Gender Diction Classifier (GDC): the Biases and the Benefits

Didi Zhou
The University of Texas at Austin
didizhou@utexas.edu

Abhilash Potluri
The University of Texas at Austin
acpotluri@utexas.edu

Abstract

Natural Language Processing models are built to pick up on patterns in text that they are trained on and use those to make predictions about new data. This makes these models incredibly powerful and scalable to a wide range of tasks. However, this also makes them prone to reinforcing social biases that are found in the data they are trained on. Our work creates models to classify if sentences are gendered male, female, or neutral. Beyond this, we show the positive benefits that this model can have by detecting gender bias within contexts like children's books and toxic statements. These use cases can lead to more equal gender representation in media, prevent future toxicity, and is expandable to other kinds of text. On the other hand, we also analyze the features that the model uses to make predictions and discuss the gender stereotypes that this model and others like it learns. We conclude with a discussion on the limitations and ethical issues that can arise from such models.

1 Introduction

Social biases are incredibly prevalent in our society yet can be difficult to recognize. In particular, gender biases are ingrained within our culture and, subsequently, the text that we read and write. With the portrayal of women as submissive and dependent as well as the stereotype of men being violent and emotionless in our media and daily language, these biases continue to be perpetuated.

This work presents methods for automatically detecting such gender biases. These models classify sentences into 3 classes: female-gendered, male-gendered or neutral statements. We use a a number of neural methods for this: a deep averaging network (DAN), a long short-term memory (LSTM) based model, a convolutional neural network (CNN), and a BERT based transformer model. We train these models using two datasets from Wikipedia.

As neural models can be a black-box, it is important to interpret these models to the best we can to ensure better algorithmic accountability and to ensure that our model is learning actual gender biases rather than making concerning assumptions. We use integrated gradients and the leave-one-out method to explore what kind of features these models are weighting the highest and lowest. We explore some of the shortcomings of such models and considerations to take before using such algorithms.

Finally, we use these models to analyze gender biases within other contexts. First, we analyze children's books for biases within them. Since gender biases are ingrained within people from a young age, having them further perpetuated in children's books could shape how children express their own gender or how they view gender expression in others. This could help shape the ways authors present characters within their books or caution them to be more intentional to look into the ways their writing and their own biases could affect young children.

Next, we analyze toxic statements to determine how gender and different forms of toxicity are related. Much of the toxicity we see in the world has to do with people's identities. These can be very toxic and hurtful to those that identify with the groups being attacked. Knowing how different forms of toxicity intersect with gendered statements can help determine who is facing what kinds of toxicity and help detect for this toxicity.

2 Related Work

Our work builds off of previous work on social bias classifiers. In particular, we build off the data gathered in Dinan et al., 2020 to train our clas-

sifiers. Their work creates a framework for gender bias based on the speaker and the people being spoken about. Previous work has also been done on gender bias within specific contexts such as fiction writing (Fast et al., 2016), Wikipedia (Wagner et al., 2015), sports journalism (Fu et al., 2016), and in translations (Hovy et al., 2020). Hoyle et al., 2019 also look for gendered language through unsupervised methods.

Much of our work on the interpretability of our models builds off of existing research as we use the leave-one-out method (Li et al., 2017) and Facebook's Captum's (Kokhlikyan et al., 2019) integrated gradients method (Sundararajan et al., 2017).

Our analysis work is also based off previous work on gender biases and toxicity. Previous work shows the that underrepresentation of female characters and gender stereotyping exists in children's books (Gooden and Gooden, 2001; Filipović, 2018). However, this work focuses on content analysis methods does not use neural methods causing it to be slower and require more expertise in the field. Our work focuses on longer classic chapter books aimed for children. Hill et al. 2016 focuses on how well language models capture meaning in children's books and we get our data from their work.

Previous work on toxicity has been done on language models (Gehman et al., 2020) and labelling different categories of toxicity within text through Google Jigsaw's Perspective API[1]. We use data labelled for training Perspective in order to look at the intersection between toxicity and gender.

3 Gender Bias Classifiers

We built 4 different types of models in order to classify data at the sentence level into the three categories of male, neutral, and female. These models were a DAN (Iyyer et al., 2015), a LSTM RNN (Sak et al., 2014), a CNN (Kim, 2014), and a BERT-based transformer model (Devlin et al., 2019). These models were trained on two datasets, Funpedia (a less formal version of Wikipedia) (Miller et al., 2018) and Wizard of Wikipedia (knowledge-based conversations from Wikipedia) (Dinan et al., 2019). We used 34346 training examples, of which 10449 examples come from the Wizard data and the rest from Funpedia. We then had 3521 validation examples (to evaluate performance against) and 3521 test examples (for the final evaluation of model performance such that the model had never seen the sentences before) where 537 development examples and 470 test examples are from Wizard and the rest are from Funpedia. Combined, the datasets were approximately 50% male labelled sentences, 35% female labelled sentences, and 15% neutral sentences.

3.1 DAN

The first model we built was a Deep-Averaging Feed-Forward Network (DAN) with a word-based tokenizer. Using 300-dimensional GloVe word embeddings (Pennington et al., 2014), the tokenizer takes an input string and outputs a tensor of one-hot embedding indices which is fed into the DAN model. The model then passes the indices through a embedding layer and uses a series of linear layers with non-linearities (such as rectified linear units (ReLU)) embedded in between in order to process the word embeddings. Finally the results are log-softmaxed and the resulting 3 values provide the model's confidence of the sentence belonging to each of the classes. This model was trained for 50 epochs using the Adam optimizer with a learning rate of 0.001.

3.2 LSTM (RNN)

The next network we tried was a recurrent neural network built using a long short-term memory (LSTM) unit (Staudemeyer and Morris, 2019). The RNN model uses the same GloVe word embeddings as the DAN but in this case, it processes the word embeddings in the order of the sentence. After evaluating on each of the tokens in the sentence, it passes a hidden state to the computation of the next token, allowing for the model to maintain some notion of context when evaluating sentences. In our network, we made the hidden size of the LSTM output 100. After the final input token is passed into the LSTM, the output of that is passed into a linear layer. The results of the linear layer are log-softmaxed in order to get a result just like the DAN provided. This model was trained for 50 epochs using the Adam optimizer with a learning rate of 0.001.

3.3 CNN

The final baseline network we built was a convolutional neural network (CNN) which was built using multiple one-dimensional convolution layers and max pooling layers (Amin and Nadeem,

[1] https://www.perspectiveapi.com/#/home

2018). The convolutions in the network act as a sliding window looking for features over n word slices where n is the size of the kernel of the convolution. After max-pooling the results of the convolutions, the network now has feature maps for each sized kernel over the passed-in sentence. In our specific model, we concatenated the outputs from four convolutional layers of kernel sizes $2, 3, 4$, and 5 (each with their own max-pooling layers). We concatenate all these feature maps together and pass them through a single linear layer. These results are again log-softmaxed to get results consistent with the DAN and RNN. In our specific model, we concatenated the outputs from four convolutional layers of kernel sizes $2, 3, 4$, and 5 (each with their own max-pooling layers). This model was trained for 50 epochs using the Adam optimizer with a learning rate of 0.001.

3.4 Transformer-Based (BERT)

BERT (Bidirectional Encoder Representations from Transformers) is a pretrained model that is built off of transformers and the concept of attention layers (Devlin et al., 2019). This allows the model to remember context. BERT's training objective was on masked language modeling and next sentence prediction and we fine tune it to our sentence classification task. To implement this, we used HuggingFace's Distillbert model with the distilbert-base-cased tokenizer and the DistilBertForSequenceClassification model (Wolf et al., 2020). In order to do this, we first use the tokenizer to tokenize our sentences and obtain the input id and attention masks from it. Then we use the model to predict which class the sentence belongs to. We fine-tuned the pre-trained BERT model described above for our task by training it for 5 more epochs on our training set using the AdamW optimizer (Loshchilov and Hutter, 2017).

4 Classifier Results

Due to the fact that the datasets contained mostly gender-biased examples, during the training of all of the above models we implemented a weighted loss. The weights are computed based on the inverse of the frequency of each label in the train dataset which leads to the model being penalized more for predicting wrong on a neutral sentence than predicting wrong for one of the other classes. All the results below are due to this weighting which greatly improved the F1-score for the neutral class, regardless of the model type.

Metrics After training and tuning the 4 models, they were evaluated on the test set. For each model, we collected the accuracy, precision, and recall of each class which we then used to compute the corresponding F1 score. The accuracy of the model is simply the total number of correct predictions out of the total number of examples in the test set. The precision for a label is equal to the number of times the model correctly predicted that label out of the total number of times the model predicted that label. On the other hand, the recall for a label is equal to the number of times the model correctly predicted that label out of the total number of times the label was found in the test set. The F1 score can be calculated as the harmonic mean of the precision and recall:

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The compiled results can be found in Table 1.

Analysis of Results As we expected, given the recent success of large pretrained models, the BERT model outperformed all of our other models across almost all our metrics. Across all of our models and metrics, the models performed best on male-gendered sentences. This is probably because our training data had more male-gendered sentences within it than any other class. The female-gendered sentences also were represented more within our training data than the neutral sentences. Because neutral sentences can be focused on a much larger range of topics, this might have also made the model perform worse on it. Additionally, it was surprising how well the DAN performed given that the DAN collects the least amount of context when evaluating the sentence (simply averaging the embeddings does not tell you much about the context of the words). We attribute this to the class imbalance that led to the DAN mostly just predicting male class.

We also tried to see if these results were due to the use of pre-trained embeddings but when we tried training the baseline models for more epochs with embeddings from scratch, we found that the accuracy dropped by an average of 5% across all the model types and so we only reported the results of the pre-trained embeddings. We believe that the reason the embeddings trained from scratch did not generalize as well was because there was not enough data to properly capture the relations between words. This meant that when new words

Model	Accuracy	Precision			Recall			F1		
	All	M	N	F	M	N	F	M	N	F
DAN	2896 / 3521 = 0.822	0.839	0.519	0.698	**0.963**	0.109	0.397	0.896	0.181	0.506
LSTM	2873 / 3521 = 0.816	0.898	0.204	0.388	0.923	0.194	0.296	0.911	0.199	0.336
CNN	2727 / 3521 = 0.774	0.832	0.589	0.618	0.910	0.328	0.309	0.869	0.421	0.412
BERT	3078 / 3521 = **0.874**	**0.901**	**0.625**	**0.779**	0.951	**0.582**	**0.713**	**0.925**	**0.602**	**0.745**

Table 1: Table displaying the accuracy, precision, recall, and F1 scores for our 4 models across the three classes. (Bolded values indicate highest out of the rest of the models)

come from the test data the model did not know how to interpret those sentences and thus predicted wrong labels.

5 Interpretability

With a model that aims to classify gender in sentences, it is important to analyze what the model is classifying on. Recent work has shown gender biases within word embeddings (Bolukbasi et al., 2016), language models (Bordia and Bowman, 2019), and translations (Hovy et al., 2020). Because the bias in our word embeddings and the bias within BERT itself (Bhardwaj et al., 2020), it's no surprise that our model has picked up on some of the concerning social biases as well. In order to explore this, we included two different methods of interpretability for our BERT models: the leave-one-out method and the integrated gradients method.

5.1 Leave-One-Out Method

The leave-one-out method finds important words in the input by removing one input token at a time and seeing the drop in prediction confidence. The greater the drop is, the more important that word is to the model (Li et al., 2017).

Methodology First, loop through each sentence of our test examples and pass it into our model to get the predicted gender label with the highest probability. We save this score and then loop through the tokens of the sentence. One by one, we mask out each token by setting it as the "UNK" token. Then, we then use our model to get the prediction probability of for the same label we had originally predicted and subtract the original probability from it. This gives us a score of how important that word was for the model's original prediction. Finally, for each label, we take the top 100 words with the greatest change in probability when the word was masked out.

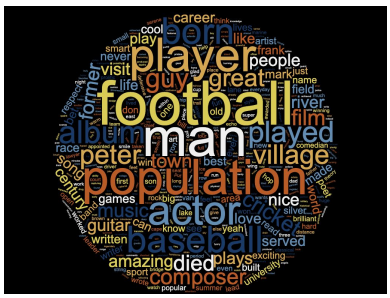
Results We hand-selected 20 of the top 100 words that created the greatest shift in probability and they can be found in Table 3. Additionally, we found the word that impacted the model's score the most in every sentence and combined them into word cloud in Table 2. From these examples we can see that the model correctly predicts overtly gendered words such as pronouns well. However, these results also show us that the model has associated words that are not innately gendered with certain genders based on societal bias. For example, the words "finance", "obliged", "governor", "vain", "convincingly" and "disagreed" help the model predict male. The words "romantically", "guest", "blonde", "emotionally", "happily", "socially", and "positively" help the model predict female. As expected, the words that help the model predict neutral are random and words that are not typically associated with any gender. Interestingly, the word "aground" is gendered female, possibly because of the way we refer to ships as female.

5.2 Integrated Gradients

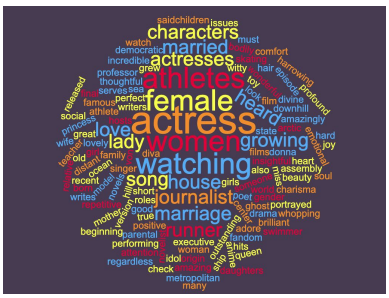
Integrated Gradients are a method of explainability proposed by Sundararajan et al. 2017 that takes the gradients along a path from a baseline (in this case, the "PAD" embedding vector) to the input vector. In doing this, we are able to attribute each the output probability of each output token to the input tokens. This allows us to see what input features our model is taking into consideration for its output probability.

Methodology To recreate this, we used Facebook's Captum (Kokhlikyan et al., 2019) to get the attributions of these gradients. Specifically, we used the Layer Integrated Gradients class which assigns an importance score to layer inputs or outputs by approximating the integral of gradients of the model's output with respect to the inputs.

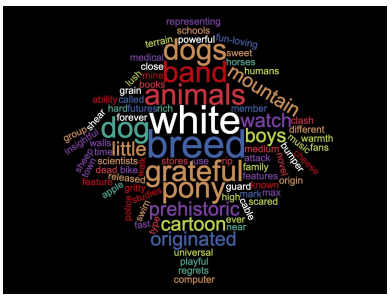
Results The results for five hand-selected sentences which our model predicts the wrong label



Male Words



Female Word



Neutral Words

Table 2: Words that created the biggest impact on the score with the sentences using the leave-one-out method. Larger words created the largest impact on scores within the most sentences.

MALE	he, him, his, father, prince, sir, bachelor, dude, finance, obliged, pledged, governor, randomly, convincingly, vain, optimistic, alongside, filling, soundly, disagreed
FEMALE	she, her, mom, wife, miss, maids, witch, teacher, queen, pregnant, romantically, guest, blonde, emotionally, happily, socially, incredibly, positively, aground
NEUTRAL	and, at, increases, factual, abundant, tropical, inland, automatically, usually, fair, ordained, low, faster, round, win, compass, along, before, variable, upgrade

Table 3: Displays a hand-selected set of 20 out of the 100 words that were most important to the model's prediction. These were found using the leave-one-out method of interpretability.

for are in Figure 1. Again, we see words that are not innately gendered being marked as important to the gender classification of sentences. Many of the words which the leave one out method found to be biased such as "romantic" help cause the model to predict the wrong resulting label.

6 Analysis of Other Data

We wanted to use our model in order to analyze societal biases within text. To do this, we focus on two different tasks: analyzing gender bias in children's books and exploring how toxic statements tend to be gendered.

6.1 Gender Bias in Children's Books

Previous work has shown that female characters are underrepresentation and gender stereotypes are perpetuated within children's books (Gooden and Gooden, 2001; Filipović, 2018) using content analysis methods. We wanted to use our model to analyze for gender bias using our neural method for faster analysis and without the need for expertise in the field. In order to do this, we use Hill et al. 2016's dataset of children's books scraped from Project Gutenberg[2]. Their work focuses on how well language models capture meaning in children's books. After getting this data, we feed it in sentence by sentence into our BERT model to classify each sentence for gender. Then we aggregate the number of sentences within each label in order to get our final results.

Results The results from this analysis are shown in Table 4. As we suspected, the results show that sentences associated with males are more highly represented in children's books. Though the percentage is smaller than the other books, it's surprising that a book like Alice's Adventures in Wonderland whose main character is female still has a higher percentage of male-gendered words than any other category. These results reinforce the idea that males are over-represented in media as compared to females. We hope that these results will be further evidence for authors to include more female voices outside of the mainstream ones in books, especially the books that are shaping the minds of young children.

6.2 Gender and Toxicity

Inspired by Gehman et al. 2020, we wanted to explore the relationship between gender and different types of toxicity. In order to do this we used 2000 comments from online forums such as Wikipedia and New York Times labeled for 16 different categories: toxicity, sever toxicity, identity attack, insult, profanity, sexually explicit,

[2]https://www.gutenberg.org/

Figure 1: Results from the integrated gradients method of interpretability for 5 hand-selected sentences

threat, flirtation, attack on author, attack on commenter, incoherent, inflammatory, likely to reject, obscene, spam, and unsubstantial (a description of each of these can be found in Table 5. We used the sample data labelled to train Google Jigsaw's Perspective API. Then, we ran these sentences through our BERT based gender classifier and labelled each sentence as either male, neutral, or female. After this, we took the average toxicity probabilities across sentences in each of the the three classes. This gave us the probability that a sentence falls into one of the Perspective categories, given that the sentence was male or neutral or female.

Results The results from this analysis can be found in Table 5. Our results show that sentences labeled female were only more probable to be obscene, rejected by New York Times's moderation, an attack on the author, and sexually explicit than male gendered sentences. Surprisingly, categories like flirtation and identity attack were more probable in male gendered sentences. One conjecture for why this might be is that Perspective's data set may not be representative of the whole internet and the authors of the comments may be disproportionately male or aimed towards males. Because there is no information on this, it is hard to analyze for.

7 Limitations and Ethical Considerations

Our work has many limitations that could lead to future work.

Sentence Level Classification Because we classify data by the sentence, our model ignores any continuous story-line or content across multiple sentences. In some cases, the gender implications of a sentence might change with different contexts.

Gender Bias As shown in Section 5, our model picks up on gender biases picked up from the data we used, the word embeddings we used, and the BERT model we used. Biases are perpetuated through our model as it often labels professions and adjectives society typically associate with males and females regardless of if they are actually referring to a male or female. Not only does this create worse results for our model, it reinforces these biases that are already ingrained into society. We hope that future work will be done on mitigating these biases within classification models like this as well as other tasks within natural

Percentage of Gendered Sentences per Book				
Book	**Male**	**Neutral**	**Female**	**Total Sentences**
The Yellow Fairy Book by Andrew Lang	0.752	0.06	0.187	5256
Alice's Adventures in Wonderland by Lewis Carroll	0.606	0.056	0.338	1638
Short Stories, 1902 to 1903 by Lucy Maud Montgomery	0.661	0.038	0.301	5493
The Jungle Book by Rudyard Kipling	0.878	0.087	0.036	3202
The Adventures of Old Mr.Toad by Thornton Waldo Burgess	0.947	0.022	0.031	1052

Table 4: Table displaying the percentage of sentences within each book that the model classified as either mostly `Male`, `Neutral`, or `Female`.

Average Toxicity Results per Category				
Category	**Description**	**Male**	**Neutral**	**Female**
Toxicity	A rude, disrespectful, or unreasonable comment that is likely to make people leave a discussion.	0.218	0.282	0.205
Severe Toxicity	A very hateful, aggressive, disrespectful comment or otherwise very likely to make a user leave a discussion or give up on sharing their perspective.	0.142	0.178	0.130
Identity Attack	Negative or hateful comments targeting someone because of their identity.	0.193	0.213	0.184
Insult	Insulting, inflammatory, or negative comment towards a person or a group of people.	0.207	0.249	0.179
Profanity	Swear words, curse words, or other obscene or profane language.	0.176	0.212	0.154
Sexually Explicit	Contains references to sexual acts, body parts, or other lewd content.	0.170	0.174	0.171
Threat	Describes an intention to inflict pain, injury, or violence against an individual or group.	0.254	0.270	0.246
Flirtation	Pickup lines, complimenting appearance, subtle sexual innuendos, etc.	0.343	0.346	0.335
Attack on Author	Attack on the author of an article or post.	0.264	0.256	0.308
Attack on Commenter	Attack on fellow commenter.	0.296	0.381	0.285
Incoherent	Difficult to understand, nonsensical.	0.501	0.530	0.443
Inflammatory	Intending to provoke or inflame.	0.229	0.269	0.240
Likely to Reject	Overall measure of the likelihood for the comment to be rejected according to the NYT's moderation.	0.581	0.623	0.614
Obscene	Obscene or vulgar language such as cursing.	0.194	0.190	0.214
Spam	Irrelevant and unsolicited commercial content.	0.337	0.262	0.241
Unsubstantial	Trivial or short comments	0.445	0.511	0.543

Table 5: Table displaying the average amount of each toxic label present in sentences that the model classified as either `Male`, `Neutral`, or `Female`.

language processing. One potential step could be to train models from scratch using debiased embeddings (such as the ones from Bolukbasi et al., 2016) and reinforce during training that the embeddings remain unbiased.

Gender Binary One of the biggest limitations of our model and many gender classification models is that it assumes a gender binary. We know that this is not representative of the real world and that gender exists on more of a spectrum. However, we were limited by our data from being able to generalize our model beyond the three classes we have.

This brings up an ethical concern that our work

and a lot of other NLP work fails to include minorities groups such as the LGBTQ+ community. We hope that future work will build on this work and create technologies that include everyone.

8 Conclusion

We create gender bias classifiers based on DAN, LSTM, CNN, and BERT that categorizes sentences into categories of male, female, and neutral. We find that the BERT model out-performs the rest of the models. We then ask whether or not our model has learned societal gender biases and use two interpretability methods to find that our model has indeed picked up on these biases. Finally, we look for relations between gender biases and children's books as well as toxic comments. We find that children's books disproportionately use male-gendered language and that toxic statements tend to also be male-gendered.

Acknowledgments

We want to thank Dr. Greg Durrett for guiding us through this project as well as teaching us what we know about NLP. This project wouldn't be possible without you!

References

Muhammad Zain Amin and Noman Nadeem. 2018. Convolutional neural network: Text classification model for open domain question answering system. *CoRR*, abs/1809.02479.

Rishabh Bhardwaj, Navonil Majumder, and Soujanya Poria. 2020. Investigating gender bias in bert.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and A. Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *ArXiv*, abs/1607.06520.

Shikha Bordia and Samuel R. Bowman. 2019. Identifying and reducing gender bias in word-level language models. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 7–15, Minneapolis, Minnesota. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Emily Dinan, Angela Fan, Ledell Wu, Jason Weston, Douwe Kiela, and Adina Williams. 2020. Multi-dimensional gender bias classification.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents.

Ethan Fast, Tina Vachovsky, and Michael Bernstein. 2016. Shirtless and dangerous: Quantifying linguistic signals of gender bias in an online fiction writing community.

K. Filipović. 2018. Gender representation in children's books: Case of an early childhood setting. *Journal of Research in Childhood Education*, 32:310 – 325.

Liye Fu, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. 2016. Tie-breaker: Using language models to quantify gender bias in sports journalism. *ArXiv*, abs/1607.03895.

Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. In *EMNLP*.

Angela M. Gooden and M. Gooden. 2001. Gender representation in notable children's picture books: 1995–1999. *Sex Roles*, 45:89–101.

Felix Hill, Antoine Bordes, S. Chopra, and J. Weston. 2016. The goldilocks principle: Reading children's books with explicit memory representations. *CoRR*, abs/1511.02301.

Dirk Hovy, Federico Bianchi, and Tommaso Fornaciari. 2020. "you sound just like your father" commercial machine translation systems include stylistic biases. In *ACL*.

Alexander Miserlis Hoyle, Lawrence Wolf-Sonkin, H. Wallach, Isabelle Augenstein, and Ryan Cotterell. 2019. Unsupervised discovery of gendered language through latent-variable modeling. In *ACL*.

Mohit Iyyer, V. Manjunatha, Jordan L. Boyd-Graber, and Hal Daumé. 2015. Deep unordered composition rivals syntactic methods for text classification. In *ACL*.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP*.

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Jonathan Reynolds, Alexander Melnikov, Natalia Lunova, and Orion Reblitz-Richardson. 2019. Pytorch captum. `https://github.com/pytorch/captum`.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2017. Understanding neural networks through representation erasure.

Ilya Loshchilov and Frank Hutter. 2017. Fixing weight decay regularization in adam. *CoRR*, abs/1711.05101.

Alexander H. Miller, Will Feng, Adam Fisch, Jiasen Lu, Dhruv Batra, Antoine Bordes, Devi Parikh, and Jason Weston. 2018. Parlai: A dialog research software platform.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

H. Sak, A. Senior, and F. Beaufays. 2014. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *INTERSPEECH*.

Ralf C. Staudemeyer and Eric Rothstein Morris. 2019. Understanding LSTM - a tutorial into long short-term memory recurrent neural networks. *CoRR*, abs/1909.09586.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks.

Claudia Wagner, David Garcia, Mohsen Jadidi, and Markus Strohmaier. 2015. It's a man's wikipedia? assessing gender inequality in an online encyclopedia.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.