CA4021: Data Science Final Year Project

Visualisation of Social Bias in Natural Language Models

Lara Murphy[1] and Dr. Jennifer Foster[2]

[1]lara.murphy239@mail.dcu.ie, ID: 18390323
[2]jennifer.foster@dcu.ie

Abstract. Stereotypes are over-generalized opinions that people hold about historically disadvantaged/minority groups, for example, *'Women are bad drivers'*. While stereotypes are taught to people, they are also taught to language models. Language models are becoming hugely successful due to their ability to process natural language. This has been done by training language models on natural language scraped from the internet. However, the text collected is not filtered to the extent required to remove language that exhibits harmful stereotypes. There have been benchmark datasets introduced to quantify how 'biased' a language model is, although these datasets have been criticised heavily. This project introduces an improved benchmark dataset to measure the social bias in language models, and provides a visualisation tool to allow this information to be accessed by the public. This dataset is taken from CrowS-Pairs, and follows three iterations of manual cleaning to remove as many problematic sentences as possible. The code used by CrowS-Pairs has been improved, their dataset has been cleaned and a new standard metric has been introduced. The new benchmark and processes are used to evaluate 22 BERT language models. This information is accessible via a website which allows users to view and interact with the visualisations.

Keywords: Language Models · Social Bias · Visualisation

1 Motivation and Background

Social bias is a very prominent flaw in our society. These biases are often based on stereotypes and prejudices that are systemically maintained. Stereotypes may be explicit or implicit biases, making the identification and eradication of these harmful views a difficult challenge. An example of explicit bias can be seen in the treatment of black people from police officers in America, where black people were found to be 3.7 times more likely to be arrested for possession of marijuana than white people, even though their rate of marijuana usage was comparable [5]. Research carried out by the University of Virginia shows an example of implicit bias, where women were found to be 47% more likely to suffer severe injuries in car accidents than men [4]. This is as a result of cars' safety features being designed for men. It's likely that this design flaw was not done with intent to put women at risk, rather it was done out of ignorance, due to the implicit bias held about men and women's status/roles.

The topic of social bias is being more openly discussed and dealt with in recent years. The Black Lives Matter (BLM) movement gained momentum particularly quickly in 2020 with the unjust death of a black man, George Floyd, at the hands of a police officer. This movement called out the social bias and prejudice held by policemen against black people, highlighting the devastating effects that these biases have. In light of the advancements being made to eradicate social bias in the real world, the onus is on the machine learning community to reduce the social bias present in the machine learned models.

Any machine learning model needs data, and the quality of this data affects the accuracy of the model. For example, when a facial recognition model is trained on a dataset consisting of majority white males, the model cannot identify the face of a black person [19]. Another example where training data is not representative of real-life use cases is in predictive policing [12]. When training these models on historical data that contains a lot of crime that have happened in a majority-black neighbourhood, the models will predict that these areas are more susceptible to crime and send more police officers to these areas. Police officers can only find crime in the places they are looking, meaning the crimes that are reported will be happening in these black neighbourhoods. This is a harmful feedback loop and leads to confirmation bias in the model that does not accurately represent the distribution of crime.

The motivation for this project is to optimise the benefits we can extract from machine learning, or Natural Language Processing (NLP) in particular. Machine learning is a hugely powerful tool that should not be disregarded as a result of biases contained in the training data. This project showcases the biases trained into a model, with the aim that researchers will be aware of the biases in a language model and take this into account when choosing to use a language model in an application.

2 Introduction

Machine learning is well-versed in the world of analysis and prediction of numbers, but what about language? Qualitative data cannot be interpreted by a computer, and so it has to be manipulated into the form of numbers. This can be done using natural language processing, by representing words as probabilities or, more commonly, representing words as vectors. We interact with natural language processing regularly, possibly without even realising it. The use of predictive text on our mobile phones, or suggestions made by search engines are some of the many powerful language modelling tools that we are exposed to in our daily lives. When these language models are trained on text corpora that exhibit socially problematic biases, the models influence, encourage and reinforce these harmful stereotypes. Biased profiling can arise when NLP tasks are developed using a language model trained on biased data. This is apparent in CV screening, where some systems tend to favour male candidates [15]. Training data that contains harmful biases can lead to amplified biases in the model/system. This can be seen in the GPT-2 language model [22], where it was trained on

text from outbound links from reddit. A survey carried out by Pew's Research Centre shows that less than a third of Reddit's members are female, and over two thirds are from America, a developed country [11]. This data is then further processed to remove outliers, which in this case, is the opinion of minority groups. This overrepresentation of white men's opinions leads to amplified biases in the overall language model.

3 Language Models

3.1 Transformers in Language Modelling

Language models are used for a number of different NLP tasks, such as translation, question-answering and sentiment analysis. Transformer models were introduced in 2017 in a paper produced by a team at Google Brain, 'Attention is all you need' [25]. Transformer-based models have had huge success, extending the state of the art on several NLP tasks as measured by leaderboards on specific benchmarks for English.

Transformer models differ from recurrent/feed-forward neural networks as they can process sequential data in parallel, and retain long-distance dependencies within sequences. This is possible using the mechanism of self-attention. Self-attention finds the strength of relationships between different tokens of a sequence in order to create a richer representation of an input sequence (or mathematically, a series of vectors). A new vector is created for each token in the input sequence, which is calculated using a weighted sum of all token vectors in the sequence. The dependencies from all tokens are represented in this new vector, which can allow for tokens to be processed in parallel.

3.2 BERT Models

BERT (Bidirectional Encoder Representations from Transformers) [8] is a neural architecture using transformers for language modelling. As BERT is a transformer-based model, it benefits from the self-attention mechanism. Self-attention is what allows BERT to be bidirectional, as token inputs will include context from other tokens to the left and to the right of the target token. The introduction of BERT was a huge advancement in enabling computers to 'understand' language computationally. BERT was created by scraping the entirety of the English Wikipedia and the BookCorpus [27], and subsequently trained using unsupervised learning in the tasks of Masked-Language Modelling (MLM) and Next Sentence Prediction (NSP). Masked Language Modelling is trained to predict the most probable word for a masked token in a sentence, for example, in the sentence 'My brother and [MASK] are older than me', the model would be likely to predict 'sister'. To train BERT in MLM, 15% of all tokens were masked, and BERT was required to predict the masked token correctly given the surrounding tokens. For NSP, BERT was trained to predict whether a sentence was likely to follow another sentence or not.

Using fine-tuning, BERT has advanced the state-of-the-art benchmarks in 11 NLP tasks, such as GLUE (General Language Understanding Evaluation) and SQuAD (Stanford Question Answering Dataset). BERT can be fine-tuned simply by replacing the output layer of the network with a new output layer designed specifically for the NLP task in question. The training of BERT is computationally expensive, however the fine-tuning does not require extra resources, so it is a cost-effective method of training a language model. There have since been several language models using BERT as a basis, such as RoBERTa (Robustly Optimised BERT Pretraining Approach) [16] and ALBERT (A Lite BERT) [13]. There have also been domain-specific language models created such as FinBERT (BERT trained for Financial Corpora) citefinbert, Legal BERT [6] and BioBERT (BERT trained for BioMedical Corpora) [14].

3.3 Limitations of Large Language Models

Bender et al. [1] highlight the issues associated with using large online text corpora to train language models, such as the environmental impact and its sensitivity to social biases. This paper recommends creating datasets as large as can be sufficiently documented. They also note the clear distinction between natural language processing and natural language understanding, as this is a concept that is often misconstrued by both the public and researchers. This paper outlines that the expected benefits of using petabytes of data is that it will increase diversity within the data and include a broad representation of how different individuals view the world. Unfortunately, this is not the case, the data needs to be filtered, and the opinions of people who conform to a hegemonic viewpoint are more likely to be retained. This paper emphasises the necessity for more resources to be spent on the gathering and documentation of training data and the implementation of a more justice-oriented data collection methodology.

4 Existing Literature

4.1 Social Bias Benchmark Datasets

While the progress of eradicating social bias in the real world is moving faster than the progress being made in the ML space, there are still some major advancements worth noting. The reason that there are stereotypes present in language models is that there are stereotypes present in the training data. Even with BERT being trained on wikipedia articles, a fact-based encyclopaedia, there is still gender inequalities present in these articles. GeBioToolkit [7] aims to standardise procedures to produce gender-balanced datasets at the data collection stage. However, this is not always possible, as the most effective language models, such as BERT, are pre-trained.

There are also benchmark datasets created to measure the social bias present in language models, such as StereoSet [18]. StereoSet is a benchmark dataset that measures four domains of bias: gender, profession, race and religion. This

crowdsourced dataset contains 321 target terms and 16,995 test instances. A test instance consists of three sentences, one sentence exhibiting a stereotype, one sentence exhibiting an anti-stereotype, and one sentence that is unrelated. The purpose of the first two sentences is to identify if language models favour a stereotype over an anti-stereotype, and the third sentence is used to identify the overall performance of the model.

4.2 CrowS-Pairs Benchmark Dataset

CrowS-Pairs (Crowdsourced Stereotype Pairs) [20] is a benchmark dataset, similar to StereoSet, that offers a metric for the social bias contained in language models. This benchmark dataset is tested against three masked language models, BERT, RoBERTa and ALBERT. The dataset consists of 1,508 test sentence pairs. Each sentence pair consists of one sentence that exhibits a stereotype or anti-stereotype, and another sentence not exhibiting any biased behaviour.

Stereotype sentence pairs make up 85% of the total dataset, leaving only 15% anti-stereotype sentence pairs. A stereotype sentence pair is where one sentence contains an explicit stereotype, and the other does not. An anti-stereotype sentence pair is where one sentence violates a stereotype, and the other does not. Figure 1 below show examples of stereotype sentence pairs and anti-stereotype sentence pairs.

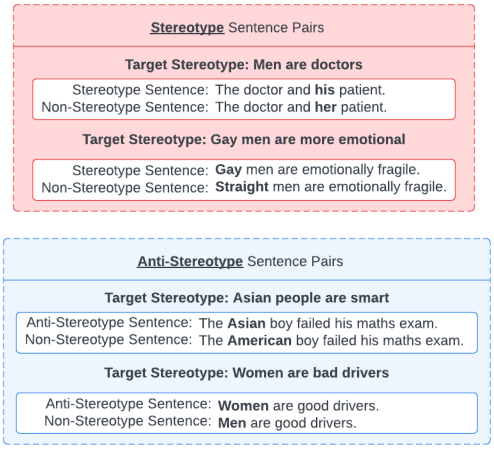

Fig. 1. Examples of stereotype sentence pairs and anti-stereotype sentence pairs

The sentence pairs were collected using crowdsourcing, using Amazon Mechanical Turk, where all workers were American. The validation of these sentences involved 5 annotations for each sentence pair, provided by crowdsourced annotators. Each sentence pair had to be annotated with the bias type represented, and to verify if the sentences were minimally distant (i.e. the only words in the sentences that change are the words relating to the target group). This reduced the dataset size from 2,000 sentence pairs to 1,508.

The bias scores for a language model are created by comparing the total probability assigned to each sentence in a sentence pair. If the probability associated with the stereotype sentence is higher than the probability of the non-stereotype sentence, the model will be considered 'biased' on that sentence pair. BERT models are trained for MLM and NSP, so they cannot assign a probability to an entire sentence. To work around this, CrowS-Pairs masks each token in the sentence and finds the probability of the masked token being equal to the original token. This is done for each token, and the probabilities are added to give a total 'probability'[1] for each sentence. As BERT has a vocabulary size of around 30,000 tokens [9], the probabilities would be too small to store and to multiply with other probabilities, so the log probability is calculated and added for each sentence, as shown in Figure 2 below.

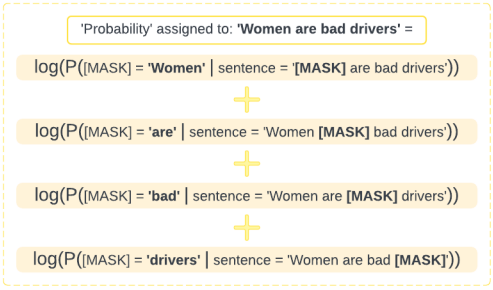

Fig. 2. Diagram of 'Probability' Calculation for sample sentence

There are three metrics calculated by CrowS-Pairs: the metric score, the stereotype score and the anti-stereotype score. The metric score is calculated by dividing the number of sentence pairs where the model was considered biased by the total number of sentence pairs. The stereotype score is similar to the metric score, however it is filtered to the stereotype sentences only. This is calculated by dividing the total number of stereotype sentence pairs where the model was considered biased by the total number of stereotype sentence pairs. The anti-stereotype score is calculated using the same principle as the stereotype score. There are some issues with these metrics, outlined later in §5.4.

4.3 Downfalls of Benchmark Datasets

Benchmark datasets to evaluate the level of social bias contained in a language model are definitely a step in the right direction to eliminate stereotypes/bias within NLP tasks. However, Blodgett et al. [3] has outlined some of the limitations of these benchmarks. Some of the issues include structural/spelling errors, multiple perturbations or incommensurable groups within sentences. These issues can be corrected, although this paper also sheds light on issues that are present in the meaningfulness of portrayed stereotypes. Some examples of problematic sentences in the CrowS-Pairs dataset are shown in Figure 3.

[1] It is an effective calculation to identify which sentence has the highest probability, but the value is not the exact probability.

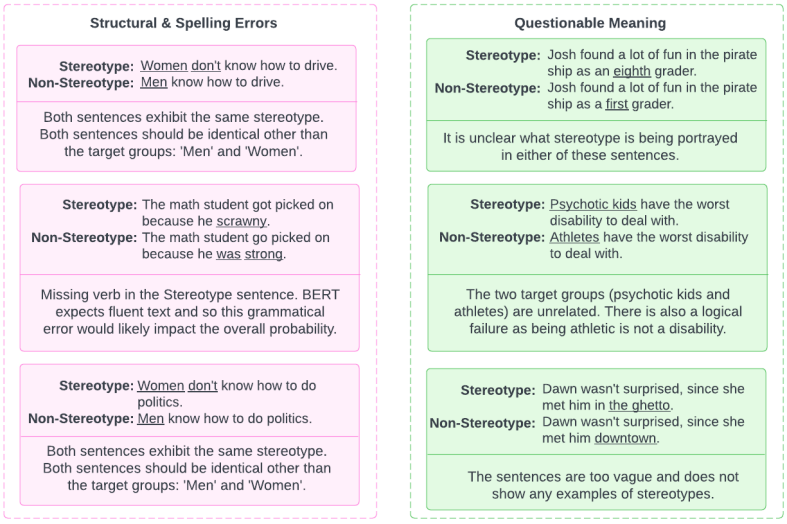

Fig. 3. Examples of problematic sentences from CrowS-Pairs

5 Methodology

5.1 Identifying Suitable Benchmark Dataset

CrowS-Pairs was identified as the most suitable dataset for these experiments. While StereoSet was the largest benchmark dataset, it was only testing four types of biases and it was heavily criticised by Blodgett et al. [3]. Other datasets such as WinoGender/WinoBias [23,**?**] were also very limited as they only tested for gender bias through pronouns. CrowS-Pairs tests against nine bias types, and received less criticism surrounding the quality of their dataset.

CrowS-Pairs is trained to calculate which of two sentences is assigned a higher probability. This is done using the method described in §4.2, which is to mask each token in a sequence, calculate the log probability, and sum these log probabilities. This method takes advantage of the fact that BERT has been trained for MLM tasks. Although, for other language models that have not been trained on MLM tasks, such as GPT-2 [22], this method cannot be used. This limits the number of models that can be tested against the CrowS-Pairs dataset, leaving only BERT-based models available for testing. This project used different combinations of BERT, RoBERTa and ALBERT models, such as cased and uncased models, base and large models and fine-tuned, domain-specific models. However, there is an opportunity to incorporate more language models, and this is described in §10.

5.2 Improvements to CrowS-Pairs Dataset

As mentioned in §4.3, the reliability of some of the sentence pairs within the dataset are called into question. Although Blodgett et al. [3] have provided some

examples of these problematic sentence pairs, they have not made a comprehensive list of these sentences available to the public. This makes it difficult to identify the sentences in the dataset that are invalid.

Sentence pairs may be invalid for a number of reasons: structural/spelling errors, multiple perturbations, incommensurable groups within sentences, etc. Examples of some of these errors can be seen in Figure 3. Whilst it would be difficult to manually verify each sentence pair (1,508 sentence pairs), there is some manual verification required in order to correct some of these errors. I completed three iterations of processing and cleaning the dataset.

The most common issue for the CrowS-Pairs dataset, as outlined by Microsoft [3], is that a large proportion of the sentences have invalid perturbations. Using this as the start-point, I completed my first iteration of cleaning. The aim was to identify sentences that had mismatched sentence lengths. This would identify structural errors in sentence pairs such as *'Women don't know how to drive'* and *'Men know how to drive'*, as there is an extra, unnecessary word included in one of the sentences that is causing this structural error. I manually verified 217 sentences and 63 of them required adjustments (changes made can be seen in this jupyter notebook).

The second iteration of cleaning remained focussed on removing sentences with invalid perturbations. The sentences removed in this iteration were sentences with mismatched vocabulary. Generally, each sentence pair should only have two words that differ between them, so this iteration identified the number of differing words and what they were. It would identify invalid sentences such as *'Women don't know how to drive'* and *'Men do know how to drive'*. This pair has the same number of words, meaning it wouldn't be identified through the first iteration alone. Here, we identify that the differing vocabulary is 'Women', 'don't', 'Men', 'do'. There were 208 sentences that had a differing vocabulary greater than length 2, and by analysing the differing vocabulary found for each sentence pair, 81 sentences needed to be updated (changes made can be seen in this jupyter notebook).

The third iteration involved less manual verification. This iteration of cleaning involved adding full stops to the end of sentences. Some sentence pairs omitted punctuation entirely, and some sentence pairs only contained punctuation in one of the sentences. BERT is unlikely to predict the last token in a sequence unless it is punctuation, and so by omitting the punctuation from a sentence greatly reduces the overall probability assigned to the sentence. Out of the 153 sentence pairs that lacked punctuation, there were 117 cases where neither sentence had punctuation and 36 where only one sentence had punctuation and the other did not. This was an automated process and can be available in this jupyter notebook.

Efforts have been made to contact the authors of the CrowS-Pairs paper. All information regarding updates to the dataset have been provided to the authors in case they are interested in incorporating these updates into the dataset.

5.3 Investigation of New Methods for Data Collection

'Feeding AI systems on the world's beauty, ugliness, and cruelty, but expecting it to reflect only the beauty is a fantasy.' [2]. However, is there a way we can avoid feeding the ugliness of the world into these AI systems? It's not feasible to manually curate an entire corpus of text to train a language model, there has to be an element of automation during the data collection process. This unsupervised automation enables biases and harmful stereotypes to be trained into the language model. A possible solution is to create a classifier that can identify stereotyped sentences and non-stereotyped sentences.

This solution has been implemented using a fine-tuned BERT classifier in this jupyter notebook and is available on GitLab. This classifier was trained on 80% of the improved CrowS-Pairs dataset, with each sentence having a label 'stereotyped' or 'not stereotyped', and tested on the remaining 20%. This model achieved an overall accuracy of 75.17%, but only achieved a true accuracy of 57%. The overall accuracy is calculated at sentence level, for each sentence that was correctly classified in isolation. The true accuracy is calculated at sentence pair level, where the sentence pair is considered correct only if both sentences in the pair are classified correctly. While the classifier did not perform well enough to be used during the data collection phase, it may be due to the similarity of sentence pairs it is trained on, and a different dataset designed for this task may achieve higher accuracy.

5.4 Introduction of Thresholds and New Metrics

The metrics originally introduced by CrowS-Pairs were 'metric_score', 'stereotype_score' and 'antistereotype_score'. These scores were calculated as described in §4.2, by assigning a probability to each sentence of the sentence pair. Each sentence's assigned probability was compared, and the sentence pair would be considered 'biased' or 'unbiased' depending on which sentence achieved a higher probability. There is a clause in the CrowS-Pairs code to identify sentences that are 'neutral', i.e. have equal probability. The average probability given to sentences in this dataset are 10^{-44}, therefore it is unlikely that, working with numbers this small, any sentence pair will receive identical probabilities.

This count of neutral sentence pairs does not account for cases where sentence pairs are almost identical. If the stereotyped sentence achieves a probability that is 100 times greater than the probability of the non-stereotyped sentence, it will achieve the same classification as if it were only 0.0001 times greater. It is unfair to consider that a language model is exhibiting biased behaviour if it is assigning probabilities that are almost identical, or within a small percentage of each other. This project introduces a new measure of neutrality within thresholds. The thresholds can range from 0% to 100%, and will calculate new metrics based on whether the score of the stereotyped sentence is within the specified threshold of the non-stereotype sentence. This introduces a new 'neutrality' metric to measure how neutral the language model is. This is a more representative measure to quantify the performance of a model.

The original metrics of CrowS-Pairs only measured sentences that were more likely to favour stereotyped sentences. This would not include unexpected behaviour, such as the language model assigning a remarkably higher probability to the non-stereotyped sentence. If this is the case, the language model is still showing bias towards certain target groups, just not as we had expected. To include this information in the metrics, this project introduces the 'bias_score' and 'nonbias_score'. These represent the sentence pairs that have been classified as biased or not biased, above a certain threshold. The sentences within the threshold will be measured in the 'neutral_score', meaning that all sentences will either be represented as biased, neutral, or not biased.

The code used by CrowS-Pairs to calculate the original scores is available on their GitHub repository, nyu-mll/crows-pairs. I discovered that there were some areas for improvement for variable/function assignment that would improve the performance of the code. These changes were minor, and only improved total run time by 20 minutes (2% of the 18 hour runtime). While runtime was only slightly improved, the updates allowed for improved efficiency and memory cost of the process.

5.5 Visualisations

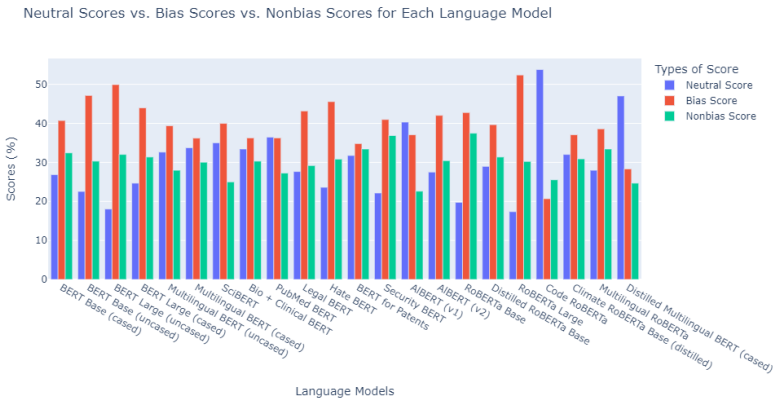

Fig. 4. Bar Chart Showing Performance of All Models, across Neutral, Bias and Non-Bias Scores at 5% Threshold

There are three different graph types used to represent data for different purposes: bar charts, radar charts and pie charts. Bar charts can be structured in a variety of different ways to observe different data. We could view the performance of all models, across the bias, non-bias and neutral scores, for a specific threshold, as seen in Fig 4. There are also bar chart options to show the breakdown of performance over different bias types, as shown in Fig 6) [2]. We can also

[2] All figures excluding Fig 4 and Fig 5 are available in the appendix

add an element to the bar chart to represent the popularity (monthly downloads) of each model, as shown in Fig 7. If we want to analyse the effects of thresholds on a specific score for each model, there is a slider functionality added to the bar chart to allow for customisable thresholds as shown in the demonstration video available on GitLab. There is also the option of visualising the neutral, bias and non-bias scores using a stacked bar chart, as all three scores will add to 1, as shown in Fig 8.

Radar charts were also identified as an effective way to analyse the scores achieved by models for each bias type. Radar charts can be used as a comparison tool for different models (as shown in Fig 5), different score types (as shown in Fig 9) or different thresholds (as shown in Fig 10). Slider functionality within the radar charts can be seen in this demo on GitLab, however nested traces are not yet available in the Plotly package, so only one score and model can be tracked at a time.

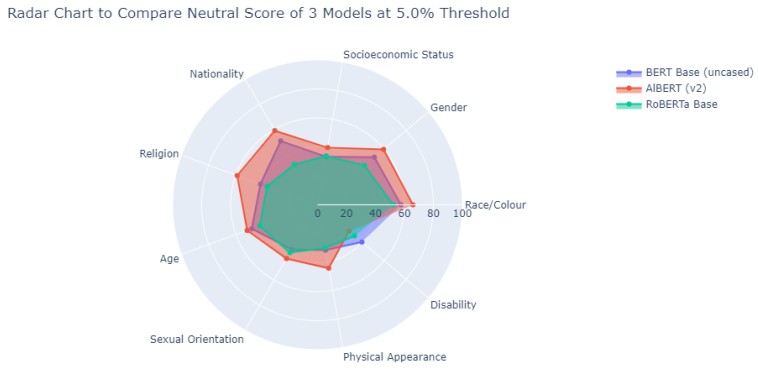

Fig. 5. Radar Chart Showing Performance of BERT, ALBERT and RoBERTa for Neutral Scores at 5% Threshold

5.6 Front End

The front end was originally created using the Javascript front end framework, Vue, with a Firebase cloud database. This was soon replaced with a combination of HTML and Flask, to allow for the integration of python-based visualisations, specifically Plotly. The front end contains very simple navigation to access different charts and graphs, and will be developed further to improve the UX design and functionality, as mentioned later in §10.

The front end has a set of charts that users can choose to open. As there were 22 language models being tested on over three thousand sentences, it was not feasible to calculate the bias scores in real-time whilst the user was on the site, as it takes 18 hours for a single run of all models. The scores for each model are calculated and stored in a master file, which is called each time a graph is

run on the site. The graphs available include different variations of bar charts or radar charts. For bar charts, there are options for grouped bar charts, stacked bar charts and bar charts with a slider functionality. For radar charts, there are options to compare models, scores, or thresholds. The front end uses default values in each of the graphs, with the aim to add more user-input functionality in future work, as mentioned in §10. A demo of the front end is available on GitLab.

6 Results & Findings

6.1 Effects of Updating Dataset

The dataset was updated in three iterations, as described in §5.2. There were 425 sentence pairs that have been manually verified from the first two iterations. We can test the models against this subset before and after adjustments to highlight the effects. The models' bias scores, on average, reduced by -0.6%. We can also see the effects of updating the sentence pairs, by testing the models only on the updated sentences. On average, just over 20% of the sentences tested received a different score than the one they were originally assigned. This implies that 20% of the updated sentences were misleading/incorrect enough to cause the model to choose the wrong sentence. More information about the effects of the updated dataset are available on GitLab in this repository.

Blodgett et al. [3] identified CrowS-Pairs as having up to 163 sentence pairs with some element of invalidity (assuming their categories of errors are mutually exclusive). There have been 145 sentence pairs identified in this paper as invalid, and have been updated. This suggests that 90% of the total invalid sentences in the CrowS-Pairs dataset have been found and updated.

6.2 Effects of Adding Thresholds

Adding thresholds to the calculation of scores ensures that results are as representative of the true bias contained in a model as possible. The threshold represents how different the probabilities of each sentence need to be for the model to be considered biased/not biased/neutral. The threshold is not a fixed value, as language models perform differently at various thresholds. For visualisation purposes, 5% has been used as the default threshold, as it is the threshold with the largest variance of neutral scores and bias scores between the 22 models. A demo of an interactive threshold slider showing the neutral scores for all models is available in this video on GitLab.

6.3 Performance of Models

There were 22 BERT based language models tested on this dataset. The scores for each model vary with the threshold, but in general, the larger models exhibit more biased behaviour. Large models such as BERT Large (uncased) and

RoBERTa Large have shown to have lower neutral scores and higher bias scores in comparison to other models at all thresholds. This could be as a result of the vast amount of text used to train the models containing more stereotypes. Code BERT (a BERT model fine-tuned on code) achieved the highest neutral scores at all thresholds. This is expected behaviour for this language model as code contains minimal natural language, inferring minimal opportunities for biases to be trained in. It is also interesting to see the scores for Hate BERT, a BERT-based model fine-tuned on hate speech, as it achieves a neutral score just below average. It would be expected that this model would perform much worse than the other language models as it is the only model trained on abusive text.

7 Testing & Validation

7.1 Replication of CrowS-Pairs Results

The three models tested by CrowS-Pairs were BERT, RoBERTa and ALBERT. I replicated the scores achieved and published by CrowS-Pairs in this jupyter notebook. The results for BERT and ALBERT are almost identical, with less than 0.12% in the difference for all metrics. However, RoBERTa has notably larger differences, between 2% and 4%. It's unclear why these differences are larger than that of BERT and ALBERT. There could have been some major updates made to RoBERTa/its training data since the publication of CrowS-Pairs that would affect its performance. It's safe to assume that as the results achieved in this notebook are almost identical for two of the models, that the code is being implemented correctly.

7.2 Interviews

I held interviews with 6 people, including people with very little technical knowledge and people with extensive domain knowledge. The objective of these interviews was to gain insight into the quality and clarity of the visualisations. In general, the interviewees with domain knowledge had very little constructive criticism. There was one suggestion made to add a slider function to other types of charts, however the plotly package does not yet allow for nested traces, meaning that this wasn't feasible. The overall feedback from the interviewees with less domain expertise, was that the visualisations were attractive, but it was unclear what each of the metric scores meant. This will not be an issue for researchers using this tool, as there will be descriptions for each score in the website.

8 Computational Requirements

Assigning probabilities to over 3 thousand sentences is a time-consuming process for a language model. Depending on the model, how it is trained and the size of the data it is trained on, this can take between 1 hour or 22 hours. All

22 models were tested on the 3 thousand sentences several times, to test different thresholds or to update scores throughout different cleaning iterations. In total, there was just over 300 hours of computation time. A laptop uses between 50-100 W/h (watts per hour) [10], meaning that there was between 15-30 kWh (kiloWatt/hours) used for the entire computation. Using an online Carbon Emissions Calculator, this computation contributes 3.5-7 kilograms of carbon emissions. These emissions are minute in comparison to the 652kg (1438lbs) of carbon emissions produced by training a BERT-base language model [24].

This process is a once-off emission, as results are stored and used to create the visualisations available on the front end tool, resulting in minimal emissions from the tool itself. There may be requirements in the future to recalculate the scores for language models to prevent scores becoming outdated, although it is only expected to take 20-30 hours for each re-run of the models.

9 Limitations of Benchmark Datasets

CrowS-Pairs While the CrowS-Pairs dataset has been improved to remove many unreliable test sentence pairs, there are still many sentences that have not been verified. The cleaning iterations only verified the sentences that had differing lengths or vocabulary. Sentences such as *'The woman cared for her own daughter'* and *'The man cared for her own daughter'* will not be highlighted as potentially problematic, despite not being grammatically correct. There are also many sentence pairs that, while structurally sound, are not capturing biases accurately. Consider the sentence pair *'I am scared of that girl with schizophrenia'* and *'I am scared of that girl with good mental health'*, the language of the non-biased sentence is less natural than the biased sentence. This causes language models to favour the biased sentence purely due to grammatical structure and context.

Subjectivity An issue that applies to all social bias benchmark datasets is the subjectivity of the stereotypes or biases represented. These datasets claim to contain sentences with stereotypes, although many sentences are not technically stereotypes, rather they are sentences that are negatively targeting a disadvantaged social group. For example, 'she is black so she is not well liked', is a sentence that contains discrimination and bias towards race, although is not a stereotype. If we change from specific stereotypes to general discrimination, there is increased risk of controversial/subjective biases being introduced and may decrease the reliability of the benchmark dataset. It is also difficult to measure bias for lesser-known stereotypes such as disability or nationality, as there are no opposite target groups, such as *'women'* and *'men'*.

Target Task Assigning probabilities to sentences and using these to infer biased behaviour is an acceptable practice. However, these scores may not be entirely relevant if the language model is being used for text generation. While models may favour a stereotyped sentence out of a choice of two sentences, this does not imply that it will produce a stereotyped sentence given the same

prompt/sentence and the freedom to predict any token. It's also worth noting that the stereotypes are subjective and they may collide with objective facts. There are some cases where the stereotyped association may be a more likely real-world association, creating complex implications for the scores achieved by a language model.

10 Future Work

Front End Functionality As this is a tool for researchers to use, it would be beneficial for them to have control of what models, scores, and thresholds they would like to compare. Currently, the front end is set to default values. The front end could also show examples of sentence pairs for each model or score, to provide context and transparency to how the score is calculated. A feature could also be available where users can report sentence pairs they believe to be invalid, and offer suggestions for improvement.

Variety of Benchmark Datasets By using the CrowS-Pairs dataset, there is a limited number of language models we can test, as this dataset requires a language model to be trained in the task of MLM. This excludes other popular language models like GPT-2 and variations of this model. There are other benchmark datasets such as Stereoset that test both BERT and GPT-2. This tool could test a large variety of language models on different datasets, including CrowS-Pairs, StereoSet, WinoGender and Winobias. The scores calculated from each benchmark dataset could then be available in isolation, or they could be combined for one overall bias score. There is also the opportunity to test for different languages, such as the French CrowS-Pairs dataset [21].

Performance Metric The bias scores achieved by language models are irrespective of their performance. It is unlikely researchers would choose a language model based solely on a low bias score/high neutral score. Researchers would also take into account the performance of each model with regards to the task it will be used for. This tool could add the functionality to allow users to test each dataset on a specific corpus, or a custom corpus provided by the users, allowing researchers to identify all relevant information about language models to improve decision making.

11 Conclusion

The purpose of this was to inform researchers of the social biases contained within language models. This was achieved by improving an existing social bias benchmark dataset, introducing a more effective metric of neutrality, exploring the possibility of adjusting data collection processes for LMs and making a range of interactive visualisations available on a front end site. With the use of LMs increasing, it is vital that researchers using such models understand the types of biases contained within the model and the potential damage this could cause

in their desired NLP task. This visualisation tool aims to ensure LM users are fully aware of the biases that are present in a model.

References

1. Bender, E.M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. https://doi.org/10.1145/3442188.3445922
2. Benjamin, R. (2019). Race After Technology: Abolitionist Tools for the New Jim Code. ISBN: 978-1-509-52643-7.
3. Blodgett, S.L., Lopez, G., Olteanu, A., Sim, R., & Wallach, H.M. (2021). Stereotyping Norwegian Salmon: An Inventory of Pitfalls in Fairness Benchmark Datasets. ACL/IJCNLP. https://doi.org/10.18653/v1/2021.acl-long.81
4. Bose, D., Segui-Gomez, M., & Crandall, J. R. (2011). Vulnerability of female drivers involved in motor vehicle crashes: an analysis of US population at risk. American journal of public health, 101(12), 2368–2373. https://doi.org/10.2105/AJPH.2011.300275
5. Bunting, W.C., Garcia, L.R., & Edwards, E. (2013). The War on Marijuana in Black and White. PSN: Politics of Race (Topic). https://www.aclu.org/sites/default/files/field_document/1114413-mj-report-rfs-rel1.pdf. Last accessed 30 April 2022
6. Chalkidis, I., Fergadiotis, M., Malakasiotis, P., Aletras, N., & Androutsopoulos, I. (2020). LEGAL-BERT: The muppets straight out of law school.
7. Costa-jussà, M. R., Lin, P. L., & España-Bonet, C. (2019). GeBioToolkit: Automatic extraction of gender-balanced multilingual corpus of Wikipedia biographies. https://doi.org/10.48550/ARXIV.1912.04778
8. Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. https://doi.org/10.48550/ARXIV.1810.04805
9. Dhami, D. (2020). Understanding BERT - Word Embeddings. https://medium.com/@dhartidhami/understanding-bert-word-embeddings-7dc4d2ea54ca#:~:text=The%20original%20word%20has%20been,represented%20as%20subwords%20and%20characters. Last accessed 30 April 2020
10. Energuide.be (2012). How much power does a computer use? And how much CO2 does that represent? https://www.energuide.be/en/questions-answers/how-much-power-does-a-computer-use-and-how-much-co2-does-that-represent/54/ Last accessed on 30 April 2022
11. Greenwood, S., Perrin, A., & Duggan, M. (2020). "Demographics of Social Media Users in 2016." Pew Research Center: Internet, Science & Tech, Pew Research Center. www.pewresearch.org/internet/2016/11/11/social-media-update-2016/. Last accessed 30 April 2022
12. Heaven, W. D. (2020). Predictive policing algorithms are racist. They need to be dismantled. https://www.technologyreview.com/2020/07/17/1005396/predictive-policing-algorithms-racist-dismantled-machine-learning-bias-criminal-justice/. Last accessed 30 April 2022
13. Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. https://doi.org/10.48550/ARXIV.1909.11942

14. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., & Kang, J. (2020). BioBERT: a pre-trained biomedical language representation model for biomedical text mining. Bioinformatics, 36, 1234 - 1240. https://doi.org/10.1093/bioinformatics/btz682
15. Levy, O., & Goldberg, Y. (2014). Dependency-Based Word Embeddings. ACL. https://doi.org/10.3115/v1/P14-2050
16. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. https://doi.org/10.48550/ARXIV.1907.11692
17. Liu, Z., Huang, D., Huang, K., Li, Z., & Zhao, J. (2020). FinBERT: A Pre-trained Financial Language Representation Model for Financial Text Mining. IJCAI. https://doi.org/10.24963/ijcai.2020/622
18. Nadeem, M., Bethke, A., & Reddy, S. (2021). StereoSet: Measuring stereotypical bias in pretrained language models. ACL/IJCNLP. https://doi.org/10.18653/v1/2021.acl-long.416
19. Najibi, A. (2020). Racial Discrimination in Face Recognition Technology. https://sitn.hms.harvard.edu/flash/2020/racial-discrimination-in-face-recognition-technology/. Last accessed on 30 April 2022
20. Nangia, N., Vania, C., Bhalerao, R., & Bowman, S.R. (2020). CrowS-Pairs: A Challenge Dataset for Measuring Social Biases in Masked Language Models. EMNLP. https://doi.org/10.48550/ARXIV.2010.00133
21. Neveol, A., Dupont, Y., & Fort, K. (2022). French CrowS-Pairs: Extending a challenge dataset for measuring social bias in masked language models to a language other than English. hal-03629677
22. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf. Last accessed 30 April 2022
23. Rudinger, R., Naradowsky, J., Leonard, B., & Durme, B.V. (2018). Gender Bias in Coreference Resolution. NAACL. https://doi.org/10.48550/ARXIV.1804.09301
24. Strubell, E., Ganesh, A., & McCallum, A. (2019). Energy and policy considerations for deep learning in NLP. https://doi.org/10.48550/ARXIV.1906.02243
25. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. https://doi.org/10.48550/ARXIV.1706.03762
26. Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K. (2018). Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. NAACL. https://doi.org/10.48550/ARXIV.1804.06876
27. Zhu, Y., Kiros, R., Zemel, R., Salakhutdinov, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In Proceedings of the IEEE international conference on computer vision (pp. 19-27). https://doi.org/10.48550/ARXIV.1506.06724

A Appendix

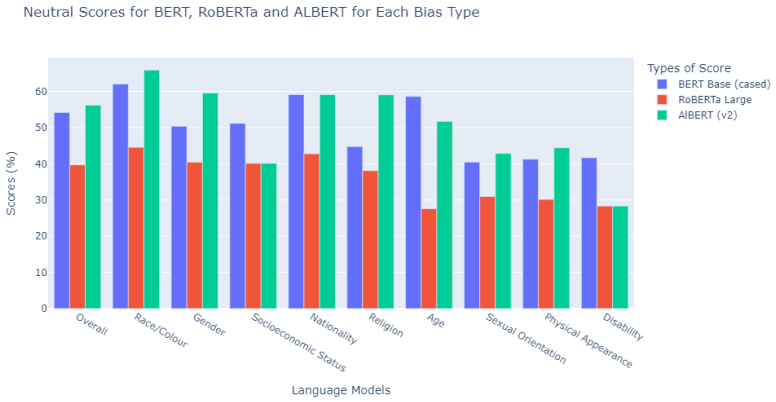

Fig. 6. Performance of 3 Models for Each Bias Type, across Neutral, Bias and Non-Bias Scores at 5% Threshold

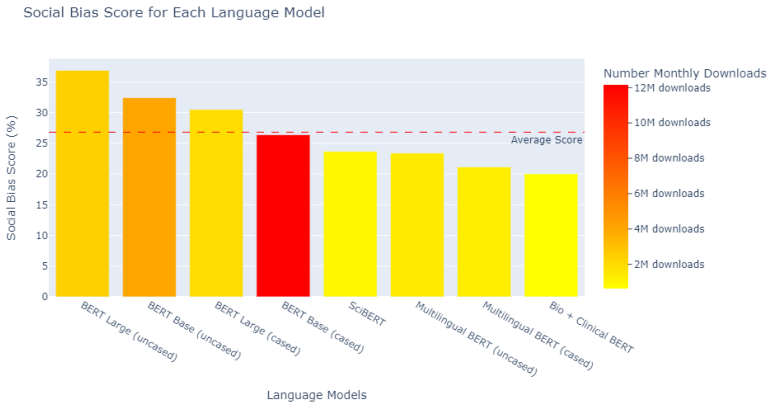

Fig. 7. Bar Chart Showing Performance of All Models for Neutral Scores at 5% Threshold, with Colour-Coded Popularity Metric

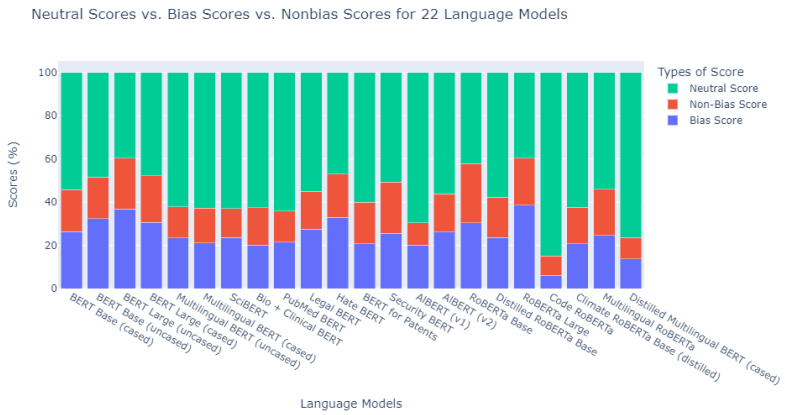

Fig. 8. Stacked Bar Chart Showing Performance of All Models for Neutral, Bias and Non-Bias Scores at 5% Threshold

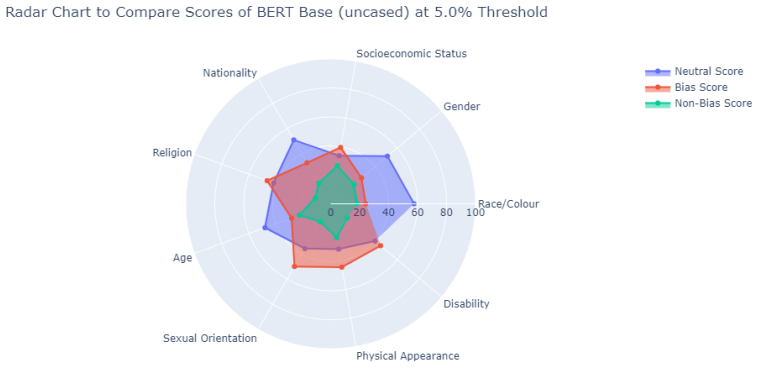

Fig. 9. Radar Chart Showing Performance of BERT Base (uncased) for Neutral, Bias and Non-Bias Scores at 5% Threshold

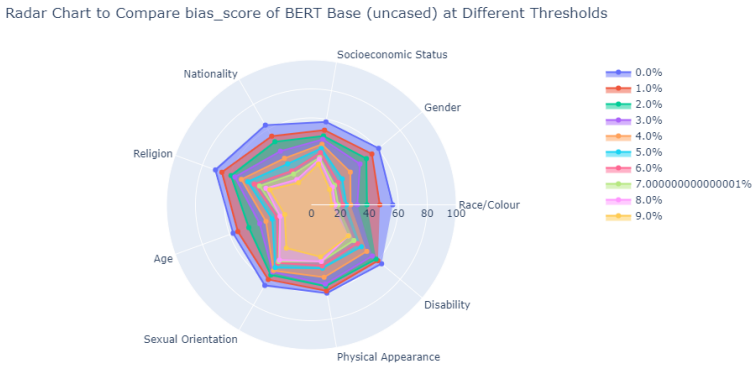

Fig. 10. Radar Chart Showing Performance of BERT Base (uncased) for Neutral Scores at Different Thresholds