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

Alicja Dobrzeniecka

2021-06-10

Contents

Chapter 1

Introduction

Natural language processing (NLP) is a subfield of computer science that processes and analyzes language in text and speech with the use of modern programming methods. It has practical applications in everyday life as it concerns tasks such as email filters, smart assistants, search results, language translations, text analytics and so on. Models used to accomplish these tasks need a lot of data to learn from. This data originates from humans activities and historical recordings such as texts, messages or speeches. It turns out that in the learning process these models can learn implicit biases that reflect harmful stereotypical thinking still present in modern societies. One can find methods that aim at identifying and measuring hidden biases and/or try to remove them by modifying the models. There are many different types of models in NLP depending on a task that they are supposed to solve. However, all of them need as an input words represented by means of numbers and this is accomplished with word embedding models. The models usually assign the values based on the context in which the words appear. It means that the input data can have enormous influence on the outcome. The biases seem to have their primary source in the way the words are assigned the numerical values.

There is considerable amount of literature available on the topic of bias detection and mitigation in NLP models. Bolukbasi, Chang, Zou, Saligrama, & Kalai (2016) focuses on gender biases that may be observable while investigating the representation of job occupations and gender in terms of their assigned numerical values. The authors apply cosine similarity measurement to investigate the phenomenon where (the vectors corresponding to) words related to jobs that are stereotypically associated with a given gender are in fact in the model situated closer to this gender. They also use analogy tasks to evaluate if the bias is present in the word embedding model. They check analogies by comparing pairs of word vectors, for example they search for the word complementing the puzzle: man is to doctor as woman is to …? First they subtract word "man" from word "woman" and then they search for the ranked list of other words pairs that have similar vectors' difference. They also include in the formula a threshold to ensure that the resulting pairs could not be randomly picked.

However, as in Nissim, Noord, & Goot (2019) it is pointed out, there are some limitations of this approach. According to the authors in practice most of analogies implementations do not return any input words. This means that it does not make sense to expect the algorithm to return the same profession for both woman and man. Therefore this method seems to be limited in terms of bias detection. The other problems regard for example the choice of pairs and words that are used to detect the presence of discrimination as it is often subjective and without proper justification. Additionally the choice of parameter set in Bolukbasi, Chang, Zou, Saligrama, & Kalai (2016) formula to ensure that word pairs are not picked by random, is also not justified and changing it drastically influences the results.

Islam, Bryson, & Narayanan (2016) touches upon the topic of biases regarding race and gender. They apply knowledge from well-known psychological studies such as Implicit Association Test to research the relation between human stereotypical thinking and

model learnt biases to discover close relationship between these two. For the evaluation they use Word Embedding Association Test (WEAT) and the Word Embedding Factual Association Test (WEFAT).

Manzini, Lim, Tsvetkov, & Black (2019) proposes a novel way of using cosine similarity to obtain the information on assumed resemblance between words. They investigate an approach that enables them to measure the bias for a class (like gender, religion, race) and express the final result with a single metric.

It is worth noticing the general distinction of biases mentioned in Islam, Bryson, & Narayanan (2016). They refer to the publication concerning Implicit Association Test (Greenwald et al., 1998) that measures the strength of associations between concepts or stereotypes by calculating the time of reaction for special tasks. It is worth noticing that humans naturally exhibit some biases and that they do not always cause social concern. One can imagine the intuitive associations between for example insects and flowers, and the feelings of pleasantness or unpleasantness. In general, people would rather associate flowers with feeling pleasant than insects, and this preference could be named a bias or prejudice in some direction. However, this type of preference does not cause an uproar and is a rather morally neutral case. Unfortunately, there are other biases and prejudices that directly influence the quality of other people's lives and therefore they should be taken care of.

One can find various definitions trying to capture what bias and fairness actually are. With the choice of the definition, implications into the real-life applications may change as well. Mehrabi, Morstatter, Saxena, Lerman, & Galstyan (2019) mark out that there exist different types of biases such as historical bias, representation bias, measurement bias (the list is long). This indicates how complex the issue of bias is. Without the proper understanding and awareness of the problem, people are prone to unconsciously sustain the bias existence.

Mehrabi, Morstatter, Saxena, Lerman, & Galstyan (2019) also distinguish different types of discrimination, some of them will be briefly described. By protected attributes we mean those qualities, traits or characteristics that one cannot legally discriminate against. Direct discrimination occurs when protected attributes of individuals explicitly result in non-favorable outcomes toward them. In contrast in indirect discrimination individuals appear to be treated equally but anyway they end up being treated unjustly due to the hidden effects of biases towards their protected attributes. Systemic discrimination takes place when policies, customs or behaviors that result from certain culture or organizational structure lead to discrimination against some groups of people. Finally, very common statistical discrimination refers to using average group statistics to judge person belonging to the group.

The topic of discrimination is entangled with another concept which is fairness. It is essential to grasp some concepts of fairness to take them into consideration while designing implementation of some machine learning model. In Mehrabi2019Survey one may notice that depending on the context and application different definitions may be applied.

The most popular methods focus on comparing the similarity between words from protected groups and those that are considered to be stereotypical or harmful in some way. One can find in this group methods such as euclidean distance or cosine similarity (which is equivalent to dot product if the vectors are normalized). There are also other ways to detect the effects of biases. For example through the investigation of the model performance on certain tasks that validate if the model returns some values independently on gender or race or not.

The currently used methods (such as cosine similarity) make the similarity values often aggregated in a way that may lead to hasty conclusions. The averaging of values and the lack of uncertainty may lead to the incomplete picture of the bias situation in the

vocabulary.

One can find a number of articles on negative real-life implications resulting from the presence of unaddressed biases in the machine learning models.

In the paper we indicate how current methods used to detect biases in natural language models are limited from the perspective of Bayesian analysis.

Our research enhances the current way in which the bias detection is performed to make sure that it is methodologically valid.

The key hypothesis is that greater understanding of data and bias implications can be achieved when Bayesian methods are applied to issue.

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 models 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 model 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.

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

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.

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.

2.3 Cosine distance in a one-class bias detection

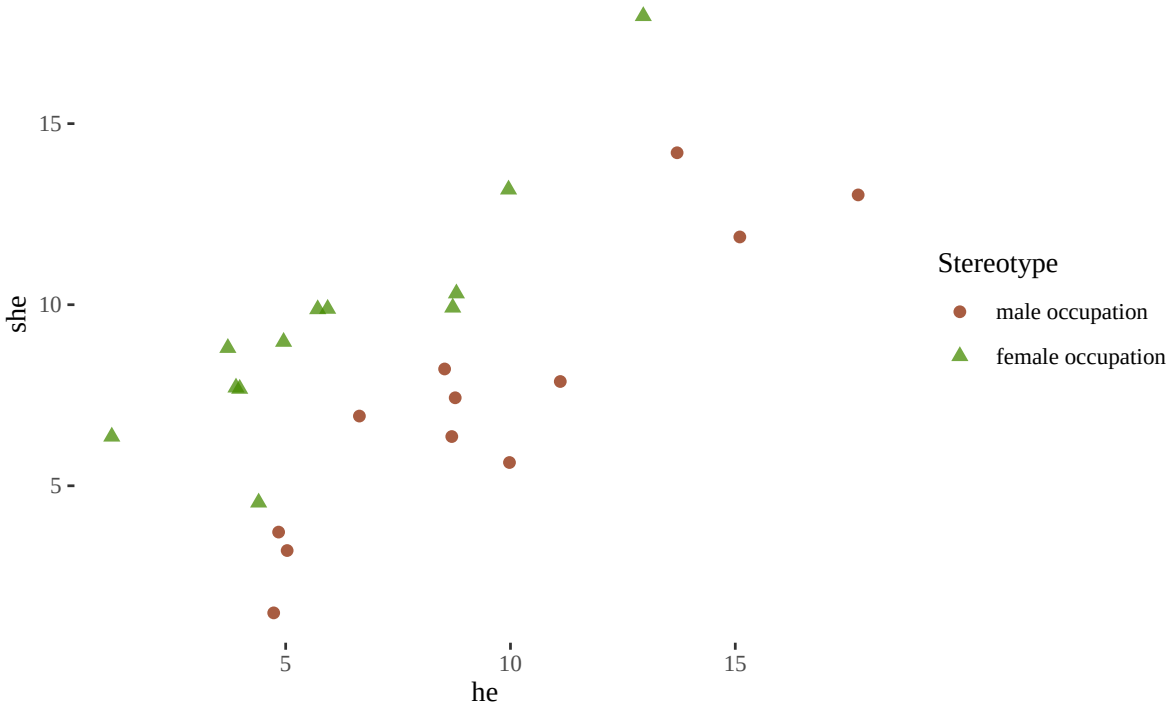
Bolukbasi, Chang, Zou, Saligrama, & Kalai (2016) define similarity between words as the outcome inner product of their normalized vectors. They focus on examining what the geometry of word embedding is in regard to "he" and "she" words. In other words, whether the similarity between those concepts and 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.

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"**

Stereotypical professions projected on the he−she axis (dot product, unnormalized)
GloVe embeddings trained on the Wikipedia 2014 and Gigaword 5th Edition corpus



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.

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.

The example refers to the process of hard debiasing on religious attributes.

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

- The word embeddings have only 50 dimensions
- The number of individual words from the dataset is 44895

2. The protected group the authors assume 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"]