Using Bidirectional Transformer Neural Networks for Advancing Gender Bias Recognition in STEM Job Advertisements

Nina Van Zandweghe

Received February 04, 2024
Accepted June 29, 2024
Electronic access July 15, 2024

Gender disparity persists in most STEM fields, where men vastly outnumber women. Recent research shows that gendered wording in advertisements for STEM jobs may have a significant influence on the appeal of a job to potential female applicants. To increase awareness and potentially address such gender bias in STEM job ads, it is important to identify the bias as clearly and accurately as possible. Research by Gaucher et al. (2011) has attempted to predict gender bias in STEM job ads by using a simple word-counting and summation approach. This paper fine-tunes a deep learning language model, Google's Bidirectional Encoder Representations from Transformers, for the specific task of recognizing biased language in STEM job ads by training the altered model on a large dataset of STEM job advertisements. This machine learning model has the advantage of a more holistic semantic understanding of the language in these ads due to its use of transformer neural networks. The analysis yields two main results. First, the ads are biased toward masculine language, in line with the findings of previous research. Second, the machine learning approach predicts gender bias in STEM job ads with high accuracy, compared to a benchmark of random prediction, and is able to outperform the word-counting and summation approach.

Introduction

Women are consistently underrepresented in STEM fields, particularly in engineering and computer science. This underrepresentation has been attributed to a variety of causes that focus primarily on the psychological and social aspects of the problem: differences in confidence level, prevailing societal stereotypes, or the lack of role models in these occupations [1]. However, recent research suggests that gendered wording in job recruitment materials has a notable influence on the appeal of a job to potential female applicants, thereby ultimately influencing the number of women in the STEM workforce [2]. Gendered wording refers to subtle and often unconscious differences in word choice that may appeal to certain job candidates.

A groundbreaking paper by Gaucher et al. (2011) provides evidence that gendered wording in job ads may influence potential candidates' perception of a job's inclusiveness, sense of belonging, and overall job appeal [3]. To measure gendered wording, the paper introduces a statistical vocabulary method in which the number of masculine-coded and feminine-coded words are summed for each advertisement in a variety of fields. Through this word-matching and summation method, advertisements with a majority of masculine-coded words were said to be biased towards male candidates while advertisements with a majority of feminine-coded words were said to be biased towards female candidates. The predetermined list of coded words is based on and consistent with prior research studying gender coded words in the English language [4]. Inevitably, this relatively simplistic approach of relying on a statistical count of a set list of coded vocabulary is prone to missing some detection of gender bias since this list of words is likely incomplete.

Aside from this current word-matching solution, paid solutions exist as well, but are often closed source and prohibitively expensive [5]. One such company is Textio, a platform whose AI tools give users the ability to incorporate inclusive language into their company's written content.

To raise awareness and potentially address gender bias in STEM job ads, it is important to recognize the bias in such ads as clearly and accurately as possible. In this paper, I revisit the issue of detecting gender-bias in STEM job ads by using Google's Bidirectional Encoder Representations from Transformers (BERT), a novel machine learning tool that utilizes transformer neural networks to gain a deeper semantic understanding of a given job advertisement's language. Transformer models are faster at processing text than many traditional models, such as recurrent neural networks (RNNs) and long-short-term memory models (LSTMs), and can effectively capture long-term dependencies. Bidirectional transformers, unlike unidirectional transformers, can gather information from both preceding and following words at the same time. BERT is currently the industry standard for Natural Language Processing (NLP) and, through semantic analysis, allows for a more holistic understanding of a sentence's meaning. A more complex understanding of sentence structure and word relations can improve the detection of gender bias within STEM job recruitment materials compared to prior word-matching methods. Ji (undated) [6] trains

BERT on data analyst job advertisements to extend the list of gender-coded words of Gaucher et al. (2011)[3] and finds that the gender-coding of some words can depend on their context within a sentence.

This paper fine-tunes BERT and trains the model on a scraped dataset of STEM job advertisements with the ultimate goal of creating a model that can correctly predict if a given ad is masculine-coded, feminine-coded, or neutral.

The results are twofold. First, using the list of gender-coded words from Gaucher et al. (2011)[3] to classify the STEM job ads in my dataset as "masculine-coded," "feminine-coded," or "neutral," I find that the ads are biased toward masculine language, in line with the findings of the previous research. This finding highlights the importance of studying gender bias in such job ads. Second, my main contribution is demonstrating that the use of BERT allows the prediction of gender bias in STEM job ads with high accuracy. Using BERT improves upon the current word-matching and summation method by predicting more accurately whether a STEM job ad is masculine-coded, feminine-coded, or neutral.

The paper proceeds as follows. Section 2 describes the STEM job ad data and provides details of the machine learning methodology. Section 3 presents the results. Section 4 concludes by summarizing the performance and implications of the BERT model.

Material and Methods

A BERT model was selected because it had the opportunity to offer greater semantic understanding than a strictly statistical model. BERT, short for Bidirectional coder Representation from Transformers, is a revolutionary natural language processing model developed by Google in 2018[7]. This model relies on transformers, another groundbreaking innovation released by Google in 2017[8].

Neural Networks

A neural network [Figure 1] consists of a series of layers where each layer is composed of multiple nodes. These nodes are connected to the nodes of the next layer by a network of weights and biases.

The data is first forward-propagated through the input layer and hidden layers, then, in the output layer, the neural network produces a value based on the probability of each possible output. For example, in the experiments performed in this paper, the classification labels feminine-coded, masculine-coded, and neutral were assigned to the output layer. The node with the highest output value, and its corresponding label, would be the model's final prediction for a given job advertisement.

The network's output, y_{pred}, is compared to the actual output, y_i, through a loss or cost function, J(x), that calculates the model's prediction error. The loss function is then minimized using stochastic gradient descent so that, if an incorrect output is predicted, the model backpropagates and adjusts the weights and biases associated with the specific nodes that generated the output, giving a higher importance to the nodes contributing to the correct output and a lower importance to the nodes contributing to the incorrect output. The equation for loss minimization is as follows:

$$\min J(x) = \frac{1}{m}\sum_{n=1}^{m}\left(y_i - w_0 - w_1x_1 - w_2x_2 - \cdots - w_nx_n\right)^2 \quad (1)$$

This process of feedforward propagation and backpropagation continuously repeats until the model achieves the correct weights and thus is able to consistently predict a correct output.

Transformers

Certain neural networks, such as RNNs or LSTMs, were developed to handle sequential data. However, due to their sequential processing nature, RRNs and LSTMs can be prohibitively slow and struggle to capture long term dependencies. In 2017, Google developed a novel neural network architecture called a Transformer, which addressed many of the gaps inherent in previous natural language processing models.

Transformers utilize a mechanism called attention, which allows them to process multiple sequences of data in parallel. This not only enables much faster processing, but also allows them to efficiently handle long range dependencies in text. Particularly, the self-attention mechanism allows a transformer neural network to understand what a particular word refers to in context of the words around it. For example, in [Figure 2, panel A], the model is able to understand that the word "it" corresponds to "The animal." Referred to as multi-headed attention, each of the attention "heads" relies on self-attention to capture differing relationships between a word and the other words in the sentence and process these various relationships and dependencies simultaneously. In [Figure 2, panel B], one attention head puts greater emphasis on the relationship between "it" and "The animal," while a separate attention head focuses on the relationship between "it" and "tired." These components of attention are vital as they allow the model to build an understanding of relationships between words, ultimately contributing to its understanding of a sentence's actual content.

Transformers also utilize word embeddings, which are vector representations of words. These embeddings capture the n-dimensional relationship between words by putting related words in proximity to each other [Figure 3]. Understanding these complex relationships also contributes to the model's comprehensive understanding of natural language.

Transformers have now become a standard for handling natural language processing tasks like text generation, translation,

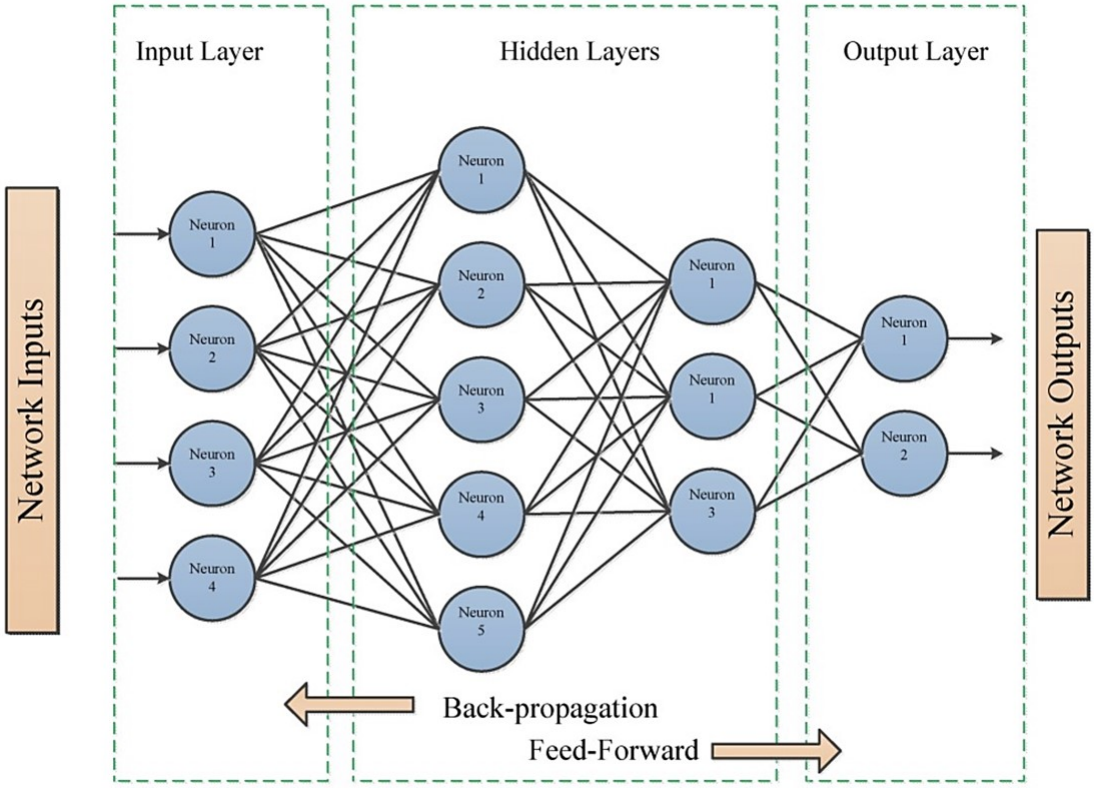


Fig. 1 Neural network Source: Reyneke, J. A. (2023)[9].

and classification. Features such as attention and word embeddings are what allow transformer neural networks, such as BERT-a bidirectional transformer neural network–to gain a comprehensive understanding of a sentence's meaning and effectively perform semantic analysis.

BERT

Figure 4 depicts BERT's architecture with an encoder on the left and a decoder on the right, both featuring layers of multi-head attention and position-wise feed-forward networks, integrated with residual connections and layer normalization. Positional encodings are added to input and output embeddings to maintain sequence information, culminating in output probabilities after passing through linear and SoftMax layers.

BERT's bidirectional capabilities are a defining feature to distinguish it among other transformers. As opposed to unidirectional transformers which apply the self-attention mechanism to only one token at a time, the bidirectional transformer has the ability to gain information from tokens to both the left and right of its position. Bidirectional transformers are therefore better able to capture the nuanced relationships between words in a text and again contribute to the model's comprehensive understanding of the meaning of a text.

BERT has been pre-trained by Google on massive unlabeled text datasets, namely BooksCorpus and Wikipedia. This pre-trained model can then be altered and fine-tuned for specific, language-related tasks. Both BERT's bidirectional attention mechanism and pretraining background make it ideal to work with for text classification tasks. Particularly, BERT's more thorough understanding of a sentence is an enormous asset in the semantic analysis of text, putting it at the technological forefront for Natural Language Processing (NLP) and thereby also making BERT an ideal model to work with for this research. To conduct the research, I fine-tuned my model by adding additional layers to the base BERT model before training the model again and refining it for the task of identifying gender-bias in job advertisements.

Dataset

The dataset I used to train the model is a concatenation of four datasets containing job advertisements scraped from Glassdoor and LinkedIn for the selected STEM careers of data analysts, data scientists, and data engineers. The data sets taken from Kaggle are Data Engineer Jobs, Data Scientist Jobs, Data Analyst

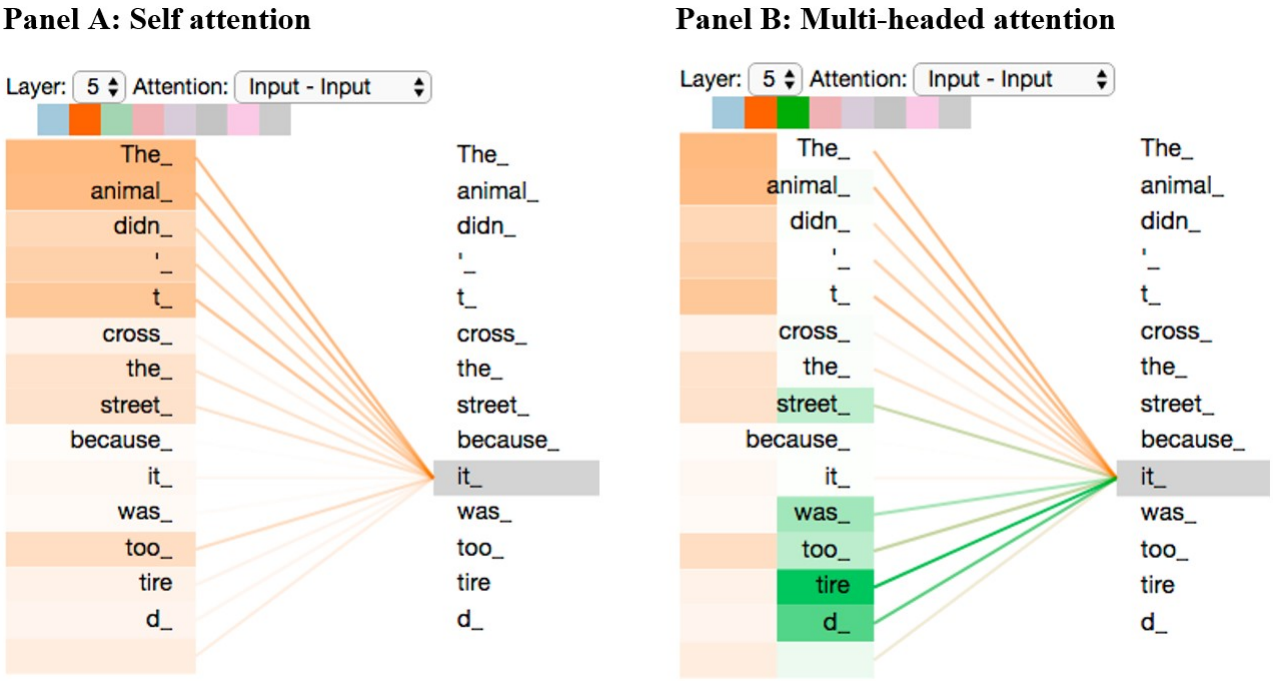


Fig. 2 Attention mechanisms Source: Alammar, J. (2018)[10].

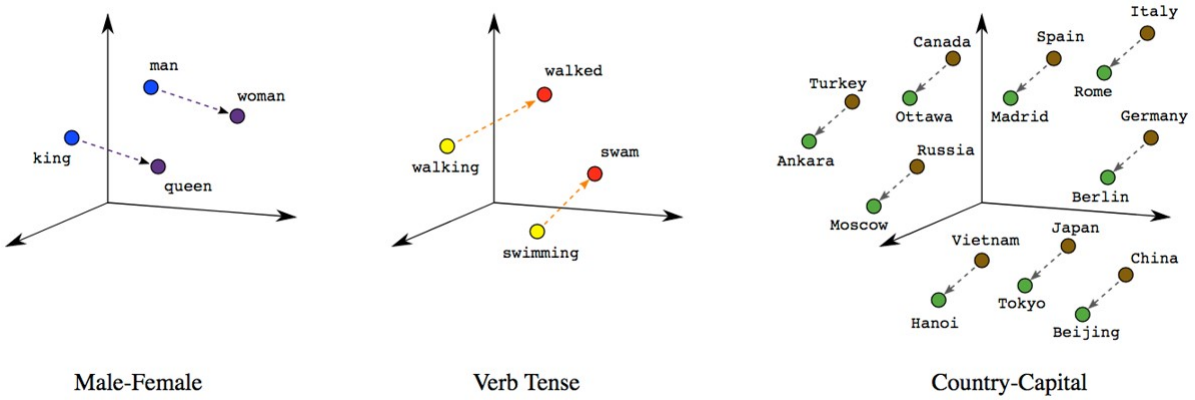


Fig. 3 Word embeddings in a multi-dimensional space Source: Bujokas, E. (2020)[11].

Jobs, and LinkedIn Data Analyst jobs listings. All job advertisements consist exclusively of text in a paragraph form. Before combining, I manually compared random samples from each of the four datasets to ensure a similar overall structure between the job advertisements, particularly in terms of their length. The final combined dataset consisted of 11,535 job advertisements.

As seen in Figure 5, the number of masculine-coded, feminine-coded, and neutral advertisements varied greatly. In particular, there are more than twice as many masculine-coded ads than feminine-coded ones. This finding confirms the gender bias in STEM job ads, which is consistent with findings of previous research[2].

Data Preprocessing

The Pandas and Numpy libraries were utilized for reading and preprocessing the dataset, including balancing data, modifying label categories, and restructuring data columns. Any characters

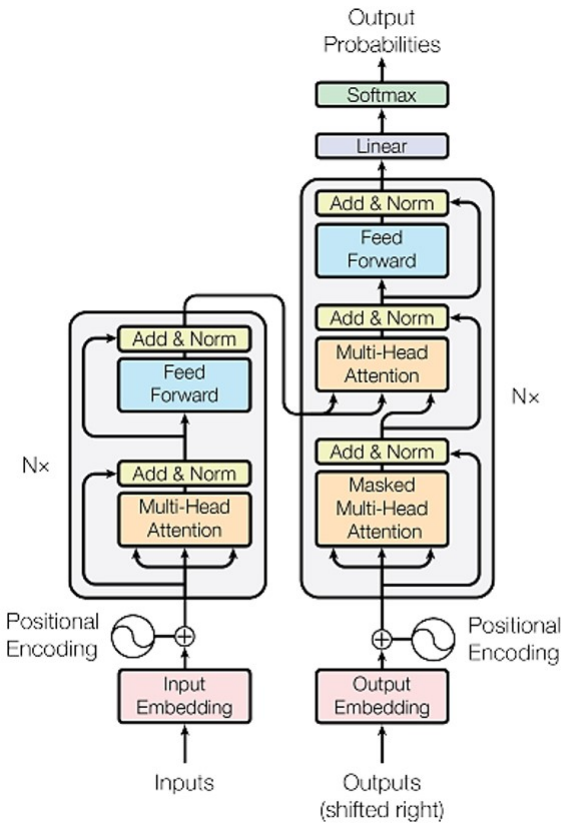


Fig. 4 BERT architecture. Source: Vaswani et al. (2017)[8].

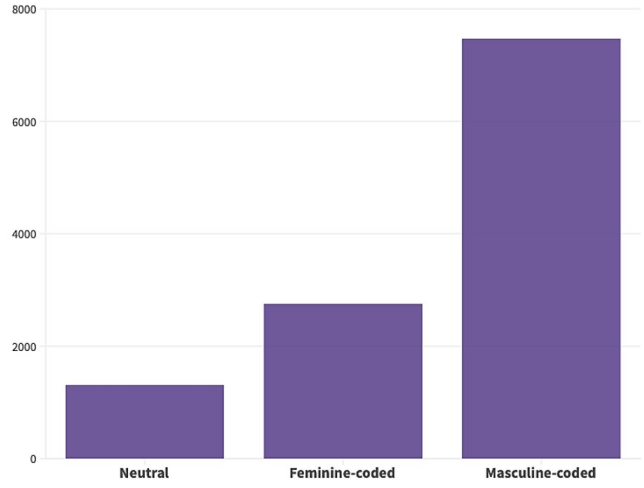


Fig. 5 Distribution of ads in the full dataset

within the text that are not part of the standard English alphabet (for example, '*n*,' '*t*,' etc.) were also removed.

The Scikit-learn (sklearn) library was used to encode the categorical labels into numerical format as well as to split the dataset into training, validation, and test sets[12]. To facilitate processing through the neural network, each label was converted into a corresponding numerical value: feminine-coded labels were assigned the value "0", masculine-coded labels were assigned the value "1", and neutral labels were assigned the value "2". The advertisements in the newly labeled dataset were then randomly shuffled and split into three categories, allocating 70% of the advertisements for train, 15% for test, and 15% for validation. Each of these subsets provides a distinct purpose in the training and evaluation of the model.

The train dataset consists of the advertisements that the model uses to learn the patterns and correlations between adverts and their respective labels. The validation dataset is employed to fine-tune and evaluate the model during training. The test dataset consists of job advertisements kept separate from the training process so that the model encounters them for the first time during testing. Using its knowledge of the data's relationships acquired from the training and fine-tuning of hyperparameters, the fully trained model is then evaluated on the test subset. The variables of interest given by the results of evaluating on the test dataset include precision, recall, and the F1 score (a weighted measure of the model's accuracy).

Model Training and Refinement

HuggingFace's transformer libraries were used to tokenize the datasets and prepare them for training. The optimal combination of parameters was found after experimenting with both the model parameters and the actual dataset.

In terms of the model, I chose the specific values of the hyperparameters learning rate, batch size, and number of epochs because these values minimized the validation loss function. In other words, these specific values allowed my model to converge at the saddle point for the validation loss.

I fine-tuned the hyperparameters by adjusting them in small increments and observing the effects on the validation loss and accuracy. Epochs refer to the number of times the model processes the entire dataset during training. I tested the model with various epoch counts, ranging from 1 to 10. The learning rate determines the size of the updates to the model's weights and influences the rate at which the model reaches the minimum of the loss function. I adjusted the learning rate incrementally (0.0001, 0.0002, etc.) and monitored its impact on the validation loss, ultimately identifying the optimal rate through experimentation. Similarly, batch size—the number of advertisement samples used in a single training iteration—was fine-tuned. After testing different batch sizes, I found that the standard default value worked best. The development environment used in this research was Pytorch.

The learning rate of the model was 2e-5 and the batch size for both training and evaluation was 16. Ten epochs with no

early stopping was chosen because, in table 4, the accuracy does not significantly improve after seven epochs with an accuracy of 0.863 and peaking at nine epochs at 0.8647. The accuracy even decreases with the 10th epoch from 0.8647 to 0.8642, strongly suggesting that increasing the number of epochs to 15 or 20 would not change the accuracy by much. Secondly, the validation loss function should decrease and reach a minimum. In table 4, the validation loss reached a minimum at three epochs with a validation loss of 0.4674, furthering the conclusion that increasing the number of epochs from 10 would not significantly increase the performance of the model. Although the minimum validation loss was reached after three epochs, I extended the training to 10 epochs due to the trade-off between accuracy and slight overfitting. The model's accuracy continued to improve even after the validation loss hit its minimum, justifying the decision to train for 10 epochs. Additionally, training for only three epochs would have been insufficient, as the model would not have been adequately exposed to the dataset.

DistilBERT, a smaller version of the original BERT model, was selected for its more compact and efficient computation abilities, similar to how BERT base was selected over BERT large for requiring less computational power, and the uncased version was used because the ad processing does not rely on uppercase or lowercase letters[13]. Additionally, the optimizer used was the Adam Optimizer. The best-performing model was selected in terms of overall greatest accuracy, precision, recall, and F1 score.

Different datasets were concatenated and tested in a variety of combinations, varying the number of advertisements that the model was trained on. The results were generally more favorable when the model was trained on a greater amount of data. However, if the advertisements in one dataset were not of a similar length or structure to those in other datasets, then the overall accuracy of the model was negatively affected.

Another significant source of experimentation was the question of whether a balanced or unbalanced train dataset would yield higher accuracy. The following combinations were tested: balanced with respect to all categories through the duplication of feminine-coded and neutral advertisements; balanced dataset only with respect to feminine- and masculine- coded ads through the duplication of feminine-coded ads; balanced with respect to feminine- and masculine-coded ads by cutting the number of masculine ads; and an unbalanced dataset. The results of the first three tests are shown in tables 1–3.

While the accuracies in tables 1-3 are acceptable, ultimately, the unbalanced dataset [Figure 6] had the most favorable results across all measures.

Results

In the training process, the model consistently increased its accuracy with each epoch, as seen in Table 4. The accuracy ultimately increased to around 86%, signaling that the majority of advertisement labels were being correctly predicted within the train dataset.

In regards to the test dataset, the effectiveness of my BERT-based model was individually measured for feminine-coded (0), masculine-coded (1), and neutral (2) ads in terms of their respective precision, recall, and F1 score in each category. Table 5 presents measures of the model's performance on the test dataset.

Overall, the model identifies masculine-coded and feminine-coded job ads with high accuracy, meaning 87% of all predictions are correct. As a benchmark for comparison, randomly predicting that an ad is masculine-biased, feminine-biased, or neutral would be correct 62.8% (1088/1731), 25.1% (435/1731), or 12.0% (208/1731) of the time, respectively. Even the best of these random predictions is well below the model's overall accuracy that takes into account scores for all three labels.

For feminine-coded ads, the model achieved precision, recall, and F1 scores all equal to or greater than 80%. For masculine-coded ads, the model achieved over 90% in all measures of success. The overall accuracy and weighted averages were consistent at 87%. The macro average, which is the arithmetic average across the three types of ads, for all measures was around 81%. The macro average is representative of the test dataset. Though the BERT-based model's scores for neutral advertisements were lower, this result does not have a significant impact, as the identification of neutral job advertisements is not integral to the primary objective of this paper: identifying gender bias.

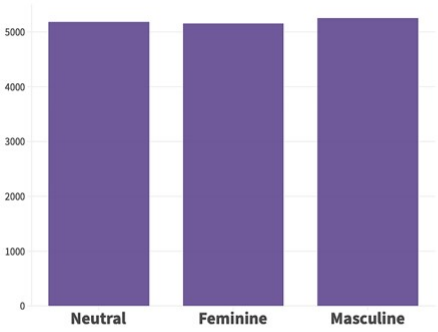
The unbalanced training dataset provides the model with many more masculine-biased than feminine-biased or neutral advertisements to learn from. This imbalance may drive the relatively high precision, recall and F1 score values for masculine-biased ads. Using this logic, it follows that since the number of feminine-biased ads greatly exceeds that of neutral ads in the training dataset, the precision, recall, and F1 score values were higher for feminine-biased ads as well.

Figure 7 displays a confusion matrix of the test dataset's performance results. Of the 1,731 job advertisements, 1,511 were correctly predicted by the model. Figure 7 proves that the model did not simply predict all advertisements to be masculine coded– an inherent risk due to the skew in the number of masculine- vs. feminine- and neutral-coded advertisements in the unbalanced test dataset.

Comparison with Word-Matching Method

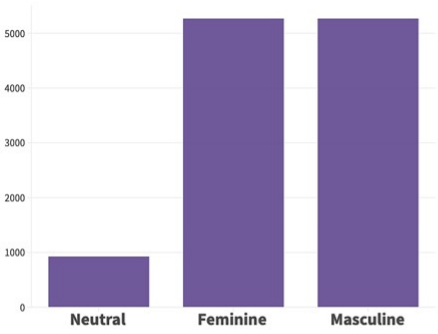
To further verify the accuracy of the BERT-based model, I compare its results with those of Gender Decoder, a tool that implements the current word-matching and summation technique used to detect gender bias in job advertisements[14]. The Gender Decoder tool is inspired by the research of Gaucher et al (2011)

Table 1: Balanced dataset with respect to all categories, through the duplication of feminine-coded and neutral advertisements



Epoch	Training Loss	Validation Loss	Accuracy
1	0.671500	0.577662	0.772254
2	0.492900	0.481638	0.823121
3	0.355300	0.460547	0.832948
4	0.259400	0.476956	0.852601
5	0.189700	0.551212	0.845087
6	0.139200	0.623316	0.855491
7	0.094400	0.723193	0.854913
8	0.063800	0.767432	0.857225
9	0.047400	0.795906	0.862428
10	0.033600	0.824407	0.858960

Table 2: Balanced dataset with respect to feminine and masculine coded ads, through the duplication of feminine-coded advertisements



Epoch	Training Loss	Validation Loss	Accuracy
1	0.687100	0.545491	0.790173
2	0.439100	0.493845	0.821965
3	0.294700	0.546334	0.840462
4	0.213200	0.610960	0.834682
5	0.152900	0.711335	0.834104
6	0.107800	0.805729	0.842197
7	0.069600	0.853035	0.846243
8	0.047000	0.996123	0.828324
9	0.034000	1.019825	0.838150
10	0.025700	1.001381	0.841618

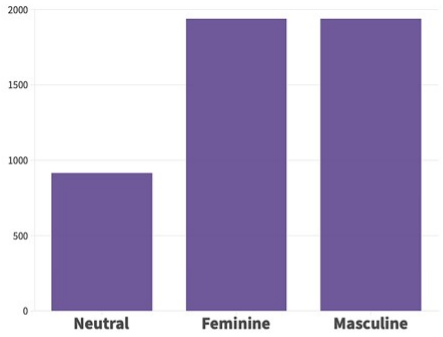
and its list of masculine-coded and feminine-coded words has been compiled based on their research[3].

Table 6 presents two examples of crafted job advertisements that were given to both the BERT-based model and Gender Decoder. In these examples, multiple sentences of neutral job advertisement text were added onto the end to increase the length of the advertisement to a length similar to the advertisements the BERT-based model was trained on. Both example ads are clearly masculine-biased and are identified as such by the BERT model. However, Gender Decoder misclassified both ads as feminine-biased, either due to the presence of feminine-coded words (example 1) or because of the presence of uncoded synonyms of masculine-coded words in the ad (example 2). These examples expose the limitations of relying on a model with a pre-specified list of gender-coded words and demonstrates the capability of the BERT machine learning algorithm to overcome these limitations.

Discussion and Conclusions

The objective of this paper was to determine the efficacy of a BERT based model fine-tuned to detect gender bias in STEM

Table 3: Balanced dataset with respect feminine and masculine coded ads, through cutting the number of masculine ads



Epoch	Training Loss	Validation Loss	Accuracy
1	0.862500	0.625691	0.754335
2	0.628000	0.547326	0.786127
3	0.457500	0.506371	0.808092
4	0.318100	0.579792	0.800578
5	0.209400	0.667005	0.800578
6	0.137100	0.682162	0.815029
7	0.096200	0.918501	0.789017
8	0.065300	0.958587	0.798844
9	0.045900	0.968388	0.809827
10	0.037300	0.993349	0.812717

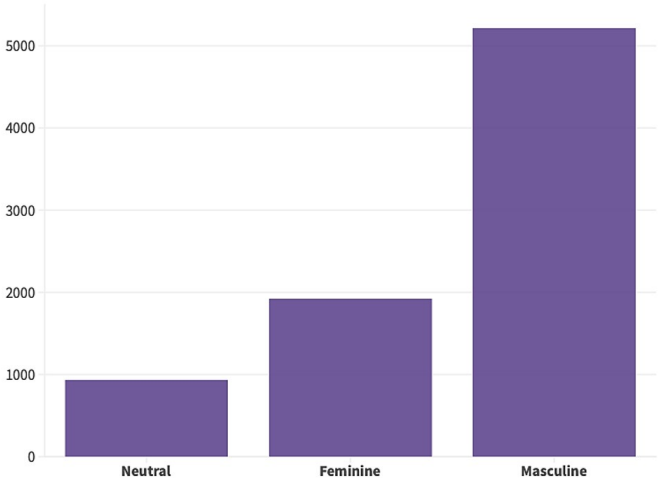


Fig. 6 Distribution of ads in the final training dataset

job advertisements, relative to that of the word-matching and summation technique used in previous research. I used Google's BERT model as a baseline and then fine-tuned it for the detection of gender bias in STEM job advertisements by training the model on a large dataset of job ads.

My BERT-based model achieved a high overall accuracy of 87% . The macro average is consistent across all measures at around 81%. Moreover, the model correctly predicts feminine, masculine, and neutral gender bias in the majority of advertisements. Most notably, detection of masculine-coded language bias in advertisements is predicted with over 90% confidence across all measures (precision, recall, f1 score). These results indicate that transformer neural networks show promise for the identification of gender biased ads.

I also measured the success of the BERT-based model in terms of its performance against the Gender Decoder tool for implementing the word-matching and summation technique. While the Gender Decoder tool incorrectly labeled ads with clear masculine bias and could not recognize close synonyms for biased words, the BERT-based model was able to leverage its greater semantic understanding of language and correctly identify gender bias in those same ads. These examples illustrate the capability of machine learning tools to overcome the limitations of working with a prespecified list of gender-coded words.

Future research opportunities in this field include exploring larger datasets, improving data balancing techniques, investigating alternative model types such as GPT, and addressing alternate forms of bias, such as ethnic biases. This model could also be applied to job advertisements from other fields to explore if significant gender-bias exists beyond STEM careers or remains largely within the field.

With BERT's holistic understanding of natural language and its aptitude for semantic analysis, it is better able to pick up on more subtle gender bias in job advertisements. More sophisticated technology to detect gender bias in job advertisements will ultimately contribute to a more diverse workforce, which is especially significant for women pursuing STEM careers. This research is also particularly relevant for recruiting companies as it encourages awareness of the importance of language in job ads while promoting the use of a more advanced machine learning tool to check their written content.

Table 4: Training progress by epoch (%)

Epoch	Training loss	Validation loss	Accuracy
1	66.80	54.83	78.38
2	48.44	59.68	80.12
3	35.92	46.74	84.91
4	25.49	47.75	84.86
5	18.36	58.72	84.51
6	14.28	62.75	85.26
7	10.26	66.36	86.30
8	7.66	78.52	86.07
9	5.98	76.44	86.47
10	4.61	78.47	86.42

Table 5: Results of test dataset

	Precision (%)	Recall (%)	F1 score (%)	Support
Feminine (0)	85	80	83	435
Masculine (1)	92	94	93	1088
Neutral (2)	66	67	66	208
Accuracy			87	1731
Macro average	81	80	81	1731
Weighted average	87	87	87	1731

Acknowledgments

This research was conducted in Solon, Ohio from June to December 2023. I am grateful to Ethan Haik for his research supervision.

References

1 C. Hill, C. Corbett and A. Rose, *AAUW report*.

2 R. Isidor, M. Wehner, J. Eickhoff and R. Kabst, *Academy of Management Proceedings*.

3 D. Gaucher, J. Friesen and A. Kay, *Journal of Personality and Social Psychology*, **101**, 109–128.

4 M. Newman, C. Groom, L. Handelman and J. Pennebaker, *Discourse Processes*, **45**, 211–236.

5 M. Laguaite, *Everything you should know about Textio*, `https://app.yoodli.ai/blog/everything-you-should-know-about-textio`.

6 M. Ji, undated manuscript).

7 J. Devlin, M. Chang, K. Lee and K. Toutanova, *BERT: Pre-training of deep bidirectional transformers for language understanding*, `https://doi.org/10.48550/arXiv.1810.04805`.

8 A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser and I. Polosukhin, *Advances in Neural Information Processing Systems*, **30**, year.

9 J. Reyneke, *The history and foundations of artificial intelligence*, `https://medium.com/@janelreyneke/the-history-`

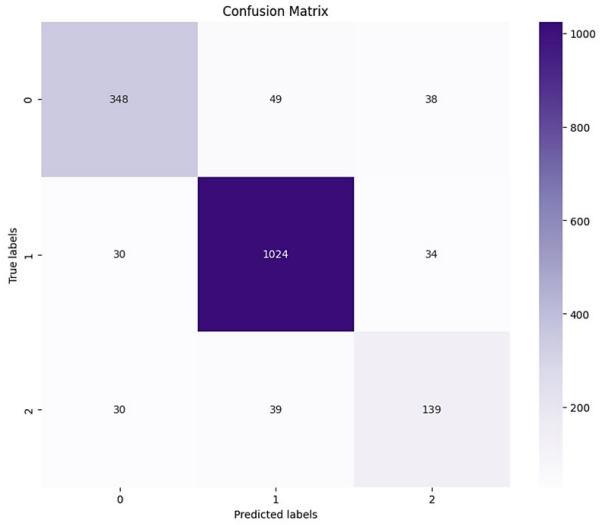


Fig. 7 Confusion matrix: true vs. predicted labels. Feminine (0), masculine (1), and neutral (2)

and-foundations-of-artificial-intelligence-c6f44986e7f.

10 J. Alammar, *The illustrated transformer*, https://jalammar.github.io/illustrated-transformer/.

11 E. Bujokas, *Creating word embeddings: coding the Word2Vec algorithm in python using deep learning*, https://towardsdatascience.com/creating-word-embeddings-coding-the-word2vec-algorithm-in-python-using-deep-learning-b337d0ba17a8.

12 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos and D. Cournapeau, *Journal of Machine Learning Research*, **12**, 2825–2830.

13 V. Sanh, L. Debut, J. Chaumond and T. Wolf, *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*, https://arxiv.org/pdf/1910.01108.

14 G. Decoder, https://gender-decoder.katmatfield.com/.

Table 6: BERT-based model vs. Gender Decoder

Ads/Input	Gender Decoder	BERT-based model
Example 1: Blatantly biased towards masculine job candidates. *"Men Who Engineer is a company exclusively looking for male engineers, data scientists, and programmers. We believe that females should not become engineers because they are overly considerate and sensitive. The men in this company are extremely supportive of each other but would likely not welcome a female engineer as one of their own. The belief that the patriarchy is simply better is deeply ingrained and we would not be able to help but explain simple concepts in a condescending way to any female colleagues. We need male engineers because they are ultimately more independent and ambitious"*	Due to the presence of the feminine-coded words "considerate," "sensitive," and "supportive," the word-matching technology determines this job advertisement to be appealing to female candidates.	The advertisement is correctly identified as biased towards male candidates.
Example 2: Advertisements 2.A and 2.B contain identical text with the exception that all the masculine coded words in 2.A have been replaced by close synonyms in 2.B (see Appendix). These synonyms do not appear on Gender Decoder's list of masculine-coded words. **2.A:** *"Looking for men who are ambitious, courageous, confident, and supportive. This profession lends itself to assertive engineers who are self-confident in their actions as the engineer is expected to frequently interact with the media. A forceful and dominant presence will be a great asset in the situation. A perfect new team member would also be adventurous and willing to explore new possibilities and pursue even risky ideas."* **2.B:** *"Looking for men who are aspiring toward great things, brave, self-assured, and supportive. This profession lends itself to authoritative engineers who are self-assured in their actions as the engineer is expected to frequently interact with the media. A powerful and commanding presence will be a great asset in the situation. A perfect new team member would also be bold and willing to explore new possibilities and pursue even risky ideas"*	**Example 2.A** returns the label masculine-coded, as expected. When **Example 2.B** is presented, Gender Decoder returns the advertisement as feminine coded due to the presence of the word "supportive." This demonstrates a shortcoming of the relatively simplistic word-matching method: the list of masculine-coded words may not be a comprehensive compilation of all potential masculine-coded words.	**Example 2.A** returns the label masculine-coded, as expected. The BERT-based model correctly predicts **Example 2.B** to be masculine-coded, which is likely due to BERT's use of word embeddings. Similar words are given similar word embeddings so BERT is able to use its more holistic understanding of language to recognize these close synonyms to masculine-coded words.