A simple approach to remove bias from text

Brit Youngmann
Tel Aviv University
brity@mail.tau.ac.il

1 INTRODUCTION

Bias is prejudice in favor of or against one thing, person, or a group, compared with another, usually in a way considered to be unfair [2]. People may develop biases toward or against an individual, an ethnic group, a gender identity, etc. Bias can come in many forms and is related to prejudice and intuition, what makes it hard to quantify.

Recent work has been shown to exhibit various biases within algorithms, such as racial discrimination and gender bias in online advertisement [4, 9], or indirect racial biases in algorithms predicting future criminals [1]. Several works have identified the bias present in word embedding [3, 8], providing a methodology for modifying an embedding to remove gender stereotypes, while maintaining desired associations between words.

In this work we propose a method to eliminate bias from sentences, while preserving their sentiment and content. As a first step, we are focusing on removing gender stereotypes form text. More concretely, we provide a framework that, given a sentence describing a person and a target gender, generates a new sentence that has similar content and sentiment, yet the gender of the person is altered to the desired attribute. To illustrate the problem this work tackles, consider the following motivating example.

Example 1.1. Consider the following movie review:

"Wonder Woman is a great movie. The story is interesting and Gal Gadot is beautiful. Go see it!"

Here, even without knowing the gender of the movie’s main actress Gadot, the word “beautiful” serves as a strong clue that Gadot is a female actress, since it used more often to describe women than men (according to [3]). Therefore, in order to eliminate all clues of Gadot’s gender from this sentence, one must replace this word with an appropriate positive-sentiment word (as the sentiment of this sentence is positive), that has similar meaning. One possible output is to use the word “awesome”, since this word is considered to be gender-neutral and it is semantically close (according to [3]).

The example above demonstrates that gender stereotypes go beyond simple gender specific words like “he” or “she” (a phenomena that was also demonstrated by [3]). Therefore, our goal is to identify and replace both the explicit (e.g., “she”) and the implicit (e.g., “beautiful”) words that may suggest the gender of the person appear in a given sentence, producing a new sentence with similar content and the same sentiment, revealing the desired target-gender of the person. For example, continuing with our running example, given the requirement to change Gadot’s gender from female to male, one possible output is to use the word “brilliant” instead of the word “beautiful”, since this word is more often associated with males (according to [3]).

In summary, this work serves the goal of eliminating or altering gender stereotypes from text. One motivation to do so is in the process of examining potential candidates’ reviews for a job opening (a process that is currently done in large corporations such as Microsoft.). Consider, for example, a high-tech company searching for new employees. As part of the process, the person in charge reads multiple reviews written by several interviewers, regarding different candidates. In this case, to avoid any kind of gender discrimination, the company may use our proposed method to transform all reviews to be gender neutral, i.e., to ensure all reviews would not reveal the candidate’s gender. This action is one step in the process of ensuring that the final choice of a candidate is based only on professional considerations.

This work’s objective is challenging from the following reasons: (i) First, in order to preserve the sentiment of the sentence, the model should capture the words associated with the sentence’s sentiment (e.g., “great”, “interesting”) (ii) Similarly, to preserve the sentence’s meaning, one must understand its content (e.g., a positive review on the Wonder Women movie and in particular on its main actress Gal Gadot); (iii) Last, the proposed method should strip the sentence of the gender marking words/phrases which give away the gender (e.g., in this case, the word “beautiful”), replacing them with the corresponding words/phrases (e.g., the words “awesome” and “brilliant”).

We propose a simple method, based on a previous work presented in [7], motivated by the observation that text attributes are often easy to identify. In essence, our method extracts content words by deleting phrases associated with the sentence’s original gender attribute, retrieves new phrases associated with a target gender (i.e., male, female or gender-neutral) which also posses the same sentiment as the deleted n-grams, and uses a sequence-to-sequence model to combine both into the output sentence. As opposed to the method provided in [7], we are also using an external source of word embedding [3] to identify the best words to insert. In our experimental study over real-life sentences we have collected, we demonstrate the advantages of our approach.

While our exposition focuses on altering gender attributes of sentences, we emphasize that the proposed system is a general-purpose system applicable to other attributes such as race or nationality as well (see the Discussion section).

2 PROBLEM FORMULATION

As mentioned, in this work we consider the task of removing bias from text documents. More formally, given a sentence describing a person, a property defining a subpopulation to be altered, e.g., race or gender, and the target property (e.g., gender-neutral, male), our goal is to generate a new sentence that maintains the same meaning and sentiment as the original sentence, and alters the text so the person described is now exhibiting the target property.

Extending the setting in [7], we assume access to a corpus of labeled sentences $D = (x_1, s_1, v_1), \ldots, (x_m, s_m, v_m)$, where for every $1 \leq i \leq m$, $x_i$ is an English sentence describing a person, $s_i$ is the sentence’s sentiment (i.e., "positive", "negative" or "neutral")
and \( v_i \in V \) is the set of possible attributes. As mentioned, in this work we are only focusing in the gender-setting, where \( V = \{ \text{"male"}, \text{"female"}, \text{"neutral"} \} \).

Our goal is to devise a model that takes as an input a triple \((x, s, v)\) and the target attribute \(v'\), where \(x\) is the input sentence with the sentiment \(s\), exhibiting the attribute \(v\), and outputs a new sentence \(x'\) that retains the content and sentiment of \(x\) and exhibits the target attribute \(v'\).

### 3 TECHNICAL BACKGROUND

As mentioned, our work is based on a previous work [7] which tackled the Text attribute transformation task; i.e., the task of transforming a given sentence to alter a specific attribute (e.g., sentiment) while preserving its attribute-independent content, e.g., changing the sentence “the chicken was delicious to “the chicken was horrible” As we demonstrate in our experimental study, this approach is not well suited for our task, as it does not ensure that the sentiment is preserved and is limited to a given dataset of sentences.

In a nutshell, the key observation underlying the work of [7] is that in order to determine, e.g., which negative sentiment words to insert, one needs to consider the remaining content words which provide strong cues. For example, given the sentence “The chicken was...”, one can infer that a taste-related word like "bland" fits, while a word like ‘rude’ does not, despite the fact that both words have negative sentiment. Namely, while the deleted sentiment words may contain non-sentiment information (e.g., “delicious”), this information can also be inferred from other words in the sentence, i.e., the sentence’s content words (e.g., “the chicken”).

The authors of [7] have first defined the following concepts:

**Attribute markers n-grams**: words or phrases in the sentence that are indicative of a particular attribute (e.g., the sentiment-related words).

**Content words**: all attribute-independent words in the sentence (e.g., what properties of a restaurant are being discussed in the sentence – in the example above, the chicken).

Then, given a sentence \(x\), its attribute \(v\) and the desired attribute \(v'\), they proposed to use the following steps to generate a new sentence \(x'\) exhibiting the property \(v'\):

1. **Delete**. Given the sentence \(x\), separate the words in \(x\) into a set of attribute markers \(a(x, v)\) and a sequence of content words \(c(x, v)\).
2. **Retrieve**. Retrieve a sentence \(x'\) that has the target attribute \(v'\) and whose content, \(c(x', v')\), is similar to that of \(x\).
3. **Generate**. Given the content of \(x\), \(c(x, v)\), the target attribute \(v'\), and the retrieved sentence \(x'\), generate a sentence \(x'\) using a neural sequence-to-sequence model that learns to stitch two sequences together. That is, the model stitches between \(a(x', v')\) and \(c(x, v)\).

As we explain in the next section, while we use some of these implemented components in our framework, we enrich this framework to also use an external source of word embedding and to ensure the outputted sentence maintains the same sentiment of the input.

### 4 SYSTEM OVERVIEW

As mentioned in the Introduction, in this work we focus on altering gender stereotypes from sentences. That is, we assume that the attribute the user wishes to alter is the gender (i.e., “male”, “female” or “neutral”). Additionally, we assume that all sentences in our corpus are describing a person.

**Useful notations.** Let \(g(x)\) denote the vector of all gender attribute n-grams of an input sentence \(x\), where \(g_i(x)\) denotes the i-th such n-gram and \(1 \leq i \leq |g(x)|\). Similarly to [7], in our setting as well, \(c(x)\) denotes the sentence \(x\) after removing all gender attribute n-grams from it. Additionally, let \(s(x)\) denote the sentiment of the sentence \(x\) and \(a(x)\) denotes the inferred gender of the person mentioned in \(x\), i.e., \(a(x) \in \{\text{"male"}, \text{"female"}, \text{"neutral"}\}\).

The main components of our system are depicted in Figure 1. We note that while the main idea is similar to the one proposed in [7], our implementation of the delete and retrieve components is novel. In short, our system works as follows.

Given an input sentence \(x\), its sentiment \(s\), its current attribute \(v\) and the target attribute \(v'\), our system:

1. First extract and delete from \(x\) its gender attribute n-grams \(g(x)\), retaining with the content of \(x\), denoted as \(c(x)\).
2. Next, given the gender attribute vector \(g(x)\) and the target attribute \(v'\), for every word \(g_i(x)\) in \(g(x)\), we retrieve the corresponding n-gram that exhibits the desired attribute \(v'\), constructing the gender attribute vector \(g'(x)\).
3. Last, using a neural sequence-to-sequence model, the system outputs a sentence \(x'\) s.t. (1) \(c(x)\) is similar to \(c(x')\), i.e., the contents of the original and generated sentences are similar; (2) \(s(x') = s(x')\), i.e., the sentiment of the two sentences is identical, and (3) \(x'\) is exhibiting the gender attribute \(v'\).
We consider several alternative solutions as baselines, including all words. Specifically, using their gender-refinements. As was done in [7], to identify and delete the gender attribute n-grams in the given sentence \(x\), as well as to extract a sequence of the content words from \(x\), denoted as \(c(x)\), we employ a simple method that identifies n-grams that can be viewed as discriminative features for a Naive Bayes classifier.

Formally, we consider an n-gram \(u\) as a gender-attribute sequence with respect to a gender \(v\) if:

\[
\frac{\text{Count}(u, D_v) + \lambda}{(\sum_{v' \in V} \text{Count}(u, D_{v'}) + \lambda)} \geq \theta
\]

where \(D_v\) denotes the set of sentences in the corpus with the attribute \(v\), \(\text{Count}(u, D_v)\) is the number of times \(u\) appears in \(D_v\), \(\lambda\) is a smoothing parameter and \(\theta\) is a predefined threshold parameter.

To identify further gender attributes words, in particular words which may reveal the gender and may are not well reflected in our corpus, we use the implementation of [3] provided in https://github.com/tolga-b/debiaswe. Specifically, using their gender-specific words, we also consider all these words as gender attributes words.

We note that this addition of using an external source ensures that our approach would better recognize all gender attribute n-grams, regardless the given corpus of sentences.

**Retrieve.** To retrieve the corresponding gender attribute words for the target attribute \(v'\), here again we use the implementation of [3]. This implementation further provides a list of 1,441 gender-specific words. We use this corpus to identify the words-to-be-replaced, as well as the learned analog-pairs. For words in \(g(x)\) that the model does not have an analog word, or, alternatively, to retrieve a gender-neutral word, we simply use the words vector and the gender-direction vector (i.e., we reduce the gender dimension, possibly adding the desired female or male direction).

Importantly, in this phase we also ensure that the sentiment of the sentence is unchanged, i.e., that for every word in \(g(x)\) it corresponding word possesses (nearly) the same sentiment. Formally, given a word \(g_i(x)\), we retrieve \(g'_i(x)\) according to:

\[
g'_i(x) = \text{argmin}_{w \in W} (d(g_i(x)−v_{\text{gender}}+v'), w)+\text{sent}(g_i(x))−\text{sent}(w))
\]

where \(W\) is the collection of words vectors, \(d\) is a distance metric comparing two vectors, \(v_{\text{gender}}\) is the gender direction vector (as was computed in the implementation of [3]), \(v'\) is the target gender, and \(\text{sent}(\cdot) : W \rightarrow [-1,1]\) is a given function determining the sentiment of a word.

As was done in [7], here again we use the Euclidean distance as the metric function, and used the nltk implementation of [5] as the sentiment function (https://www.nltk.org/_modules/nltk/sentiment/vader.html).

**Generate.** Given the content of \(x\), \(c(x)\), the retrieved gender attribute words \(g'_i(x)\), we used the sequence-to-sequence model of [7] that learns to stitch the two sequences together, obtaining the output sentence \(x'\).

As we present in the next section, in our experimental study we consider several alternative solutions as baselines, including all variants proposed in [7], demonstrating the effectiveness of our refinements.

### 5 EXPERIMENTAL RESULTS

#### 5.1 Experimental setup

We first present the dataset we constructed, then the baselines used throughout the experimental study. Last, we explain how the obtained results were evaluated.

**Dataset.** As opposed to the text attribute transformation task, where several benchmark datasets exist (as was used in [7]), to the extent of our knowledge, there are no such ones for the gender-transformation task which we study in this work. Therefore, we constructed our own repository as follows.

First, we collected all tweets from Twitter that were written about 12 American celebrities\(^1\), during a period of 10 days. We then applied several simple operations for the sake of data cleaning, such as removing hash tags and non English words/phrases, retaining only with simple English sentences where the person appears at the beginning of the sentence (e.g., "Donald Trump is a...").

Next, each sentence was associated with a tag according to a pre-trained classifier that encodes each sentence into a vector using a bidirectional LSTM with an average pooling layer over the outputs. We then trained the classifier to identify the gender by minimizing the logistic loss (this classifier is based on a similar one proposed in [7]). As for determining the target attribute, for each sentence we randomly picked a different gender-tag. Additionally, each sentence was associated with a sentiment score (again, using the implementation of [5] as provided by nltk).

This dataset contains approximately 6k labeled sentences.

**Baseline algorithms.** We compared our system to the following baselines:

1. **Retrieve-only**, which simply retrieves the retrieved sentence \(x^{tgt}\).
2. **Template-based**, which replaces the attribute markers deleted from the source sentence with those of the retrieved sentence \(x^{tgt}\).
3. **Delete-only**, using a RNN decoder, it attempts to produce words indicative of the source content and the target attribute, while remaining fluent.
4. **Delete-and-retrieve** is similar to **Delete-only**, but uses the attribute markers of the retrieved sentence \(x^{tgt}\) and the content of the source sentence \(x\).

#### 5.2 Quality evaluation

To properly quantify the results of each approach, we have evaluated four main aspects of the problem:

\(^1\)The celebrities considered are: 'Katy Perry', 'Justin Bieber', 'Rihanna', 'Taylor Swift', 'Cristiano Ronaldo', 'Ellen DeGeneres', 'Jimmy Fallon', 'Gal Gadot', 'Hillary Clinton', 'Angela Merkel', 'Barack Obama' and 'Donald Trump'.

---

3
Our approach & 0.62 & 0.72 & 0.31 & 10.1 \\
Our approach using templates & 0.58 & 0.72 & 0.34 & 10.3 \\
RETRIEVE-ONLY & 1.0 & 1.0 & 0.53 & 0.7 \\
TEMPLATE-BASED & 0.60 & 0.69 & 0.47 & 10.2 \\
DELETE-ONLY & 0.61 & 0.68 & 0.45 & 8.3 \\
DELETE-AND-RETRIEVE & 0.62 & 0.67 & 0.44 & 9.8 \\

Table 1: Experimental study results summary.

<table>
<thead>
<tr>
<th>Baseline</th>
<th>Sentence</th>
<th>Attribute</th>
<th>Sentiment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Katy Perry is so cute! She is the most sexy women in the world</td>
<td>Female</td>
<td>1</td>
</tr>
<tr>
<td>Our approach</td>
<td>Katy Perry is so clever! He is the most nerdy man in the world</td>
<td>Male</td>
<td>1</td>
</tr>
<tr>
<td>RETRIEVE-ONLY</td>
<td>Justin Bieber is the most handsome singer in the world, he is perfect!</td>
<td>Male</td>
<td>1</td>
</tr>
<tr>
<td>TEMPLATE-BASED</td>
<td>Katy Perry is so cute! He is the most god in the world</td>
<td>Neutral</td>
<td>0</td>
</tr>
<tr>
<td>DELETE-ONLY</td>
<td>Katy Perry is so cute! He is a god</td>
<td>Male</td>
<td>1</td>
</tr>
<tr>
<td>DELETE-AND-RETRIEVE</td>
<td>Katy Perry is so cute! He is so handsome</td>
<td>Male</td>
<td>1</td>
</tr>
<tr>
<td>Source</td>
<td>Taylor Swift is a bitch and a horrible dancer</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>Our approach</td>
<td>Taylor Swift is a mean and a horrible singer</td>
<td>Neutral</td>
<td>−1</td>
</tr>
<tr>
<td>RETRIEVE-ONLY</td>
<td>Angela Merkel is a lier and the worst politician in Europe today</td>
<td>Neutral</td>
<td>−1</td>
</tr>
<tr>
<td>TEMPLATE-BASED</td>
<td>Taylor Swift is a lier and a horrible dancer</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>DELETE-ONLY</td>
<td>Taylor Swift is a stupid and a horrible dancer</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>DELETE-AND-RETRIEVE</td>
<td>Taylor Swift is a lier and a horrible dancer</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>Source</td>
<td>Jimmy Fallon is not funny. He is not a good actor and he is a racist.</td>
<td>Male</td>
<td>−1</td>
</tr>
<tr>
<td>Our approach</td>
<td>Jimmy Fallon is not hysterical. She is not a good actress and she is a sexist.</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>RETRIEVE-ONLY</td>
<td>Jimmy Fallon is not hysterical. She is not a good actress and she is a sexist.</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>TEMPLATE-BASED</td>
<td>Ellen DeGeneres is so funny. She is hilarious!</td>
<td>Female</td>
<td>1</td>
</tr>
<tr>
<td>DELETE-ONLY</td>
<td>Jimmy Fallon is not funny. She is not a good actress and she is a racist.</td>
<td>Female</td>
<td>−1</td>
</tr>
<tr>
<td>DELETE-AND-RETRIEVE</td>
<td>Jimmy Fallon is not funny. She is not a good actress and she is a racist.</td>
<td>Female</td>
<td>−1</td>
</tr>
</tbody>
</table>

Table 2: Example outputs. Attribute markers are colored (red for female, blue for male and green for neutral). In the first example the target attribute is male, in the second it is neutral and in the third it is female.

(1) **Grammar:** is the output of the examined approach a grammatically correct English sentence?

(2) **Attribute transformation:** is the output sentence exhibits the desired gender?

(3) **Preservation of sentiment:** is the output sentence similar to the source sentence in terms of their sentiment?

(4) **Preservation of content:** is the output sentence similar to the source sentence in terms of content?

Following previous work (e.g., [6, 7]), we are automatically evaluating the results of all baselines.²

Specifically, we have used the Language-Check module for Python to evaluate the grammar (https://pypi.org/project/language-check/). We define the **grammar score** as the fraction of output sentences classified as correct English sentences.

²Due to time limitation, we do not include in our excitement results collected by human evaluators, as we were unable to collect enough results.

We have trained a classifier to assess whether the baselines outputs have the desired attribute (we used the same model that classified the original sentences in the dataset). Here again, the **output classifier score** is defined as the fraction of outputs classified as having the target attribute.

To assess sentiment, we computed the averaged (absolute) differences between the sentiment score of the original and the outputted pair-sentences, using nltk implementation of [5].

Last, to quantify how well the content of the sentence was preserved, following [7], we computed the BLEU scores between the output and the original sentences, reporting here the averaged obtained score. In this case, the higher the BLEU score is, the better the system preserves the original sentence’s content, by retaining the same words from the source sentence.

The results are depicted in Table 1. As can be seen, none of the examined approaches outperformed all of the competitors in...
all accounts, yet, nonetheless, our proposed solution managed to achieve a (slight) advantage in terms of both transforming the attribute and preserving the sentiment of the original sentence. Examples of typical outputs are presented in Table 2.

First, as was demonstrated in [7], here as well we observe that compared with the templates-based approaches, solutions that use the RNN model to generate the sentences are performing better according to the obtained grammar scores. For example, as depicted in Table 2, the template-based approaches yielded ungrammatical sentences such as "Taylor Swift is a mean and." or "He is the most god in the world".

Regarding the classifier scores, not surprisingly, the RETRIEVE-ONLY approach achieved the perfect score of 1, as it returns only sentences possessing the desired attributes. However, as can be seen, this approach does not ensure that neither the sentiment or the content of the original sentence are preserved. For example, as can be seen in Table 2, the retrieved sentences may have completely different content (e.g., as in the second example).

The results indicate that our solution better transforms the sentences to the desired attribute, than the solution proposed in [7]. This stems from the fact that our solution, on top of using the naive Bayes classifier to identify attribute words, also uses an external corpus of word embedding. Furthermore, it replaces a word with the best possible option, whereas other approaches are only relying on the dataset at hand, and hence are limited to sentences with similar content.

Another strength of our approach is the preservation of sentiment. As the original work in [7] had focused only on preserving the content and altering the attribute, in many cases, the outputted sentences had changed the original sentences’ sentiment (see, for example, examples 1 and 3 in Table 2). In contrast, our approach ensures that the sentiment of each replaced word would be unchanged.

Last, in terms of content preservation, one can see that the BLEU scores our approach achieved, as well as the template-based solutions, is higher than the solutions proposed in [7]. This stems from the fact that our approach uses simple manipulations on the words’ vectors, that ensure the semantic meaning is unchanged. For example, our solutions replaced the word "bitch" with the semantically close word "mean", while other solutions chose the word "lier" instead.

6 DISCUSSION

Our work refines the method proposed in [7]. As demonstrated in our experimental study, our main contributions is the use of an external source to better identify the attribute markers, and replacing them with the corresponding words while preserving the original sentence’s content and sentiment. In contrast to their work, our approach does not rely solely on the dataset at hand, and hence can handle cases where sentences with similar content and/or sentiment are not to be found in the dataset.

We demonstrated how related work in word embedding can be used in the attribute altering task, enjoying both the advantages of the simple and elegant approach suggested in [7] and the insights of [3].

In summary, in this work we took the initial steps towards developing a general framework that serves for the goal of removing bias from sentences. We demonstrate our approach in one example domain, where the goal was to alter the gender of a person in a given sentence, while preserving the original sentence’s content and sentiment. As was discussed in the Introduction, our method could be useful in other domains such as removing racist stereotypes from text documents. In the future, we intend to include human raters evaluation in our experimental study and to examine the capabilities of the proposed method in other domains as well.

REFERENCES


