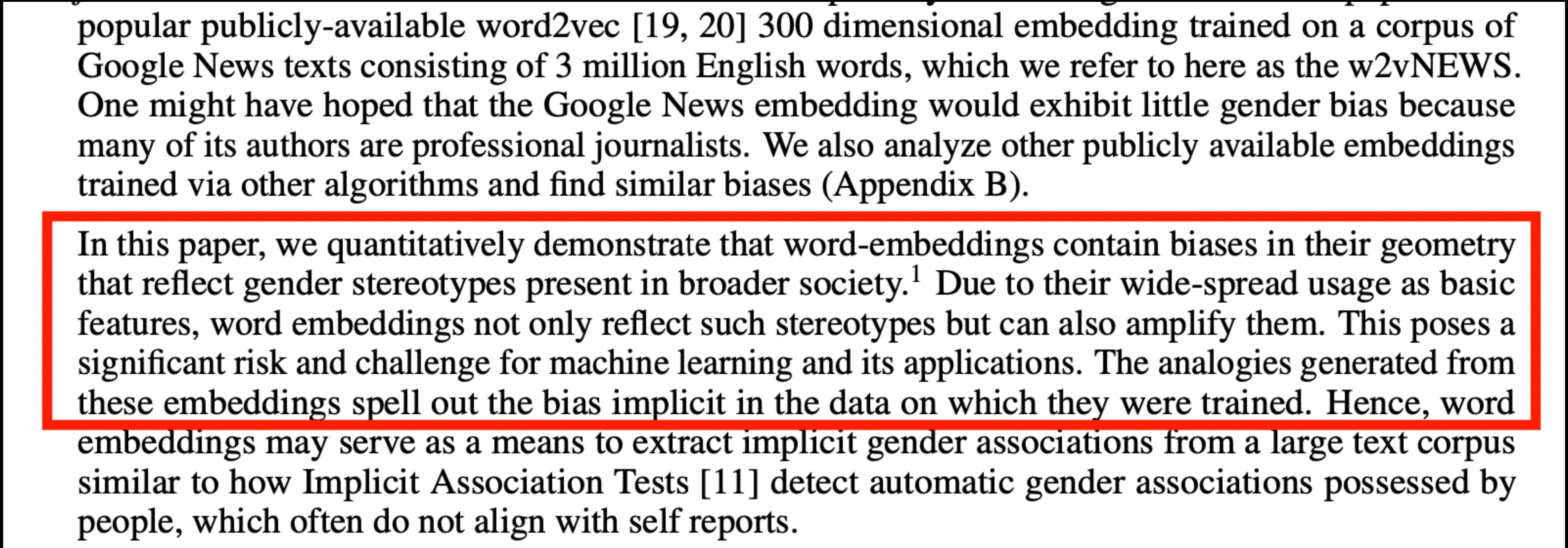
Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings

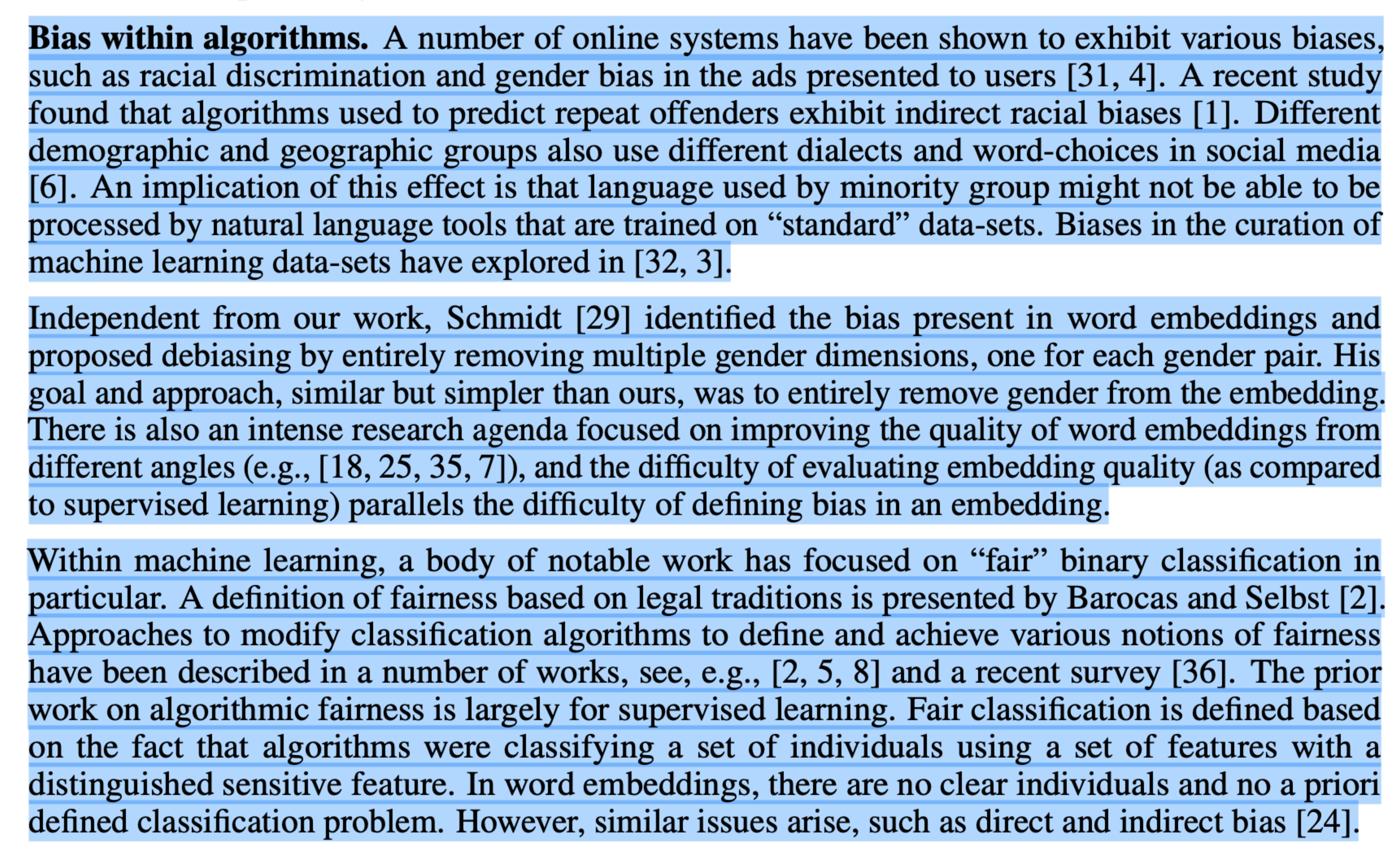
- https://papers.neurips.cc/paper/6228-man-is-to-computer-programmer-as-woman-is-to-homemaker-debiasing-word-embeddings.pdf

- 30th Conference on Neural Information Processing Systems (2016)



popular publicly-available word2vec [19, 20] 300 dimensional embedding trained on a corpus of Google News texts consisting of 3 million English words, which we refer to here as the w2vNEWS. One might have hoped that the Google News embedding would exhibit little gender bias because many of its authors are professional journalists. We also analyze other publicly available embeddings trained via other algorithms and find similar biases (Appendix B).

In this paper, we quantitatively demonstrate that word-embeddings contain biases in their geometry that reflect gender stereotypes present in broader society.[1] Due to their wide-spread usage as basic features, word embeddings not only reflect such stereotypes but can also amplify them. This poses a significant risk and challenge for machine learning and its applications. The analogies generated from these embeddings spell out the bias implicit in the data on which they were trained. Hence, word embeddings may serve as a means to extract implicit gender associations from a large text corpus similar to how Implicit Association Tests [11] detect automatic gender associations possessed by people, which often do not align with self reports.



Bias within algorithms. A number of online systems have been shown to exhibit various biases, such as racial discrimination and gender bias in the ads presented to users [31, 4]. A recent study found that algorithms used to predict repeat offenders exhibit indirect racial biases [1]. Different demographic and geographic groups also use different dialects and word-choices in social media [6]. An implication of this effect is that language used by minority group might not be able to be processed by natural language tools that are trained on “standard” data-sets. Biases in the curation of machine learning data-sets have explored in [32, 3].

Independent from our work, Schmidt [29] identified the bias present in word embeddings and proposed debiasing by entirely removing multiple gender dimensions, one for each gender pair. His goal and approach, similar but simpler than ours, was to entirely remove gender from the embedding. There is also an intense research agenda focused on improving the quality of word embeddings from different angles (e.g., [18, 25, 35, 7]), and the difficulty of evaluating embedding quality (as compared to supervised learning) parallels the difficulty of defining bias in an embedding.

Within machine learning, a body of notable work has focused on “fair” binary classification in particular. A definition of fairness based on legal traditions is presented by Barocas and Selbst [2]. Approaches to modify classification algorithms to define and achieve various notions of fairness have been described in a number of works, see, e.g., [2, 5, 8] and a recent survey [36]. The prior work on algorithmic fairness is largely for supervised learning. Fair classification is defined based on the fact that algorithms were classifying a set of individuals using a set of features with a distinguished sensitive feature. In word embeddings, there are no clear individuals and no a priori defined classification problem. However, similar issues arise, such as direct and indirect bias [24].

Google News Dataset

- https://papers.neurips.cc/paper/6228-man-is-to-computer-programmer-as-woman-is-to-homemaker-debiasing-word-embeddings.pdf

- 30th Conference on Neural Information Processing Systems (2016)

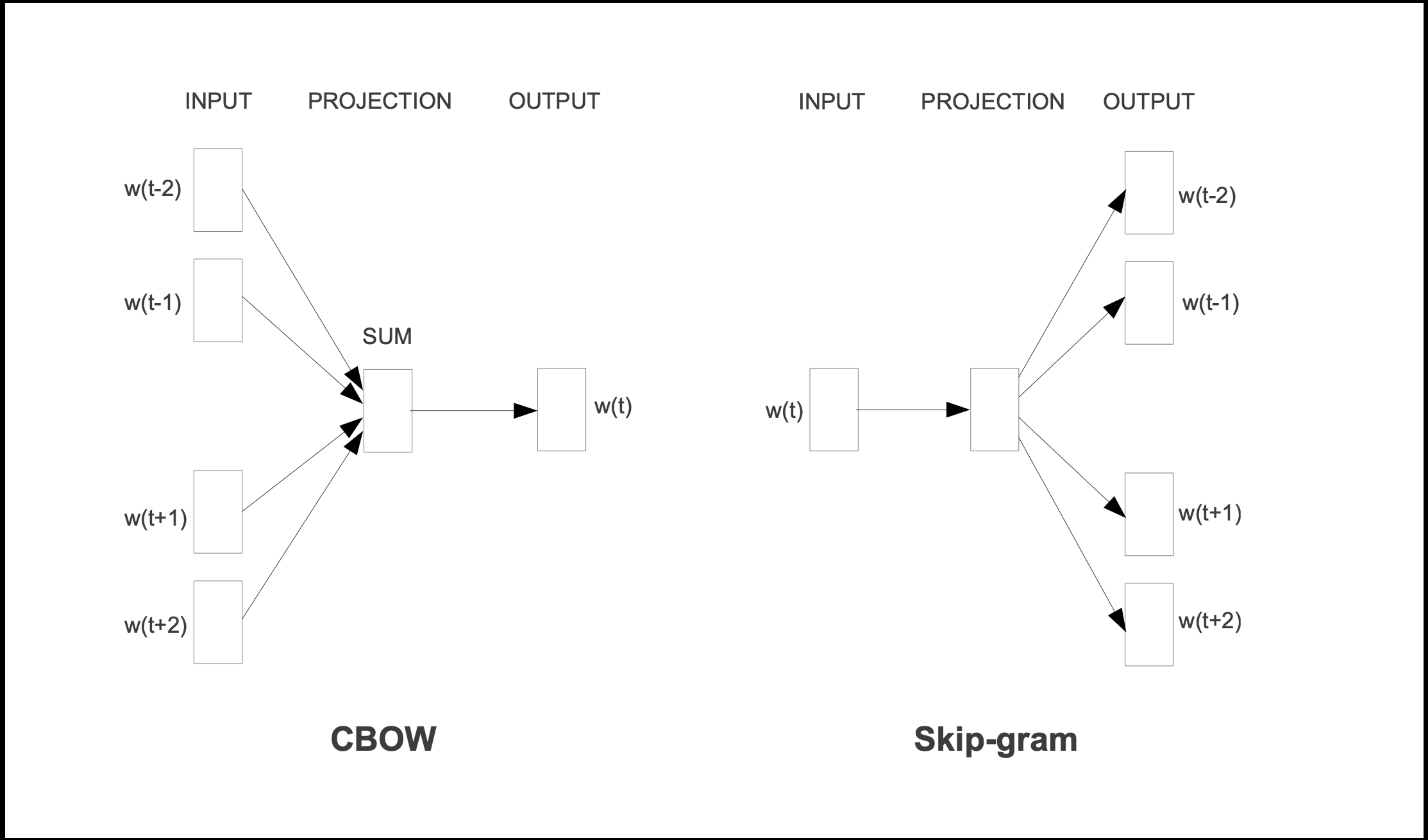
- https://code.google.com/archive/p/word2vec/

- "We are publishing pre-trained vectors trained on part of Google News dataset (about 100 billion words). The model contains 300-dimensional vectors for 3 million words and phrases. The phrases were obtained using a simple data-driven approach described in [2]. The archive is available here: GoogleNews-vectors-negative300.bin.gz."

2 strategies

- From context predict the middle (CBOW)

- From word predict context (Skip-gram)



GoogleNews vectors

- library(ggplot2)
- library(tidyverse)
- library(tidytext)
- library(wordVectors)
- model = read.vectors("GoogleNews-vectors-negative300.bin")
- model
- #A VectorSpaceModel object of 3000000 words and 300 vectors

X is to Y as A is to ???

- Paris is to France as Tokyo is to?
- I.e. France vector - Paris vector + Tokyo vector is close to what?
- model %>% closest_to(~ "france" - "paris" + "tokyo")
- #runtime is ~ 20 seconds
- Do we have this relationship: $\overrightarrow{France} - \overrightarrow{Paris} \approx \overrightarrow{Japan} - \overrightarrow{Tokyo}$

X is to Y as A is to ???

- Man is to King as Woman is to?
- I.e. King vector - Man vector + Woman vector is close to what?
- model %>% closest_to(~ "king" - "man" + "woman")
- #runtime is ~ 20 seconds
- Do we have this relationship: $\overrightarrow{man} - \overrightarrow{woman} \approx \overrightarrow{king} - \overrightarrow{queen}$

X is to Y as A is to ???

- Man is to Computer Programmer as Woman is to?
- I.e. Computer Programmer vector - Man vector + Woman vector is close to what?
- model %>% closest_to(~ "computer_programmer" - "man" + "woman")

Bias is implicit in the dataset

- model %>% closest_to(~ "business" - "he" + "she",20)

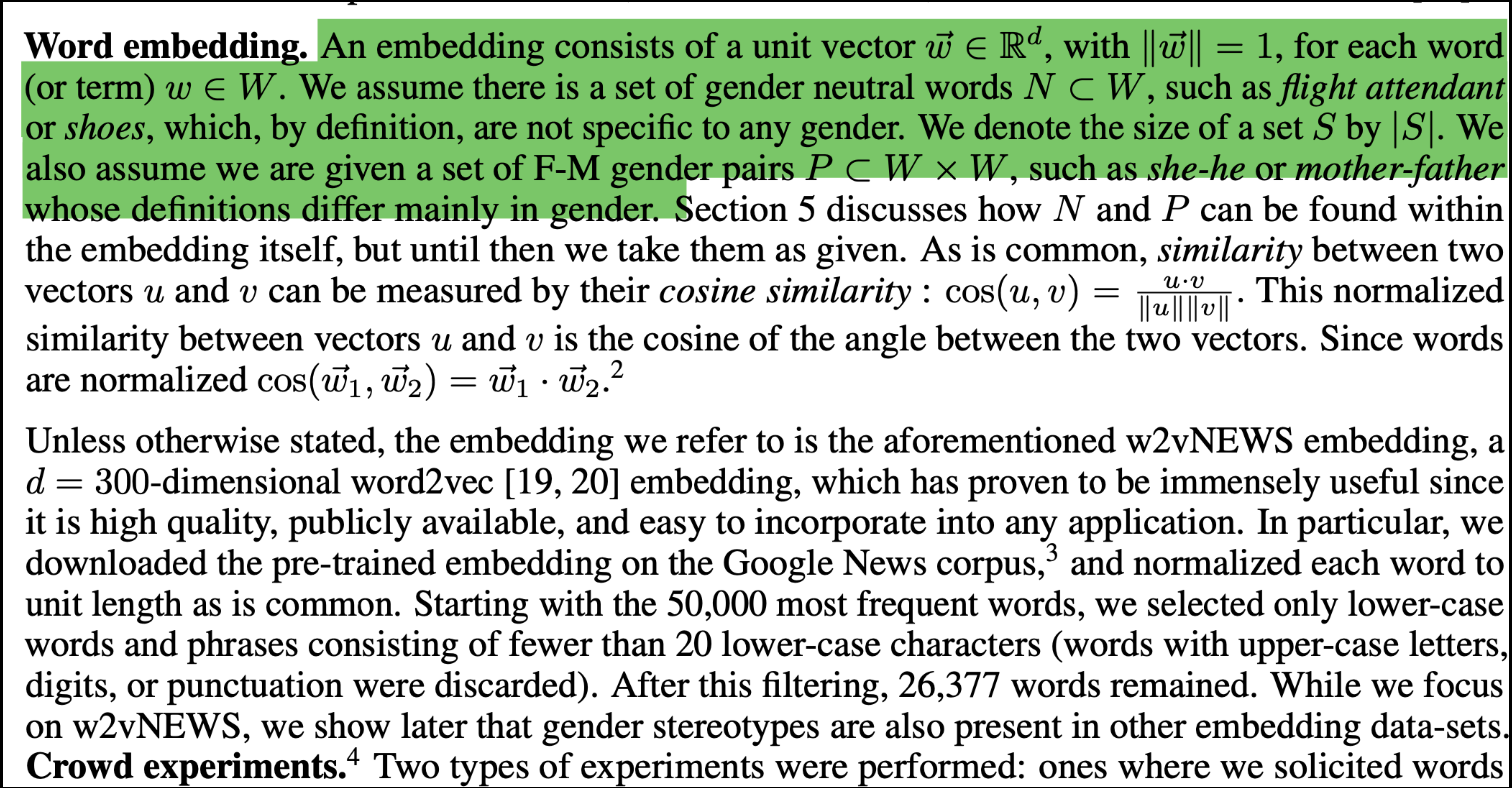
- model %>% closest_to(~ "business" - "she" + "he",20)

she-he

- model %>% closest_to(~ "she" - "he")

he-she

- model %>% closest_to(~ "he" - "she",50)



Word embedding. An embedding consists of a unit vector $\vec{w} \in \mathbb{R}^d$, with $\|\vec{w}\| = 1$, for each word (or term) $w \in W$. We assume there is a set of gender neutral words $N \subset W$, such as *flight attendant* or *shoes*, which, by definition, are not specific to any gender. We denote the size of a set S by $|S|$. We also assume we are given a set of F-M gender pairs $P \subset W \times W$, such as *she-he* or *mother-father* whose definitions differ mainly in gender. Section 5 discusses how N and P can be found within the embedding itself, but until then we take them as given. As is common, *similarity* between two vectors u and v can be measured by their *cosine similarity* : $\cos(u, v) = \frac{u \cdot v}{\|u\|\|v\|}$. This normalized similarity between vectors u and v is the cosine of the angle between the two vectors. Since words are normalized $\cos(\vec{w}_1, \vec{w}_2) = \vec{w}_1 \cdot \vec{w}_2$.[2]

Unless otherwise stated, the embedding we refer to is the aforementioned w2vNEWS embedding, a $d = 300$-dimensional word2vec [19, 20] embedding, which has proven to be immensely useful since it is high quality, publicly available, and easy to incorporate into any application. In particular, we downloaded the pre-trained embedding on the Google News corpus,[3] and normalized each word to unit length as is common. Starting with the 50,000 most frequent words, we selected only lower-case words and phrases consisting of fewer than 20 lower-case characters (words with upper-case letters, digits, or punctuation were discarded). After this filtering, 26,377 words remained. While we focus on w2vNEWS, we show later that gender stereotypes are also present in other embedding data-sets.

Crowd experiments.[4] Two types of experiments were performed: ones where we solicited words

Filter the model

- ShortWordIndex = which(nchar(rownames(model))<20)

- model2 = model[ShortWordIndex,]

- grep(rownames(model2),pattern=paste(LETTERS,collapse="|"),value=TRUE)

- model3 = model2[-grep(rownames(model2), pattern=paste(LETTERS,collapse="|")) ,]

- model3 = normalize_lengths(model3)

Closeness of terms

- Cosine similarity:
- $cos(\theta) = \frac{\sum_{i=1}^{N} A_i B_i}{\sqrt{\sum_{i=1}^{N} A_i^2}\sqrt{\sum_{i=1}^{N} B_i^2}}$ for vectors A, B with components A_i, $B_{i\backslash}$
- -1 = exact opposite, 1 = exactly the same, 0 = orthogonal

- cosineSimilarity(model3[c("shoe","shoes","flight_attendant"),],model3[c("mother","father"),])

Project occupations into the he-she plane

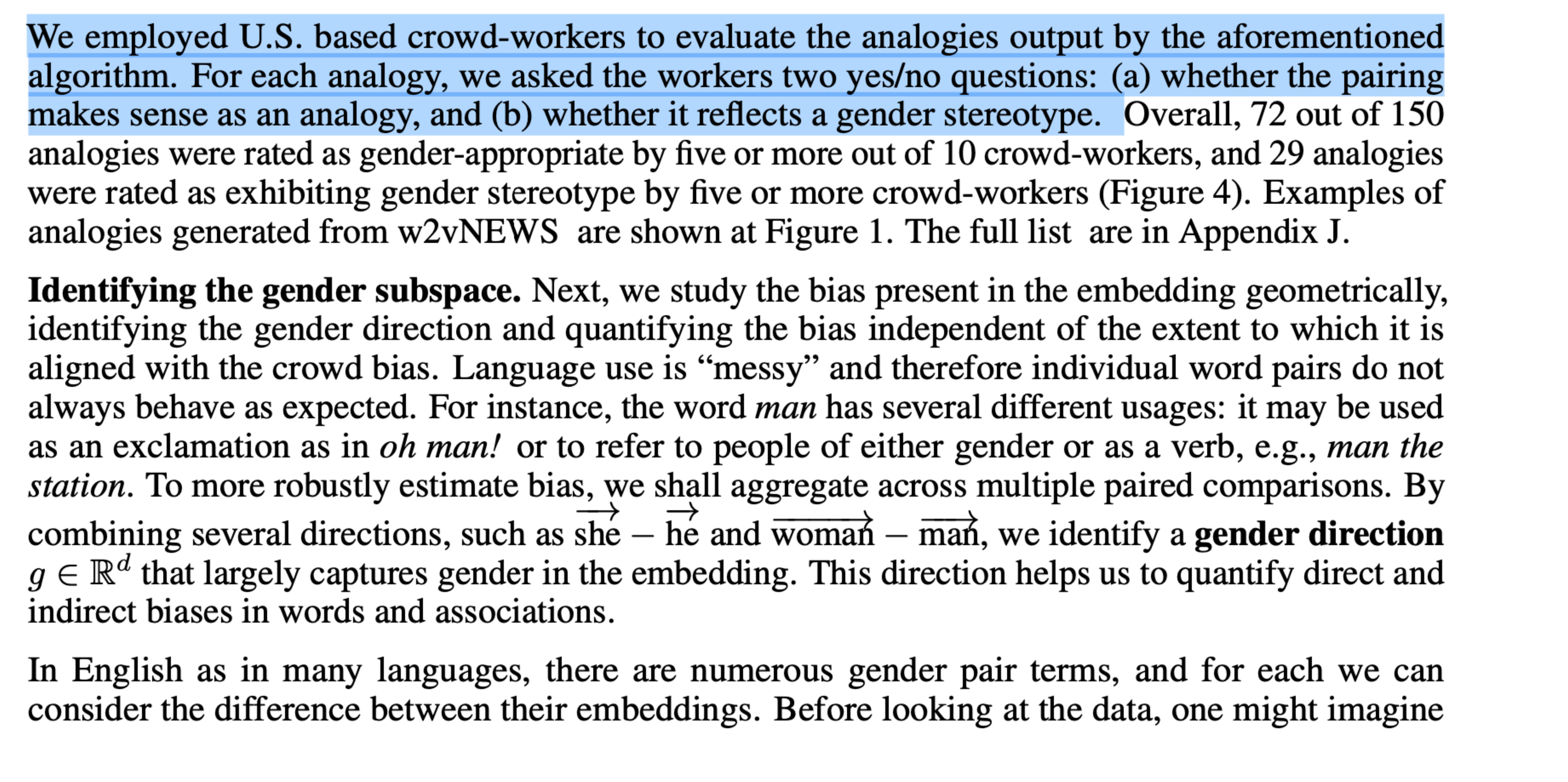
- #Jod titles: https://github.com/jneidel/job-titles
- Jobs = read.csv(file="job-titles-master/job-titles.txt",header=FALSE)
- DataJobs = unique(unlist(strsplit(
        - gsub(
            - grep(Jobs$V1,pattern="data",value=TRUE),
        - pattern = "[[:punct:]]",replacement=""),
    - split="\\s")))
- DataJobsIndex = grep(rownames(model3),
- pattern=gsub(paste("(",**sort(DataJobs)[-1]**,")",collapse="|"),pattern="\\s",replacement=""))
- rownames(model3[DataJobsIndex,])

Relationship Planes

- heshe = model3[[c("he","she"),average=F]]
- ##### JOB TITLES
- # model3[DataJobsIndex,] here restricts to any data job titles
- he_and_she = model3[DataJobsIndex,] %>% cosineSimilarity(heshe)
- # Filter to the top 50.
- he_and_she = he_and_she[
-   rank(-he_and_she[,1])<50 |
-   rank(-he_and_she[,2])<50,
-   ]
- as_tibble(he_and_she) %>% **mutate(dataword=rownames(he_and_she))** %>%
- ggplot(aes(he,she)) +
- **geom_jitter**(alpha = 0.1, width = 0.1) +
- geom_text(aes(label = dataword), **check_overlap** = TRUE, vjust = 1.5) +
- geom_abline(color = "red")

Gendered words and non gendered words

- Mother, father, likely_candidate, flight_attendant,…



We employed U.S. based crowd-workers to evaluate the analogies output by the aforementioned algorithm. For each analogy, we asked the workers two yes/no questions: (a) whether the pairing makes sense as an analogy, and (b) whether it reflects a gender stereotype. Overall, 72 out of 150 analogies were rated as gender-appropriate by five or more out of 10 crowd-workers, and 29 analogies were rated as exhibiting gender stereotype by five or more crowd-workers (Figure 4). Examples of analogies generated from w2vNEWS are shown at Figure 1. The full list are in Appendix J.

Identifying the gender subspace. Next, we study the bias present in the embedding geometrically, identifying the gender direction and quantifying the bias independent of the extent to which it is aligned with the crowd bias. Language use is "messy" and therefore individual word pairs do not always behave as expected. For instance, the word *man* has several different usages: it may be used as an exclamation as in *oh man!* or to refer to people of either gender or as a verb, e.g., *man the station*. To more robustly estimate bias, we shall aggregate across multiple paired comparisons. By combining several directions, such as $\overrightarrow{\mathrm{she}} - \overrightarrow{\mathrm{he}}$ and $\overrightarrow{\mathrm{woman}} - \overrightarrow{\mathrm{man}}$, we identify a **gender direction** $g \in \mathbb{R}^d$ that largely captures gender in the embedding. This direction helps us to quantify direct and indirect biases in words and associations.

In English as in many languages, there are numerous gender pair terms, and for each we can consider the difference between their embeddings. Before looking at the data, one might imagine

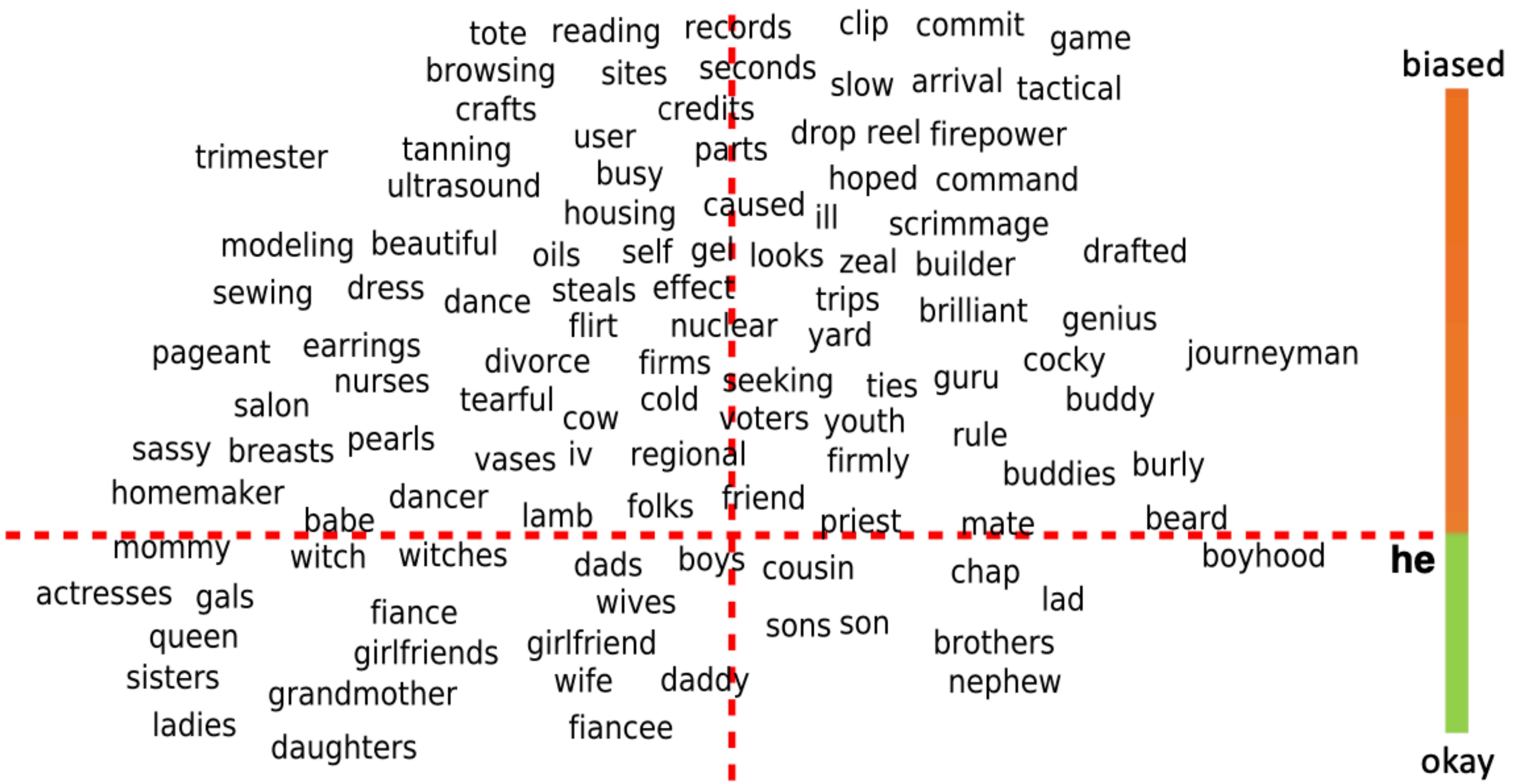
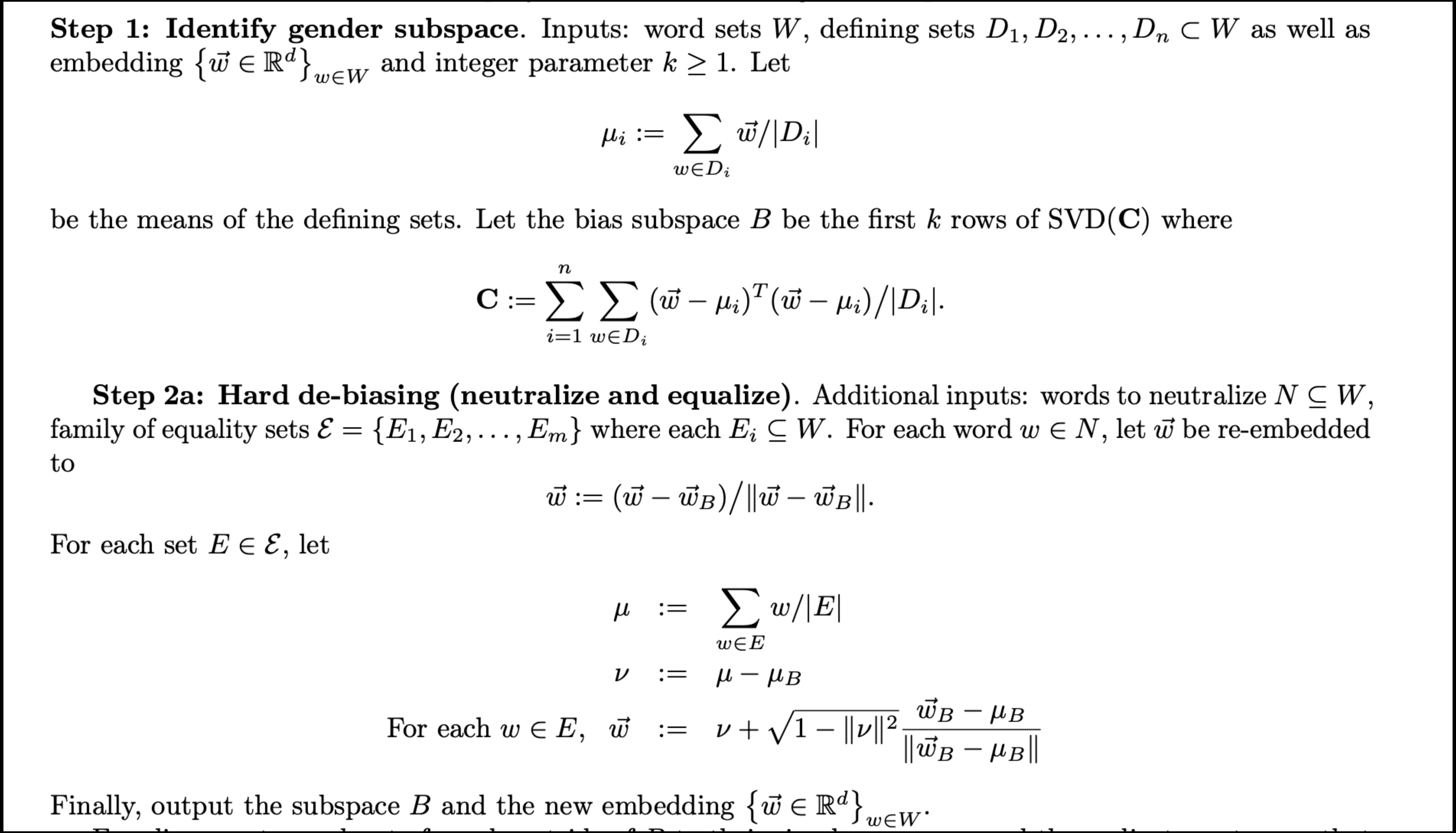


Figure 3: Selected words projected along two axes: x is a projection onto the difference between the embeddings of the words *he* and *she*, and y is a direction learned in the embedding that captures gender neutrality, with gender neutral words above the line and gender specific words below the line. Our hard debiasing algorithm removes the gender pair associations for gender neutral words. In this figure, the words above the horizontal line would all be collapsed to the vertical line.

- Identify a gender direction / subspace

- **Neutralize** by subtracting out this direction

- **Equalize** perfectly equalizes sets of words outside the subspace (granfather —> he == grandmother —> she)



Step 1: Identify gender subspace. Inputs: word sets W, defining sets $D_1, D_2, \ldots, D_n \subset W$ as well as embedding $\left\{\vec{w} \in \mathbb{R}^d\right\}_{w \in W}$ and integer parameter $k \geq 1$. Let

$$\mu_i := \sum_{w \in D_i} \vec{w}/|D_i|$$

be the means of the defining sets. Let the bias subspace B be the first k rows of SVD($\mathbf{C}$) where

$$\mathbf{C} := \sum_{i=1}^{n} \sum_{w \in D_i} (\vec{w} - \mu_i)^T (\vec{w} - \mu_i)/|D_i|.$$

Step 2a: Hard de-biasing (neutralize and equalize). Additional inputs: words to neutralize $N \subseteq W$, family of equality sets $\mathcal{E} = \{E_1, E_2, \ldots, E_m\}$ where each $E_i \subseteq W$. For each word $w \in N$, let $\vec{w}$ be re-embedded to

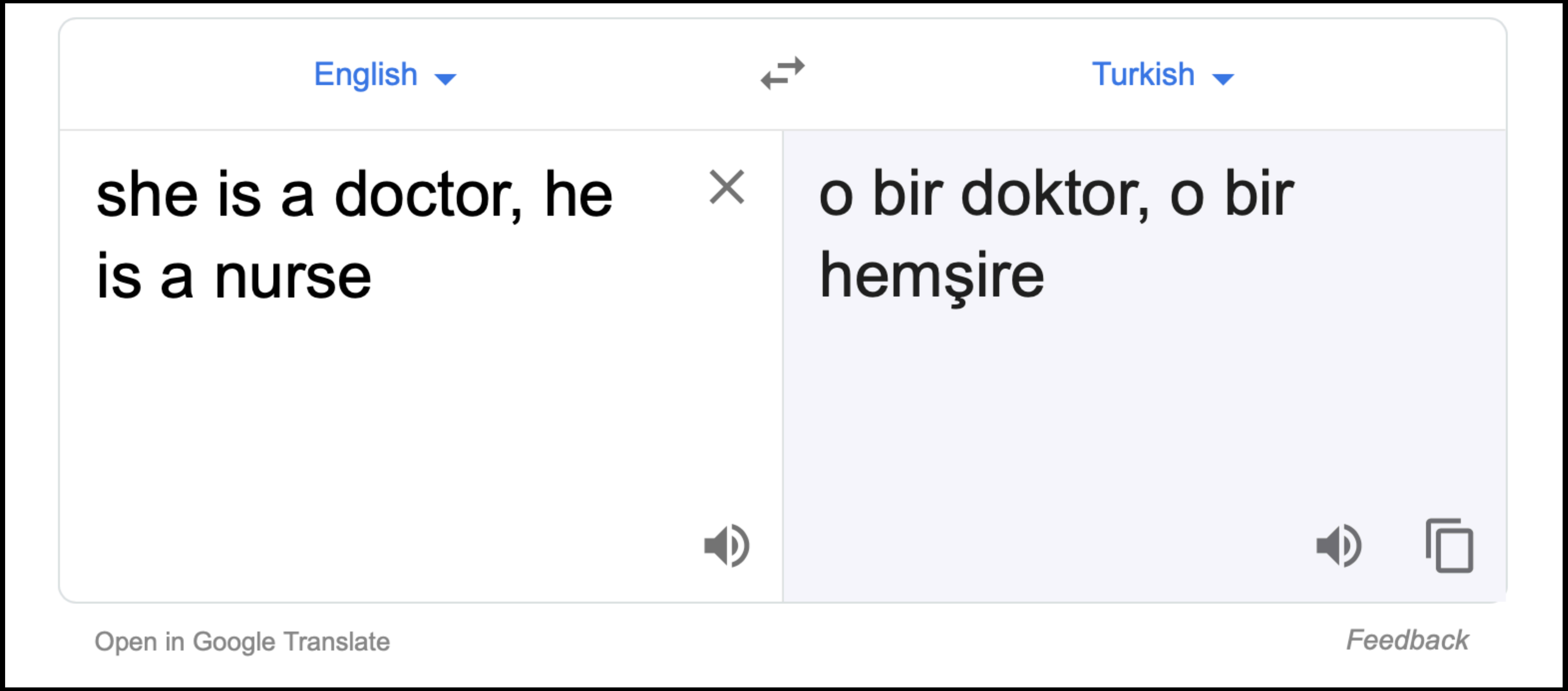
$$\vec{w} := (\vec{w} - \vec{w}_B)/\|\vec{w} - \vec{w}_B\|.$$

For each set $E \in \mathcal{E}$, let

$$\begin{aligned} \mu &:= \sum_{w \in E} w/|E| \\ \nu &:= \mu - \mu_B \\ \text{For each } w \in E, \quad \vec{w} &:= \nu + \sqrt{1 - \|\nu\|^2} \frac{\vec{w}_B - \mu_B}{\|\vec{w}_B - \mu_B\|} \end{aligned}$$

Finally, output the subspace B and the new embedding $\left\{\vec{w} \in \mathbb{R}^d\right\}_{w \in W}$.

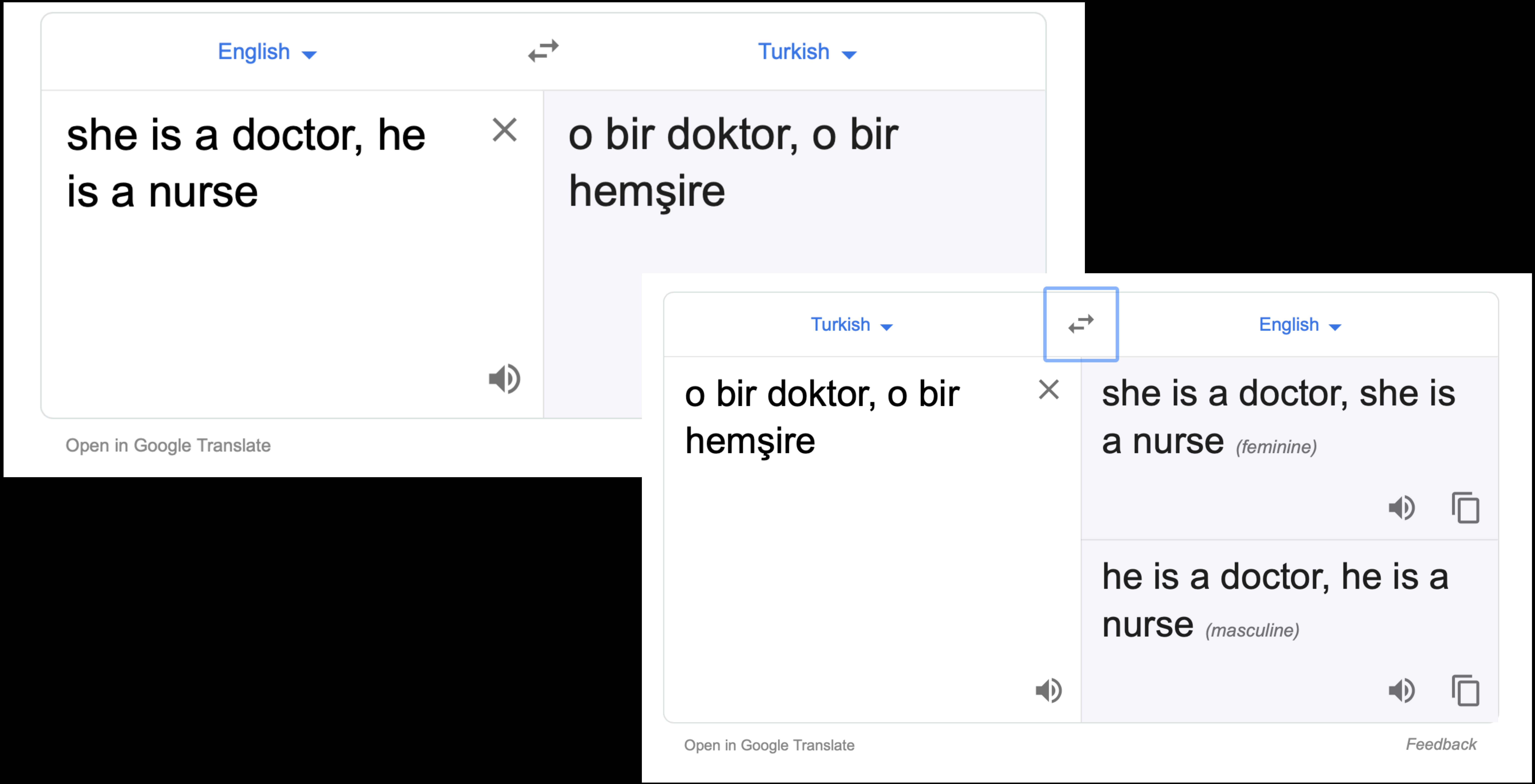
De-biased algorithms



she is a doctor, he is a nurse

o bir doktor, o bir hemşire

De-biased algorithms



English ▾ ⇄ Turkish ▾

she is a doctor, he is a nurse ✕

o bir doktor, o bir hemşire

Open in Google Translate

Turkish ▾ ⇄ English ▾

o bir doktor, o bir hemşire ✕

she is a doctor, she is a nurse *(feminine)*

he is a doctor, he is a nurse *(masculine)*

Open in Google Translate

Feedback