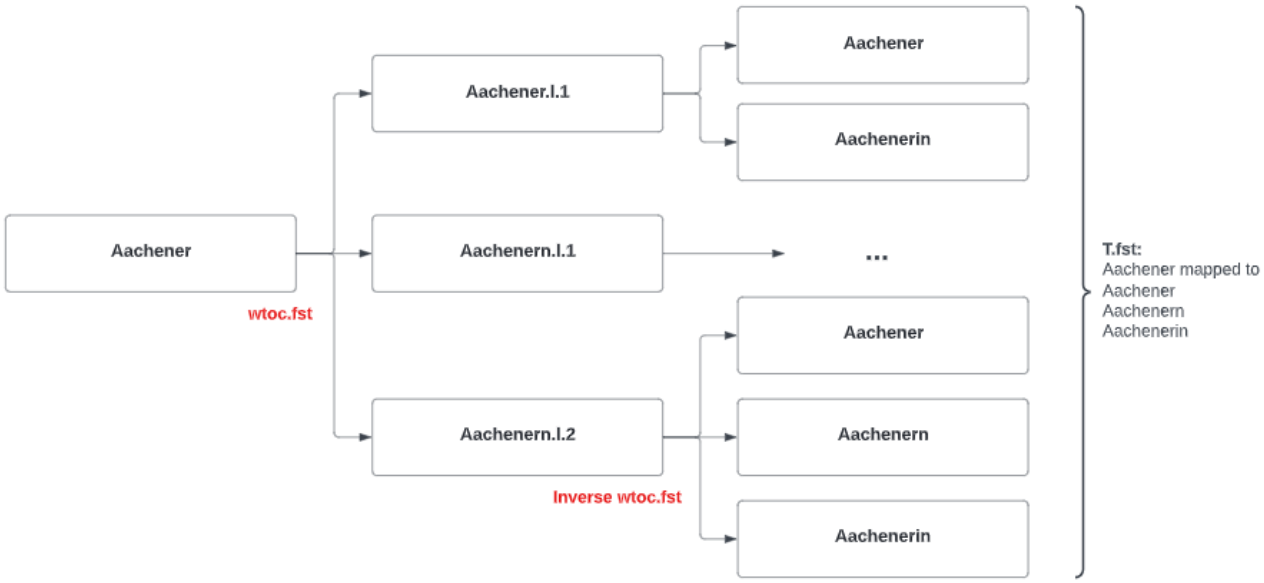

Figure 1: Diagram of a toy example demonstrating what is happening when `wtoc.fst` is composed with its inverse. The first mapping is word to class and uses the flower transducer `wtoc.fst`, the second is class to word, and is the inverse of `wtoc.fst`. The resultant is a transducer `T.fst` that maps single gendered words to a set of multi-gendered ones.

The OpenFST commands to do this are as in the following code listing:

```
fstinvert wtoc.fst |\
fstarcsort --sort_type=ilabel - | \
fstremepsilon - ctow.fst

fstarcsort --sort_type=olabel wtoc.fst |\
fstcompose - ctow.fst T.fst

fstcompose $GDBNMTBDIR/fsts/wmt18.sgnmt.wmt18ensemble.1/1.fst T.fst |\
fstrmepsilon - 1_gender_mapped.fst
```

In the listing above, the top 3 lines are shell commands to take the already built and compiled `wtoc.fst` and invert it to map from classes back to words (after sorting arcs ready for composition). The middle 2 lines then compose the new `ctow.fst` transducer with the original `wtoc.fst` one, which has the overall effect of mapping a single word to many different gendered forms of itself (see Figure 1 for a diagramatic explanation). This net transducer is called `T.fst`. The bottom 2 lines are used for each specific translation file, taking the gender mapping transducer `T.fst` and applying it on each input translation file (WMT18's 1.fst is used in the code listing). The `fstrmepsilon` commands remove epsilon-to-epsilon mappings to simplify the FST and the `fstarcsort` command is a means of correctly aligning the FSTs that are being composed.

Following the pipeline of commands above for each translation file maps each translated sentence to every gendered alternative. For example, for WMT18's 1.fst, the first 4 (of 175,104[2]) lines of the output `1_gender_mapped.fst` read:

```
<s> München 1856 :  Vize Katen , dieser diesen Blick auf dieser Stab verändern </s>
<s> München 1856 :  Vize Karteien , dieser diesen Blick auf dieser Stabes verändern </s>
<s> München 1856 :  Vikars Kauen , dieser diesen Blick auf die Stadt verändern </s>
<s> München 1856 :  Vize Kauen , dieser diesen Blick auf dieser Stabes verändern </s>
```

The overall FST that produces the gender inflected phrases printed above can be seen in Figure 2 below. The overall number of possible sentence combinations is vast due to the combinatoric number of different pathways.

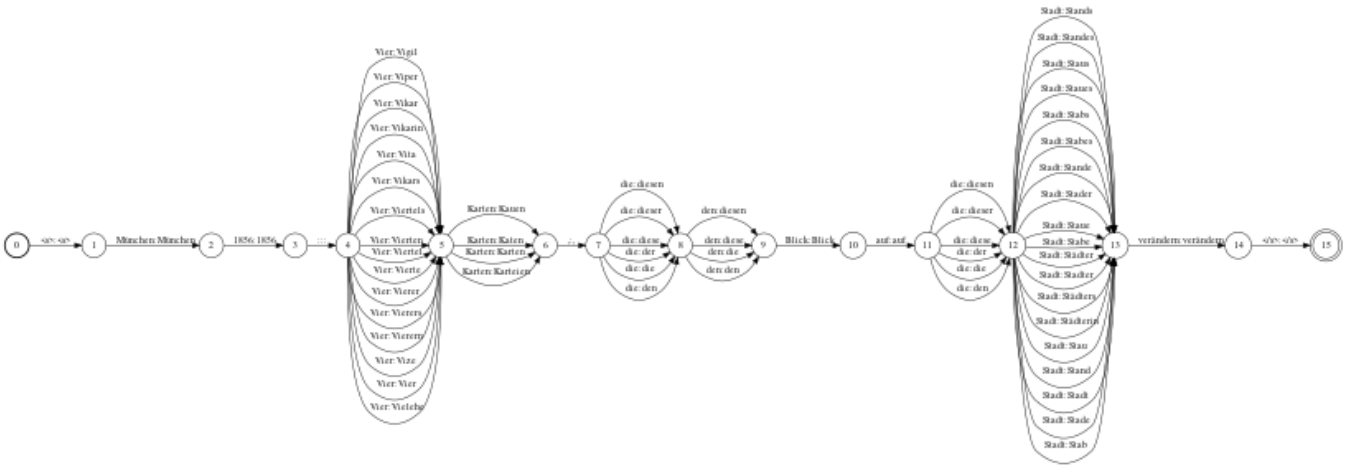

Figure 2: Diagram of the resultant FST when WMT18's 1.fst is composed with `T.fst`.

[2]There are $1 \cdot 1 \cdot 1 \cdot 16 \cdot 4 \cdot 1 \cdot 6 \cdot 4 \cdot 1 \cdot 1 \cdot 6 \cdot 19 \cdot 1 \cdot 1 = 175,104$ pathways.

3 Exercise GDBNMT.2

The next transducer maps from the multi-gendered translations generated by `T.fst` in Question 1 to the Byte Pair Encoded (BPE) version. This transducer is more complex than the previous two as now a single string is mapped to multiple strings, for example 'Abbruchmethoden' to 'Ab@@ bru@@ ch@@ methoden'. This introduces the need to use intermediary nodes within a petal of the flower transducer and make use of `<epsilon>` symbols to ensure correct sub-sequence order within the petal.

This is most easily seen through a diagram, and so Figure 3 shows a toy example using only 3 words for simplicity. The word to BPE flower transducer (top) has multiple nodes per petal if the mapped sequence has multiple sub-sequences, and uses `<epsilon>` inputs to signify no additional input. The bottom of Figure 3 shows the FST for a simple translated sentence 'A abbruchmethoden abati' (the words were selected at random and the sentence is not intended to make sense in German).

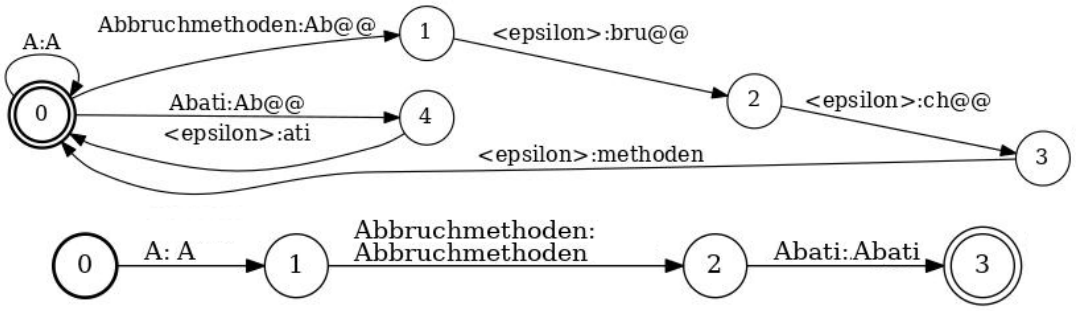

Figure 3: A toy example of using a flower transducer when there are multiple sub-sequences within a sequence mapping. `<epsilon>` symbols are used to signify no extra input after the first word in a petal, but allows for the continuation of that sequence in the mapping.

Composing these two FSTs together then generates the BPE string as shown below in Figure 4.

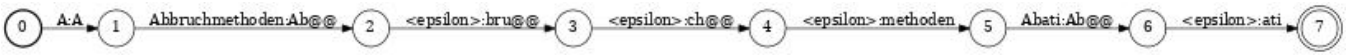

Figure 4: The output of the toy example in Figure 3. When composed with the input sequence, the word to BPE flower transducer breaks aQuestion the sequence into alternatively gendered sub-words. Due to the simplicity of this example, only one set of sub-words is generated, but in the real case, there are multiple per input phrase.

This process is the same as that carried out on the real translation texts, except the `wtobpe.fst` includes all word to BPE mappings from `$GDBNMTBDIR/fairseq.pretrained/word_bpe.dict` in addition to self-mappings for any other required symbols (for example `<s>, </s>, <unk>`). The translation FSTs are also the WMT18 and WinoMT translations used thus far in the report.

The `wtobpe.fst` transducer is now composed with the multi-gendered FST outputs from Question 1 such that the whole pipeline is now `wtoc.fst + Inverse wtoc.fst + wtobpe.fst +` n`.fst` $\rightarrow$ n`_BPE.fst` for each file n`.fst` in both the WinoMT and WMT18 translation sets. The content of the output files then take the following form (where again WMT18's 1.fst is used as an example):

```
<s> München 18@@ 56 :  Viertel K@@ aten , den diesen Blick auf diesen Stan@@ des verändern </s>
<s> München 18@@ 56 :  Vier Karten , diesen diese Blick auf die St@@ ab@@ s verändern </s>
<s> München 18@@ 56 :  Vier@@ er K@@ aten , diesen diese Blick auf die St@@ ab@@ s verändern </s>
<s> München 18@@ 56 :  V@@ ierten K@@ aten , diesen diese Blick auf die St@@ ab@@ s verändern </s>
```

4 Exercise GDBNMT.3

Using the BPE FSTs for each of the translations in the WMT18 and WinoMT data-sets, the remapping of the start symbol `<s>` for the Fairseq/PyTorch scoring libraries is done by composing these FSTs with `remapstartsym.fst` as provided. It is also possible to optimise these FSTs prior to remapping to improve the rescoring procedure and thus also improve gender accuracy. This is done in practice using the `fstdeterminize` and `fstminimize` OpenFST shell commands (N here is each file in the respective data-set, and WinoMT is used as the example, but this is also applied to the WMT18 data-set):

```
fstarcsort --sort_type=olabel - fsts/winomt.sgnmt.wmt18ensemble.1.ga/N.fst |\
fstproject --project_type=output - |\
fstdeterminize - |\
fstminimize - |\
fstcompose - $GDBNMTBDIR/fsts/remapstartsym.fst |\
fstproject --project_type=output > fsts/winomt.sgnmt.wmt18ensemble.1.ga/N_remapped.fst
```

The `fstdeterminize` command carries out determinisation of the FST. This is achieved by ensuring that the FST has a unique starting state and no arcs from the same state share the same input label. This means that the BPE FST input from Question 2 had to be projected onto its outputs using `fstproject` prior to determinisation for it to work. This is done without loss of information, as the inputs of each arc between any two fixed states are the same (as they represent a single word in the original translation), it is the outputs that are different. The result of determinisation is an FST that performs in the same manner as before, but that is smaller and requires less space or computation [2].

After determinisation, minimisation can be carried out via `fstminimize`. This process creates a version of the current FST that performs exactly the same mappings on input strings, but with the minimal number of states and arcs necessary to do so [2], thus reducing the size and complexity of the FST without any effect on functionality.

As is expected, this optimisation and start symbol remapping do not change the resultant mapping from single translation to gender inflected WFSAs, but solely make the lattices more optimised and ready for rescoring. Therefore, as a test, 4 of the output phrases produced from the remapped WMT18 1.fst were generated and compared to the output of Question 2. They are shown below, and whilst they do not exactly match (due to random ranking of the lattices in printing), it is apparent from how similar they are in structure that this process has not affected the contents of the outputs. It was also manually verified that all 4 lines below exist in the 175,104 output lines of WMT18's 1_BPE.fst from Question 2 (and vice versa).

```
München 18@@ 56 :  Vier K@@ auen , diesen diesen Blick auf diesen Stau@@ es verändern </s>
München 18@@ 56 :  Vier Kar@@ teien , diesen die Blick auf diesen Stau@@ es verändern </s>
München 18@@ 56 :  Vier Kar@@ teien , diesen diesen Blick auf diesen Stau@@ es verändern </s>
München 18@@ 56 :  Vier Karten , diesen diesen Blick auf dieser St@@ abe verändern </s>
```

The effect of the optimisation on the translation and gender accuracy can be seen in Question 4, where accuracy statistics for optimised and un-optimised FSTs are reported.

5 Exercise GDBNMT.4

Historical approaches to gender debiasing of MT tasks have involved methods such as fine-tuning an existing system on a small de-biased data-set, due to the difficulty of producing by hand large data-sets that are not gender biased. A shortcoming of this approach is that whilst gender accuracy increases, the transfer learning process degrades translation accuracy compared to the original system [7]. The purpose of this project was to reduce gender bias whilst retaining as much of the translation accuracy of the original system as possible. Therefore, success would be represented by a translation accuracy (given by the BLEU score) that is similar after the lattice rescoring to before, in addition to an improved gender accuracy.

Such tests were thus carried out on the WMT18 data-set (for BLEU score evaluation) and WinoMT data-set (for gender accuracy evaluation), and their results shown in Subsection 5.1 and 5.2 respectively. Prior to applying the lattice based rescoring method the raw translations were the output of one component of Facebook-FAIR's ensemble of transformer models, which had performed very strongly on En-De MT tasks in the 2018 Conference on Machine Translation [3]. This system is therefore used for the baseline performances.

5.1 Low Degradation of BLEU scores on WMT18

As mentioned above, the main aim of this approach is to keep the translation accuracy high whilst minimising gender bias (increasing gender accuracy). Therefore, it is important to compare the translation accuracies before and after the FST based rescoring method is applied to the raw translations. The means of testing accuracy is via the BLEU score [1], a commonly used metric for measuring accuracy in translation tasks. It has a value between 0 and 1 and combines different n-gram precisions between a reference and hypothesis phrase, also including a brevity penalty to penalise the over-insertion of words into the hypothesis.

	BLEU Score	1-gram Precision	2-gram " "	3-gram " "	4-gram " "	Brevity Penalty
Baseline	44.5	73.1	51.3	38.6	30.0	0.976
Un-optimised	42.3	71.7	49.2	36.3	27.6	0.976
Optimised	44.5	73.1	51.2	38.6	30.0	0.975

Table 1: Comparison of BLEU scores [1], n-gram precision ($n = 1...4$), and brevity penalty for the translated WMT18 data-set texts before and after FST based rescoring has been performed. The top row represents the translations before (i.e. the raw outputs of the MT system), and the bottom 2 rows represents them afterwards, with the middle row representing the un-optimised FSTs and the bottom row the optimised ones from Question 3. Un-optimised refers to the FST systems if determinisation and minimisation are not carried out.

Table 1 above shows that whilst there is some degradation to the BLEU score and n-gram precisions caused by the rescoring, the decrease is only very minor (and only in the 2-gram accuracy and brevity penalty in the optimised FST case). It is clear that there is not the extent of catastrophic forgetting with lattice rescoring that is common to other gender debiasing approaches such as fine-tuning on a non-gender biased domain [6, 7].

Therefore this is grounds to suggest that lattice rescoring is an effective approach to make if it is also able to improve gender accuracy. This is discussed in the following subsection.

5.2 Improvements to Gender Bias Scores on WinoMT

In order to gauge efficacy of the lattice rescoring it thus makes sense once again to compare accuracies before and after application, but this time using gender bias metrics.

In order to measure gender bias the original authors of the WinoMT task, Stanovsky et al [5], suggest using various metrics, two of which are overall gender accuracy and the difference between masculine and feminine F1 scores in translation (ΔG). A higher gender accuracy is indicative of better performance, as is smaller ΔG. We also report the male to female ratio indicating the number of male predicted translations to female ones.[3] For this test set, as there were a few neutral sentences, a M/F target of 1.003 was optimal.

The results of the lattice rescoring in terms of gender bias impact can therefore be seen from Table 2, with the best performing for each column shown in bold.

	Accuracy (%)	ΔG (%)	M/F
Baseline	78.3	-3.3 (M F1: 79.6, F F1: 74.3)	**1.000**
Un-optimised Performance	72.5	**1.5** (**M F1 : 75.8**, **F F1 : 74.3**)	1.455
Optimised Performance	**80.7**	-3.5 (M F1: 81.8, F F1: 85.3)	0.970

Table 2: Performance of the FST based rescoring system on the WinoMT data-set compared to the baseline values from the adapted raw Facebook-FAIR model outputs.

Table 2 therefore shows that the un-optimised FST rescoring actually degrades gender accuracy compared to the original translations. Whilst the number of correctly gendered translations are lower, the similarity in treatment between male and female texts is more similar though, as demonstrated by a ΔG nearer to 0. However, even then this is driven predominantly by a lowering of F1 score on male gendered texts compared to the original translations, with no change to female scores. The lattice rescoring procedure was not entirely in vain however, as when the FSTs are optimised using determinisation and minimisation, the gender accuracy improves over the baseline, and gives similar ΔG results (which interestingly are now negative, indicating that the model is more accurate on female text translations than on male ones, which is somewhat counter-intuitive when considering the training data bias is towards male texts). What is more, the optimised lattice rescoring procedure significantly improves F1 score on the female translations compared to the originals, whilst also slightly improving male F1 score. For this reason, it can be concluded that lattice rescoring with optimised FSTs is a worthwhile exercise to mitigate gender bias.

It is worth noting too though that the Facebook-FAIR model has unusually good gender accuracy for a MT system. In fact, the authors of the WinoMT data-set give their own state-of-the-art (SOTA) MT system baselines, namely for Microsoft Translator, Amazon Translate, Google Translate and SYSTRAN [5]. These can be seen in Table 3, and when comparing them with the model performance shown in Table 2 above, it can be seen that the optimised FST rescoring system on Facebook-FAIR translations provides a very large improvement to gender accuracy and ΔG.

[3] It is suggested that M/F ratio is a better predictor of gender bias than ΔS as used in the original [5], as ΔS is easily skewed to near 0 for very low accuracy systems, where almost all translations are single gendered irrespective of assigning pro- or anti-gender stereotypical roles [7].

System	Accuracy (%)	ΔG (%)
Microsoft Translator	74.1	0.0
Amazon Translate	62.4	12.9
Google Translate	59.4	12.5
SYSTRAN	48.6	34.5
Average	61.1	15.0

Table 3: Alternative baseline gender accuracy statistics for the En-De WinoMT data-set when no FST mapping or rescoring has been used on the raw MT outputs. The baselines are quoted from [5].

6 Discussion and Conclusion

This report has shown that lattice rescoring methods using FSTs are an effective means of reducing gender bias in MT tasks without compromising on translation accuracy. Whilst there was a very small drop in BLEU score on the WMT18 test set after producing gender inflected translations, it was very minor and a lot smaller than the drop from other methods such as fine-tuning, where catastrophic forgetting is an issue. What is more, this small decrease in translation accuracy is outweighed by the increase in both gender accuracy and F1 scores on the WinoMT test set, with Question 4 showing an 11% absolute improvement in female F1 score compared to the raw translations from Facebook-FAIR's adapted system. The difference in F1 scores between male and female subject translations was also negative, demonstrating that the typical bias of MT systems being more accurate on male subject sentences has been reversed somewhat, but not to a major extent.

There is still room for improvement however, as the male to female translation ratio is still slightly off from 1, despite the WinoMT test set being very nearly equally gender balanced [5]. In addition, the phrases being translated in this assignment are unrealistically simple for everyday life. For example, "The owner had a dispute with the designer because she didn't like the design" is one of the sentences. This is simple to translate (excluding gender details) with the current SOTA systems, and only includes one gendered term (assigned to the owner). This task would likely be more of a challenge if we were to consider longer and more complex sentences with more gendered terms in them that are more representative of real natural language.

The small size of the WinoMT data set also presents its own complications. For example, compared to the number of professions that exist, the number addressed by the data-set is far from exhaustive. This means that there is a performance bias when a model is both fine-tuned and tested on WinoMT: the model may be able to learn the un-biased gender forms for the covered subset of professions well and give high gender accuracy statistics, but it will then go on to perform more poorly in domains not covered by the data-set, meaning that the gender statistics are artificially inflated.

Therefore, whilst this approach shows a step in the right direction for gender debiasing of MT systems whilst retaining translation accuracy, in order to be fully utile in real-world scenarios, the systems need to be able to handle more complex translation tasks over more contexts than those presented in WinoMT.

Finally, if the de-biasing task were to be expanded form English-German to English-Spanish or English-French also, then we would require similar structures to those used in this practical, such as the word-to-class mappings, word-to-BPE mappings, and respective symbol files assigning every string in the translations and mapping files to a unique positive integer. One extra complication when moving to French or Spanish translations is that, unlike German, plurals also have gendered nouns and pronouns, and so their mappings would also need to be accounted for in the dict files.

References

[1] Kishore Papineni et al. "BLEU: A Method for Automatic Evaluation of Machine Translation". In: *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*. ACL '02. Philadelphia, Pennsylvania: Association for Computational Linguistics, 2002, pp. 311–318. DOI: 10.3115/1073083.1073135. URL: https://doi.org/10.3115/1073083.1073135.

[2] Mehryar Mohri. "Weighted Finite-State Transducer Algorithms: An Overview". In: *AT&T Labs - Research* (2004).

[3] Ondřej Bojar et al. "Findings of the 2018 Conference on Machine Translation (WMT18)". English. In: *Proceedings of the Third Conference on Machine Translation (WMT), Volume 2: Shared Task Papers*. EMNLP 2018 Third Conference on Machine Translation (WMT18), WMT18 ; Conference date: 31-10-2018 Through 01-11-2018. Association for Computational Linguistics, Oct. 2018, pp. 272–307. DOI: 10.18653/v1/W18-6401. URL: http://www.statmt.org/wmt18/.

[4] Jeffrey Dastin. *Amazon scraps AI recruiting tool that showed bias against women*. Oct. 2018. URL: https://www.reuters.com/article/us-amazon-com-jobs-automation-insight/amazon-scraps-secret-ai-recruiting-tool-that-showed-bias-against-women-idUSKCN1MK08G.

[5] Gabriel Stanovsky, Noah A. Smith, and Luke Zettlemoyer. "Evaluating Gender Bias in Machine Translation". In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, July 2019, pp. 1679–1684. DOI: 10.18653/v1/P19-1164. URL: https://aclanthology.org/P19-1164.

[6] Brian Thompson et al. "Overcoming Catastrophic Forgetting During Domain Adaptation of Neural Machine Translation". In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, June 2019, pp. 2062–2068. DOI: 10.18653/v1/N19-1209. URL: https://aclanthology.org/N19-1209.

[7] Danielle Saunders and Bill Byrne. "Reducing gender bias in neural machine translation as a domain adaptation problem". In: *arXiv preprint arXiv:2004.04498* (2020).

[8] Danielle Saunders, Rosie Sallis, and Bill Byrne. "Neural Machine Translation Doesn't Translate Gender Coreference Right Unless You Make It". In: *arXiv preprint arXiv:2010.05332* (2020).