Conceptual and methodological problems with bias detection and avoidance in natural language processing

Alicja Dobrzeniecka

2021-06-16

Contents

Chapter 1

Introduction

Placeholder

Chapter 2

Cosine similarity and bias detection

2.1 Word embeddings

To understand what cosine similarity measurement is, one first needs to grasp the concept of translating words to a computer-readable form. In the field of natural language processsing there are two main types of words representation — localist and distributed. One-hot encoding is an example of a method used to achieve a localist representation of words. Here each vector contains information only about a single data point, this is achieved by first mapping categorical values (words) to integers and then to each integers a binary vector is assigned which contains only 0s except for the index of the integer, which is assigned 1. An example of a localist representation is:

word	1	2	3	4	5
woman	1	0	0	0	0
man	0	1	0	0	0
girl	0	0	1	0	0
boy	0	0	0	1	0

word	1	2	3	4	5
monarch	0	0	0	0	1

In the example above it is clear that the length of the vectors increases with the number of words in a vocabulary. It is not a very computationally efficient representation. It has other flaws as well. For example, it is unable to capture the resemblance between words appearing in similar contexts.

In contrast to the localist representation, a distributed representation returns vectors that contain continuous values instead of discrete 1s and 0s. Word embeddings are a class of various techniques that allow one to represent words as distributed vectors. Such learned representations of text have certain properties. At least prima facie, they store similar (or at least co-occurring) words close to each other in a vector space. An example of distributed representation is:

word	1	2	3	4
woman	0.456	0.267	0.675	0.131
man	0.451	0.897	0.472	0.088
girl	0.604	0.262	0.414	0.706
boy	0.279	0.172	0.475	0.010
monarch	0.565	0.678	0.463	0.975

One of the advantages of using a distributed representation is that one is able to represent an enormous number of concepts with a smaller number of units. It is also possible to better capture similarities as words of similar meanings can have similar numeric vectors.

The numbers occurring in such representations are not random. They are learned in a

process that uses a very shallow neural network. There are various types of techniques used for learning the vectors representations. One of the most straightforward ones is a skip-gram model. Given a word the model tries to predict its neighboring words from the sentence. The mathematics behind the process relies on the idea that the prediction concerns the conditional probability of the adjacent words. The algorithm tries to minimize the loss function, which penalizes the system for discrepancy with actual co-occurrence frequencies in the corpus. One can choose various parameters of the model, such as the window size that determines how many surrounding words the model should predict. After preparing such a fitted modeln=, one takes only the learned weights from a neural network, and uses them as vectors in a word embeddings representation.

Word embeddings have many applications in natural language processing. They are handy in document search and information retrieval. They also play their part in improving automatic translations. Well learned word representations may also contribute to the improvement of sentiment analysis or spam detection.

2.2 Cosine similarity and distance

Cosine similarity is often used as a method of finding out whether vector representations for two words suggest that they are similar or somehow connected. Cosine similarity is the cosine of the angle between two vectors: the result of dividing their inner product (dot product usually) by the product of their magnitudes.

It is worth mentioning one more point concerning cosine similarity. After the vectors are normalized to have length equal to 1, inner product itself (often dot product) is used to measure the similarity.

$$\mathsf{cosineSimilarity}(A, B) = \frac{A \cdot B}{||A||\,||B||} \tag{Sim}$$

Cosine similarity is considered a proper tool for this operation as its result has a clear connection to geometry and at least for a low number of dimensions may be easily interpreted. Using this scale, one can compare vector similarities in a fairly clear manner. When the vectors are aligned perpendicularly to each other, their similarity equals 0 (which is the same as the cosine of 90 degrees). This tells us that the similarity between the vectors is small. As the angle between vectors decreases, cosine similarity approaches one, which stands for the greatest similarity.

One of the limitations of this measure is that it informs us only about similarities between vectors in terms of their orientation. However, it is often argued that in comparing words in terms of this metric, the magnitude of vectors may be treated as irrelevant, as the most important information pertains to direction.

In what follows, it is important to distinguish between cosine similarity and cosine distance, defined as:

$$\begin{aligned} \mathsf{cosineDistance}(A, B) &= 1 - \mathsf{cosineSimilarity}(A, B) \\ &= 1 - \frac{A \cdot B}{||A||\,||B||} \end{aligned} \tag{Sim}$$

The greater the similarity between two vectors, the smaller the distance between them. The cosine distance ranges between 0 and 2. If the vectors are in an opposite direction to each other, the cosine distance is 2. And if the vectors are extremely similar then the cosine distance is very close to 0.

One should note that cosine distance is not exactly a distance measure while it does

not meet triangle inequality requirements. The triangle inequality formula says that for any triangle, the sum of the lengths of any two sides must be greater than or equal to the length of the remaining side. As shown in Statexchange.com in the case of cosine distance it would have to fulfill this equation 1+cos-sim(,) < cos-sim(,) + cos-sim(,). If one chooses specific unit vectors it is easy to demonstrate that the triangle inequality is not preserved.

2.3 Cosine distance in a one-class bias detection

Bolukbasi, Chang, Zou, Saligrama, & Kalai (2016) define similarity between words as the outcome inner product of their normalized vectors. They focus on examining what the geometry of word embedding is in regard to "he" and "she" words. In other words, whether the similarity between those concepts and other words reflects expected gender stereotypes. They test this hypothesis by investigating whether there is a connection between word embeddings representing certain professions and words referring to gender. They also evaluate whether automatically produced analogies between words reflect the stereotypes as well.

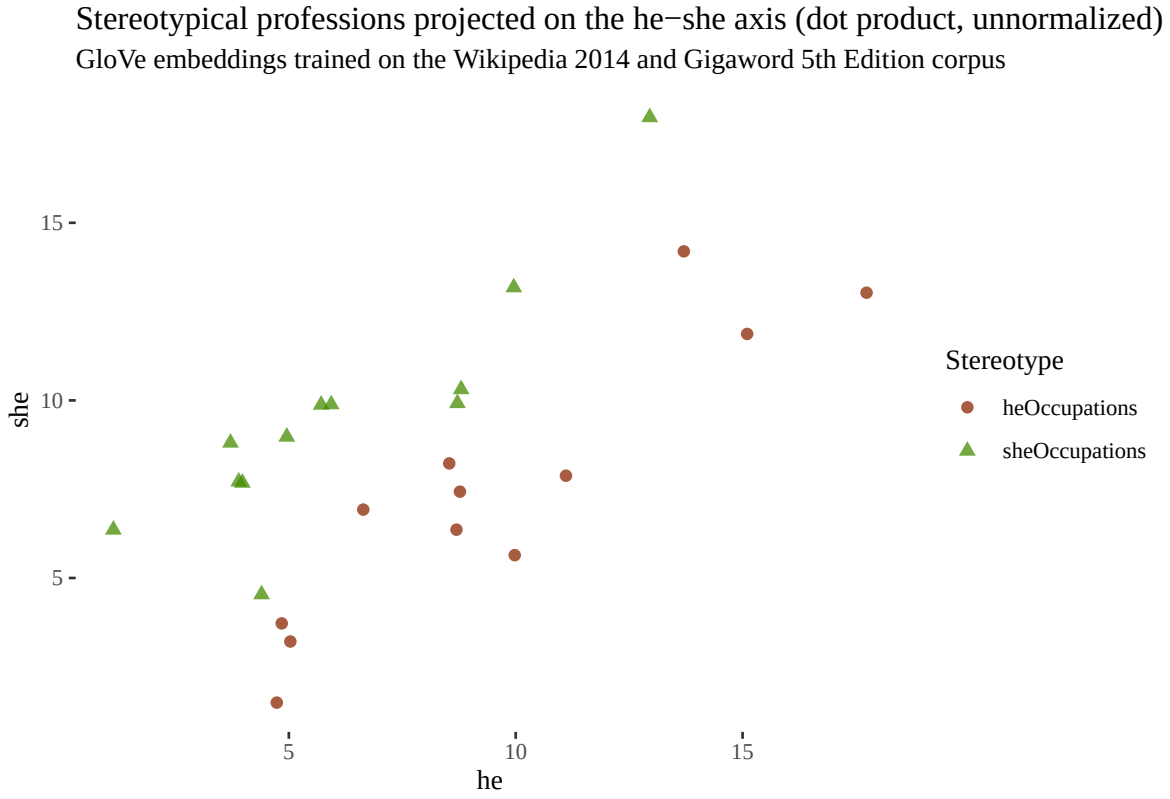
Here the gender bias of a word w is understood as its projection on the gender direction $\vec{w} \cdot (\overrightarrow{he} - \overrightarrow{she})$ (the gender direction is the top principal compontent of ten gender pair difference vectors). The underlying idea is that no bias is present if non-explicitly gender words are in equal distance to both elements in all explicitly gender pairs. Given the (ideally) gender neutral words N and the gender direction g the direct direct gender bias is defined as the average distance of the words in N from g (c is a parameter determining how strict we want to be):

$$\mathsf{directBias_c(N, g)} = \frac{\sum_{w \in N} |\mathsf{cos}(\vec{w}, g)|^c}{|N|} \tag{2.1}$$

A very vivid way to follow their method of arguing that bias in word embeddings is real is to plot the values of inner product of chosen words. The plot below does not originate from the original paper (it is from https://www.kaggle.com/rtatman/gender-bias-in-word-embeddings) but similar visualization may be found there. Data used to create our plot is as follows.

Occupations associated with feminine: **"homemaker", "nurse", "receptionist", "librarian", "socialite", "hairdresser", "nanny", "bookkeeper", "stylist", "housekeeper", "interior designer", "guidance counselor"**

Occupations associated with masculine: **"maestro", "skipper", "protege", "philosopher", "captain", "architect", "financier", "warrior", "broadcaster", "magician", "fighter pilot", "boss"**



The points in the plot above result from the calculation of the inner product of a chosen vector for a profession word and a vector for a gender word (she or he). Inner product of two vectors expresses similarity between words. This assumption originates from the geometry and properties of a vector space.

2.4 Cosine distance in a multi-class bias detection

Manzini, Lim, Tsvetkov, & Black (2019) present a different approach towards finding similarities between classes of words. The authors claim that texts available online are full of direct or indirect human stereotypes. As a result, word embeddings are prone to learn and maybe amplify those biases and propagate them further into AI models that are used for various applications. Cosine distance is used in the article as a measure to first argue for the existence of multi-class bias and then to show how through bias mitigation techniques the bias may be decreased.

They modify WEAT to a multi-class setting, introducing Mean Average Cosine similarity as a measure of bias (in fact, in the paper they report distances rather than similarities). Let $T = \{t_1, ..., t_k\}$ be a class of protected word embeddings, and let each $A_j \in A$ be a set of attributes stereotypically associated with a protected word). Then:

$$S(t_i, A_j) = \frac{1}{|A_j|} \sum_{a \in A_j} \mathsf{cos}(t, a) \tag{2.2}$$

$$MAC(T, A) = \frac{1}{|T|\,|A|} \sum_{t_i \in T} \sum_{A_j \in A} S(t_i, A_j) \tag{2.3}$$

As the code has been provided by the authors we were able to reconstruct their results. The main steps in the procedure are as follows. Let us go through an example that refers to the process of hard debiasing on religious attributes.

First we load word embeddings from reddit.US.txt.tok.clean.cleanedforw2v.w2v dataset.

The word embeddings have only 50 dimensions and the number of individual words from the dataset is 44895. The authors assume that the protected group should ideally not have high cosine similarity to stereotypical words. The word embeddings geometry should not place this group close to harmful stereotypes, if it is to be bias-free. For instance, let's look at the religion-related words.

Protected words by religion type:

"jew" : ["judaism", "jew", "synagogue", "torah", "rabbi"]

"christian" : ["christianity", "christian", "church", "bible", "priest"]

"muslim" : ["islam", "muslim", "mosque", "quran", "imam"]

Stereotypical words by religion type:

"jew" : ["greedy", "cheap", "hairy", "liberal"]

"christian" : ["judgemental", "conservative", "familial"]

"muslim" : ["violent", "terrorist", "dirty", "uneducated"]

We have prepared a table presenting the values of cosine distance for each protected word with each attribute (stereotype). The part of the results is shown below.

```
religion <- read.csv("../datasets/religionReddit.csv")[-1]
colnames(religion) <- c("protectedWord","wordToCompare","wordClass",
                        "cosineDistance","cosineSimilarity","connection")
religion$wordClass <- as.factor(religion$wordClass)
levels(religion$wordClass) <- c("christian","human","jewish","muslim","neutral")
head(religion)  %>%  kable(format = "latex",booktabs=T,
                      linesep = "",  escape = FALSE,
                      caption = "Head of the religion dataset.") %>%
                      kable_styling(latex_options=c("scale_down"))
```

Table 2.3: Head of the religion dataset.

protectedWord	wordToCompare	wordClass	cosineDistance	cosineSimilarity	connection
judaism	violent	muslim	0.7141939	0.2858061	different
judaism	terrorist	muslim	0.7461333	0.2538667	different
judaism	dirty	muslim	1.2002599	-0.2002599	different
judaism	uneducated	muslim	0.7885469	0.2114531	different
judaism	greedy	jewish	1.0026172	-0.0026172	associated
judaism	cheap	jewish	1.2323229	-0.2323229	associated

In the article there was no analysis of individual distances but the general look at the data through the usage of mean. The authors introduced a metric that tries to generalize the presence of bias through the classification of multi-class bias in groups of words connected with gender, religion or race. In the process they first take the mean of cosine distances between a given protected word and attributes assigned to each stereotype. They do not differentiate between stereotypes associated with a word and stereotypes associated with different words (in the case of religion, stereotypes characteristic for Christianity has also cosine distance measured with for instance, Judaism or Islam). Then, after collecting the list of mean cosine distances, they average the list to obtain one final value representing the whole group, in this example religion, for which the final mean of all mean distances is equal to 0.859.

In the article the authors also try to remove previously defined biases from word embedding. First they identify the bias subspace using Principal Component Analysis (PCA) which is a technique for dimensionality reduction. It is applied here to choose the subspace that contains the greatest amount of information. There can be many subspaces found in a given group, for example in terms of religion one can identify at least a few sets that are to grasp the concept of religion in general:

[”judaism”, ”christianity”, ”islam”]

[”jew”, ”christian”, ”muslim”]

[”synagogue”, ”church”, ”mosque”]

The idea is to find a set that provides enough information to create from it a vector representing the concept of religion among words. This strategy is based on the idea that different dimensions of vectors contain different types of information and in some words in vector layers (subspaces) the information about religiousness is implicitly conveyed. In some cases this knowledge is useful but in the case of harmful stereotypes one does not want to include the concept of religion in stereotypical words.

After finding the bias subspace, they use it to modify the vector values individually so that their cosine distances towards certain words are changed. In the case of stereotypes the aim is to make the cosine distances larger so that the association between protected word and harmful stereotype is smaller.

In the final step there is an evaluation of the results. The cosine distances are calculated again but this time using debiased vocabulary. After taking the mean of all distances one final value is obtained and then it is compared with the average value from the beginning. If the cosine distances are on average greater than before then it leads the authors to the conclusion that improvement has been achieved. As the cosine distance increases it is assumed that the association between protected and stereotypical words decreases.

2.5 Limitations of the approach

1. Selection of attributes

The attributes are taken from different sources, there is no principled justification for their choice. From our analysis it will become clear that the list is rather uneven.

There is no mention of methodology for deciding on the number of attributes necessary to decide a hypothesis on the given size of dataset. There are however some ways to

estimate how many samples we need to make sure that the result is significant. Our research will show that the numbers used are rather insufficient.

2. No control while taking the mean of cosines

The authors use the mean average cosine similarity to check on multi-class similarity between protected word and harmful stereotypes. They average the results until they obtain one final value to represent the mean cosine distance between protected word from a given class and the attributes of that class.

3. Hiding the impact of uncertainty

A mean hides issue and as there are pairs having negative and small similarities and there are those that have similarity equal to 0.5, the resulting calculation seems to be in norm. Additionally in such method the uncertainty is also not included which makes it even more difficult to give reasonable interpretations of the results. We propose the use of Bayesian method to obtain some understanding of the influence the uncertainty has on the interpretation of final results.

4. No word class distinction

In the original paper words from all three religions were checked with all of the stereotypes which means that there was no distinction between classes in which the stereotype is associated with a given religion, as opposed to the situation in which it is associated with another one. Not all of the stereotypical words have to be considered as harmful for all of the religions. In our analysis we distinguished between stereotypes associated with a given group, stereotype associated with different groups and control groups; neutral words and stereotypes-free human predicates.

5. Interpreting the results

Assuming for a moment that the value of multi-class cosine distance is correct, one may

question the interpretation. Manzini, Lim, Tsvetkov, & Black (2019) summarize the averages of cosine distance per group (gender, race, religion). We would like to focus now on analyzing the values relating to religious biases. Here is the relevant fragment of table:

Religion Debiasing	MAC
Biased	0.859
Hard Debiased	0.934
Soft Debiased ($\lambda = 0.2$)	0.894

MAC stand for mean average cosine similarity, although in reality the values of cosine distance are given. What may attract attention is the fact that the value of cosine distance in "Biased" category is already quite high even before debiasing. High cosine distance indicates low cosine similarity between values. One could think that average cosine similarity equal to approximately 0.141 is not significant enough to consider it as bias. However the authors aim to mitigate "biases" in vectors with such great distance to make it somewhat larger. Methodologically the question is, on what basis is this small similarity still considered as a proof of a bias presence, and whether these small changes are meaningful. This is in general the problem of scale and the lack of universal intervals. Proper intervals could be used to decide whether given cosine similarity is high enough to consider the words to be more similar than if we choose them at random.

6. The curse of dimensionality

In our case, the curse of dimensionality may take place when there is an increase in volume of data that results in adding extra dimensions to the Euclidean space. According to the article Analyticsindiamag Curse of dimensionality as the number of features increases, it may be harder and harder to obtain useful information from the data with the usage of available algorithms. One may notice that more data should contribute to

greater amount of information but more information also means greater risk of noise and distractions in data. At the same time, many times modern solutions are adapted to smaller dimensions and their results in higher ones are not intuitive or may be prone to be mistaken.

Using cosine similarity in high dimensions in word embeddings may also be prone to the curse of dimensionality. According to Venkat (2018) there are reasons to consider this phenomenon when searching for word similarities in higher dimensions. An experiment is conducted that aims at showing how the similarity values and variation change as the number of dimensions increases. The hypothesis made in the paper states that two things will happen as the number of dimensions increase. First, the effort required to measure cosine similarity will be greater, and two, the similarity between data will blur out and have less variation. The authors generate random points with increasing number of dimensions where each dimension of a data point is given a value between 0 and 1. Then they pick one vector at random from each dimension class and calculate the cosine similarity between the chosen vector and the rest of the data. Then they check how the variation of values changes as the number of dimensions increases. It seems like the more dimensions there are, the smaller the variance and therefore it is less obvious how to interpret the resulting cosine similarities. Maybe the scale should be adjusted to the number of dimensions and variance so that it still gives us sensible information about data. According to some researches the cosine similarity in high dimensions is not reliable enough to trust it as it may be the case that choosing words on random may result in getting similar values as when picking them consciously.

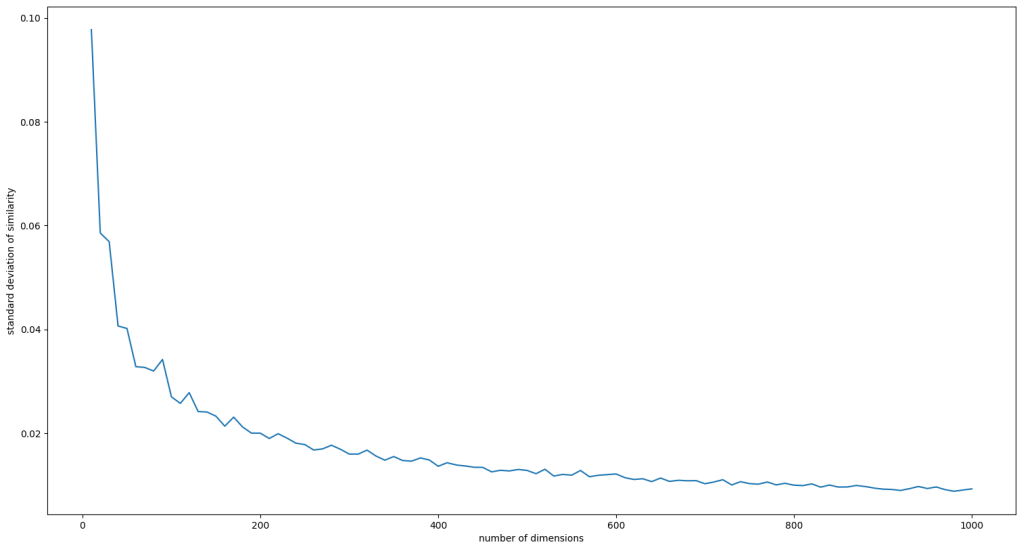


Figure 2.1: curse of dimensionality, number of dimensions on the x axis, standard deviation of similarity on the y axis

Chapter 3

Walkthrough with the religion dataset

Placeholder

3.1 Loading and understanding the dataset

3.2 First look at the empirical distributions

3.3 Looking at the islam-related words

3.4 Bayesian model structure and assumptions

3.5 Choosing predictors

3.6 Dataset-level coefficients

3.7 Model structure and assumptions

3.8 Protected classes in Reddit and Google embeddings

3.9 Dataset-level coefficients after debiasing

3.10 Protected classes after debiasing

Chapter 4

Discussion

Placeholder

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *CoRR*, *abs/1607.06520*. Retrieved from http://arxiv.org/abs/1607.06520

Manzini, T., Lim, Y. C., Tsvetkov, Y., & Black, A. W. (2019). Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. Retrieved from http://arxiv.org/abs/1904.04047

Venkat, N. (2018). The curse of dimensionality: Inside out.