Technical Disclosure Commons

Defensive Publications Series

October 08, 2018

Gender-Aware Natural Language Translation

James Kuczmarski

Melvin Johnson

Gender-aware natural language translation

ABSTRACT

Languages differ in the way they express gender. Some languages are gender-neutral, while others have masculine, feminine, or neuter forms. When translating gender-neutral text into a language with gender, there can be more than one valid translation. For example, the Turkish query “O bir doktor” is translated into English as either “He is a doctor” or “She is a doctor.” Historically, machine translation has not been able to generate more than one gender-specific translation. Put differently, machine translation would “choose” a single masculine form, feminine form, or other gender variant as the translation output. In neural machine translation, this “choice” reflects bias in the training data used to train the translation model.

This disclosure presents techniques to generate both a masculine and a feminine translation for gender-neutral text, thereby reducing or eliminating gender bias in machine translation. The technique includes three main components: detection, generation of alternatives, and validation. In the detection phase, it is detected whether a given query is gender-ambiguous or not. If the detection component triggers, two translations for the query are generated: one masculine and one feminine. Finally, the validation step verifies that the two translations are high quality before showing them to users.

KEYWORDS

- Machine translation
- Neural translation
- Algorithmic bias
- Bias removal
- Gender bias

- Model training
- Training data
- Natural language processing

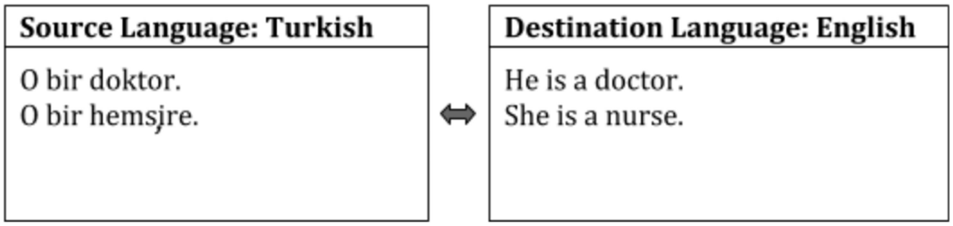
BACKGROUND

Languages differ in the way they express gender. Some languages are gender-neutral, while others have masculine, feminine, or neutral forms. When translating gender-neutral text into a language with gender, there can be more than one valid translation. For example, the Turkish query "O bir doktor" is translated into English as either "He is a doctor" or "She is a doctor." Similar rules exist in many other languages.

Gender in training data	Percentage of training pairs
Male-only examples	1.34
Female-only examples	0.48
Both male & female	0.08
Neither male nor female	98.10

Table 1: Gender statistics for Turkish-to-English data used to train a neural machine translation model

Table 1 shows gender statistics for an example Turkish-to-English corpus used to train a neural machine translation model. The number of gendered phrases is 1.34+0.48+0.08=1.9% of the total corpus. Of this set of gendered phrases, the number of masculine phrases is nearly thrice the number of feminine phrases. Bias in the training data leads to a bias in translations produced by machine-translation systems, as illustrated in Fig. 1 below.



Source Language: Turkish		Destination Language: English
O bir doktor. O bir hemsjre.	⇔	He is a doctor. She is a nurse.

Fig. 1: Gender-biased machine translations due to bias in the training set

In the example illustrated in Fig. 1, the machine-translation system produced translations in which “doctor” corresponded to male gender, while “nurse” corresponded to female gender.

DESCRIPTION

An overview diagram of the system to provide gendered translations is in Fig. 2 below. From the figure, it can be seen that input query “Q” from the user passes through three main components before fulfilling the user’s request. The three components are described briefly below:

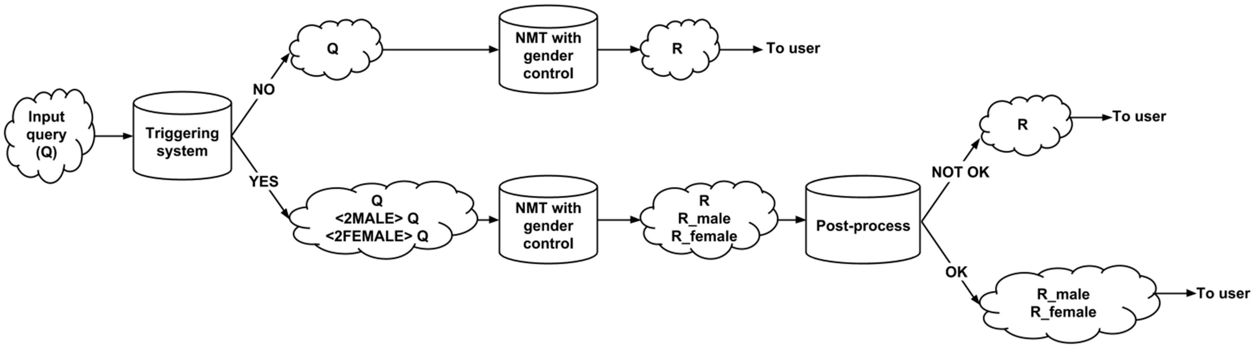


Fig. 2: Overview of system to provide gendered translations

Triggering system: The triggering system triggers when an input query is eligible for gendered translations. If this system triggers on an input query Q, three requests (one male, indicated by preceding token “<2MALE>”; one female, indicated by preceding token “<2FEMALE>”; and one ungendered, indicated by no preceding token) are sent to the next system as shown (“YES” branch in Fig. 2). If this system does not trigger on a query, only one request Q is sent to the next system (“NO” branch in Fig. 2).

Back-end with NMT and gender control: The back-end is responsible for providing the requested translations. This component receives from the triggering system requests that are either gendered (e.g., preceded by gender tokens) or ungendered. Depending on the kind of request, the back-end produces gendered responses (“R_male”, “R_female”, in Fig. 2) and/or an ungendered response (“R” in Fig. 2). Each of the translations can be generated by either a lexicon service or a neural machine translation (NMT) decoder. Both the lexicon and the NMT decoder are able to surface gender-specific responses if requested. If the response is for a query that was not eligible for gendered translation (“NO” branch of Fig. 2), then the response is displayed to the user without any processing or filtering, as shown in the upper branch of Fig. 2.

Post-process / filter step: This is the post-processing step that analyzes the male and female translations generated by the previous step and determines whether to show the gendered translations to users. Specifically, if the male and female translations satisfy certain established criteria (“OK” branch in Fig. 2), the system approves them for display; if not (“NOT OK” branch in Fig. 2), the system decides to show only the conventional, e.g., gender-biased, translation.

Each of the above components is explained in greater detail below.

<u>Details of the triggering system</u>

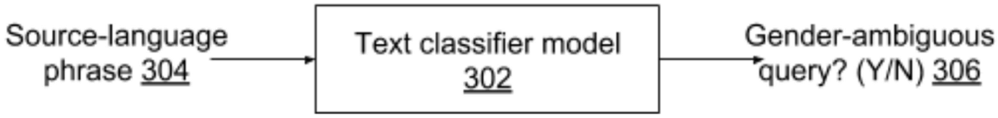

Source-language phrase 304
Text classifier model 302
Gender-ambiguous query? (Y/N) 306

Fig. 3: Triggering system: A text-classifier model trained to detect gender-ambiguous query phrases

Fig. 3 illustrates training of the triggering system, per techniques of this disclosure. The triggering system is a machine-learned text-classifier model (302) that detects gender ambiguity (306) in input query phrases (304). Such a text-classifier model is trained using positive and negative training examples. Positive training examples have gender ambiguity. Negative training examples do not have gender ambiguity.

Phrases with gender ambiguity (i.e., positive training examples) and phrases without gender ambiguity (i.e., negative training examples) in a corpus of potential training examples can be identified in a variety of ways, including via human raters or by detecting words in the source or target language that correspond to a specific gender. Seemingly small text changes in a string of text can render a training example negative or positive. For instance, in Turkish, “Ondan” can translate to “from him” or “from her” and is a positive training example, while “On-da” translates to “at 10” and is a negative training example.

Details of the back-end with NMT and gender control

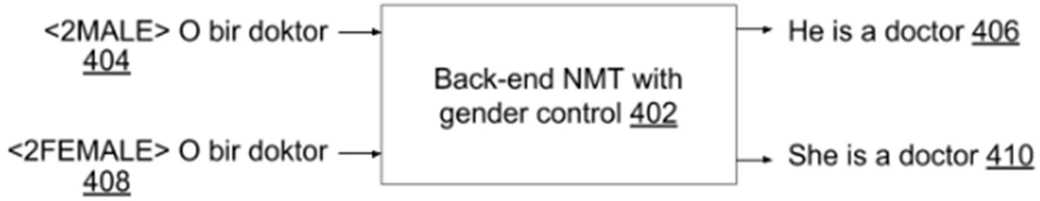


Fig. 4: Back-end NMT with gender control: Training a machine translator to produce output based on a gender tag

Fig. 4 illustrates training of the back-end NMT model (402) such that the gender of the output translation can be controlled. Source-language phrases tagged by a masculine token (404) can produce a male-gendered phrase in the destination language (406). Source-language phrases tagged by a feminine token (408) can produce a female-gendered phrase in the destination language (410).

Training a machine translation model that produces male and female translations involves several steps. First, masculine vs. feminine vs. ungendered training data are identified. For example, in the case of a Turkish-to-English translation model, this is accomplished by identifying personal pronoun words (e.g., she, herself, hers, her, he, himself, his, him) or other gender-specific words (e.g., mother, father) in the English training data that correspond to a specific gender.

Based on the presence of those words, the gender of the training data is determined and the training data is labeled appropriately as male, female, both or neither (these categories appear in Table 1.) To label male vs. female training data, a text tag is applied to the training data. To label male training data, a <2MALE> tag is added. To label female training data, a <2FEMALE> tag is added.

- Example: <2FEMALE> O bir doktor. → She is a doctor.
- Example: <2MALE> O bir doktor. → He is a doctor.

Then, the tagged data is used to train a machine translation model. To account for differences in the proportions of male vs. female training data (see Table 1) in this training, the female training data is oversampled relative to the male data. This helps ensure that the translation model can reliably generate both a masculine and feminine translation. Without the oversampling, a model using training data from Table 1 may be able to generate a masculine translation more reliably than a feminine translation.

Finally, the trained machine translation engine can produce a masculine translation, feminine translation, or non-gendered translation based on using the relevant tagged training data when generating the translation (i.e., training data with <2MALE> tag produces male translation; training data with <2FEMALE> tag produces female translation; training data with no tag produces non-gendered translation). Based on whether the triggering system detects gender-ambiguity in the query or not, the translation engine choose which gendered translation types to request and generate (e.g., non-gendered translation only; masculine, feminine, and non-gendered translation; etc.).

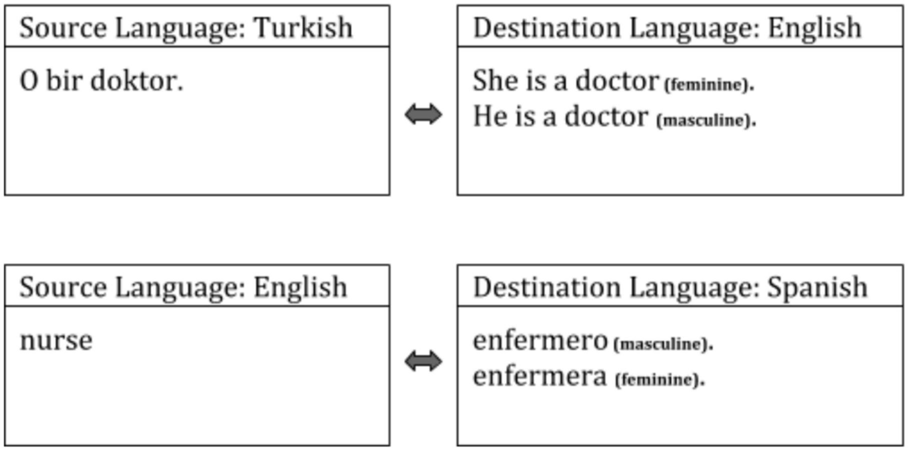
Ultimately, gendered translations can be generated by either a lexicon service or a machine translation decoder. In the system design per this disclosure, both the lexicon and the machine translation decoder are able to surface gender-specific responses if requested.

Details of the post-process / filter step

The translations are validated in the post-processing or filtering step as follows. Of the translations, if the male-gendered and female-gendered translations *differ in any aspect other than gender*, the validation test fails. If the only differences between the male-gendered and female-gendered translations are gender-related, then the validation test passes.

- *Example*: The male-gendered translation is “Yuan, did he really say those words?” The female-gendered translation is “Yuan, did she actually say those words?” In this case, in addition to gender-specific differences (he vs. she), the two translations have another difference not related to gender (really vs. actually). Therefore, the validation test fails.
- *Example*: The male-gendered translation is “He wants to make everything his own.” The female-gendered translation is “She wants to make everything her own.” In this case, the only differences between the two translations are related to gender (he, his vs. she, her). Therefore, the validation test passes.

If the validation test passes, both male-gendered and female-gendered translations are provided to the user. If the validation test fails, the conventional, e.g., gender-biased, translation is provided to the user.



Source Language: Turkish

O bir doktor.

Destination Language: English

She is a doctor (feminine).
He is a doctor (masculine).

Source Language: English

nurse

Destination Language: Spanish

enfermero (masculine).
enfermera (feminine).

Fig. 5: Multiple, valid, gender-sensitive translations are obtained in the destination language

Fig. 5 illustrates example translations obtained per techniques of this disclosure. As shown, a gender-ambiguous phrase in a source language is translated into two valid, gender-specific translations in the target language.

CONCLUSION

This disclosure presents techniques to generate both a masculine and a feminine translation for gender-neutral text, thereby reducing or eliminating gender bias in machine translation. The technique includes three main components: detection, generation of alternatives, and validation. In the detection phase, it is detected whether a given query is gender-ambiguous or not. If the detection component triggers, two translations for the query are generated: one masculine and one feminine. Finally, the validation step verifies that the two translations are high quality before showing them to users.