Gender Bias In Pronoun Resolution

Frederik Warburg, warburg@berkeley.edu, 3034485572
Isak Kyrre Lichtwarck Bjugn, isak@berkeley.edu, 3034433796
Rune Langergaard, runel@berkeley.edu, 3034493801

August 28, 2019

Abstract

Natural language understanding relies on being able to recognize expressions that refer to the same entities or concepts. This is the task of coreference resolution. We investigate neural models for resolving pronouns, and discover that many state-of-the-art models tend to perform better for male pronouns than female. Furthermore, we find that one commonly used dataset for training such models is highly biased, suggesting bias is transferred from training data to models. This finding is alarming, because biased models can lead to gender discrimination when involved in decision-making, e.g. for job applicants. We are motivated to highlight the problem and explore methods for mitigating such bias. We explore debiased word embeddings, and show through experimental work how replacing gendered pronouns with gender-neutral ones can reduce bias. Both techniques are tested on state-of-the-art neural models, where they show the ability to lower bias. We believe these approaches can reduce the influence of gender stereotypes in decision-making.

Abbreviations—CR = coreference resolution; E2E = end-to-end; GAP = gendered ambiguous pronouns; ML = machine learning; NLP = natural language processing; NN = neural network; RNN = recurrent neural network

1 Introduction

Coreference resolution (CR) concerns determining which expressions refer to the same entities, and is a key element of natural language processing (NLP) systems for text comprehension. Pronoun resolution concerns resolving coreferent nouns and pronouns. Considerable advances have been made in the field CR systems in recent years, particularly by later neural models. Despite this development, models are still disposed to be influenced by and relying on societal stereotypes that are present in the data on which they are trained. Specifically, gender bias poses a problem, where models tend to perform better for male pronouns than for female pronouns. An example is models linking pronouns to the wrong referring entity, based on stereotypes of male and female occupations (Zhao et al., 2018; Caliskan et al., 2017). The bias poses a challenge for the models that carry it, and Webster et al. (2018) also found baseline models to struggle on the gender-neutral Gendered Ambiguous Pronouns (GAP) dataset. Thus, there still lies a challenge in understanding and mitigating this bias, such that the models perform better on tasks of resolving ambiguous pronouns.

Our motivation lies in reducing the risk of amplifying existing bias in data and preventing discrimination in model predictions. This is done by highlighting where the data used for training coreference models show gender bias. Furthermore, we compare performance between baseline models, such as the state-of-the-art E2E model presented by Lee et al. (2017) with a modified version that is specified for the given task presented by Webster et al. (2018)[1]. We also study the effect of using different debiasing methods such as data augmentation and using debiased word embeddings. Data augmentation can help by equalizing the number of male and female examples the model is trained on, to prevent teaching the model societal stereotypes. Another way of mitigating this is to substitute gender-neutral pronouns such as "ne/nem/nir/nirs/nemself" for gendered pronouns, so that the model is not trained to connect specific mentions to specific genders.

To evaluate the models we will look at both the general model performance and how successful they are in overcoming gender bias. The gender bias is studied by comparing performance across genders. As done in (Webster et al., 2018) a comparison is made between the F1 scores for male and female, to explore how well the model fares for each of the individual genders. All our code and experiments are available on our Github[2].

2 Related Work

Coreference resolution is an important part of text understanding, and is employed in NLP areas such as machine translation and information extraction (Clark, 2015). CR is broadly distinguished into two groups: methods that are heavily based on linguistic knowledge in the area of the task, methods based on machine learning (ML), that are more data-driven and therefore dependent on the data-quality (Elango, 2005). Biased predictions arise in ML-driven methods due to biased data. Our focus is specifically on the gender bias in ML models.

Linguistic-based work in CR dates back to the seventies, with major advances made by Hobbs (1978), using syntactic constraints, and Elango (2005). ML models include naive bayes, decision trees, conditional random fields, clustering and neural network (NN) models, and have gained much attention in recent years. NN methods will be the focus of this paper, and the recent end-to-

[1] `https://kaggle.com/c/gendered-pronoun-resolution`

[2] Code: `https://frederikwarburg.github.io/NeuralCoref/`

end (E2E) neural model by Lee et al. (2017) will be explained in great detail and later implemented on the GAP data (Webster et al., 2018), as well as a modification of this model targeted to the GAP challenge (Webster et al., 2018). By learning dense vector representations for mention pairs, NN models can improve the generalizibility of CR systems to new tasks and data (Clark, 2015).

2.1 Gender bias in machine learning

Several difficulties have arisen with ML methods used for CR modeling. Gender bias arises when training data for the models is highly skewed towards one of the genders. Kiritchenko and Mohammad (2018) test 219 sentiment systems using their `Equity Evaluation Corpus (EEC)` benchmark dataset to see whether they are biased towards certain races or genders. They find that 75% of these systems have significant bias towards certain races or genders (Kiritchenko and Mohammad, 2018). This is backed by Rudinger et al. (2018), who also find a systematic gender bias in currently available coreference systems.

The widely used OntoNotes corpus, which is often used for training coreference models, also carries a large bias into the ML models. This corpus has over 2,000 gendered pronouns, where less than 25% of these are feminine (Webster et al., 2018). Webster et al. find similar gender biases in several widely used training datasets.

Zhao et al. (2018) present a new benchmark for the CR models focused on the gender bias, introducing the WinoBias corpus to identify whether a system exhibits gender bias. A system is characterized as biased if it links a gender pronoun wrongly to the occupation that is dominated by that gender rather than correct occupation-antecedent pair. Zhao et al. furthermore suggest a method for altering the training data to overcome bias, by creating an auxiliary dataset where all gendered entities are replaced by its counterpart of the other gender. This is referred to as data augmentation. The algorithm is trained on the union of the original and auxiliary data, and thus learns equal probability of linking an occupation to either gender. This is found to help mitigating bias, without worsening overall model performance significantly (Zhao et al., 2018). Word embeddings also tend to be biased: E.g. the word *programmer* is closer to the word *man* than it is to *woman*. Here they suggest replacing the normal word GloVe word embedding with other debiased ones (Bolukbasi et al., 2016).

2.2 Coreference resolution performance metrics

In addition to efficient discrimination between good and bad systems, Luo (2005) states that a performance metric should also be fairly easy to interpret. The F-measure is a commonly used metric in the Message Understanding Conferences (MUC). Though popular, one caveat of the F-measure is that it ignores single-mention entities, and favors systems producing fewer entities with potentially lower performance. To overcome these faults B^3 was later introduced, whose final performance metric is a weighted sum of precision and recall for each individual mention (Bagga and Baldwin, 1998).

Luo (2005) introduces a measure called the Constrained Entity-Alignment F-measure (CEAF). System entities and reference entities are aligned by maximizing the total entity similarity. Then you can calculate precision, recall and F-metric from the similarity with a constraint. A system that outputs too many entities will be penalized with a bad precision and system that outputs too few will have a bad recall score. A perfect system gets an F-value of 1. The score that outputs no correct entities will have a score of 0. Therefore the metric reflects the percentage of mentions that are in the correct entity (Luo, 2005). They define a similarity measure $\phi(R, S)$ that takes a non-negative value. If $\phi(R, S)$ is equal to zero then it means that R and S have nothing in common.

In evaluating our models we will report precision, recall and F1 scores for our models, together with these same scores for each individual gender. This enables a way of quantifying the model performance in general, as well as the performance on each of the genders, to quantify the gender bias. We will measure bias as done by Webster et al. (2018), by dividing the male F1 score with the emale F1 score. An completely unbiased system will therefore have a ratio value of 1. The F1 score will be used as the main performance metric for this given task, as we are dealing with a special case of the coreference resolution problem. In this particular case we have a tertiary classification problem between antecedent A, B or neither. The B^3 metric therefore will not apply as it uses all chains found in each document when being computed. We calculate a single gender bias score B as the ratio between the F1 score for female and male predictions, respectively.

3 Theory

3.1 E2E coreference resolution model

The Lee et al. end-to-end (E2E) coreference resolution model is a state-of-the-art neural system, that does not make use of syntactic parsers. Syntactic parsing is the task of recognizing a sentence and assigning a syntactic structure to it. The model consists of three main parts, which are document encoding, mention scores and pairwise scores. Input into the model is a document D containing T words along with metadata. There will be $N = \frac{T(T+1)}{2}$ spans in D. The task at hand is to assign each span an antecedent y_i. The goal is to find a conditional probability of all the possible antecedents, given the document D, such that you can maximize the probability and find the most likely antecedent $P(y_1, y_2, ..., y_N \mid D)$.

The ***document encoding*** will first make use of three word embeddings (GloVe, Turian and Character embeddings) that are concatenated together. It will then be shaped/-packed into fitting a LSTM RNN. Dropout will be applied on the concatenated word embeddings both before and after it is run through the LSTM. The embeddings are then unpacked/unpadded.

For finding the ***mention scores*** the spans are found and, together with the output-states from a long short-term memory (LSTM) recurrent neural network (RNN) in the document encoding, the attention weights are computed. These weights are scaled using a softmax function to be between zero and one. The most low-scoring spans will be pruned, to make the task more feasible, such that spans that have no mentions are pruned away. The attention weights are then used to calculate the attention-weighted embeddings. These are in turn used to calculate the span-mention scores. In each span there are two important feature groups, that is the surrounding context of the span, and the internal structure of the span.

The span-mention scores are then used to calculate the ***pairwise scores***. This is done by using the embedded features and the mention scores. For each mention a pairwise score is computed with its possible antecedents. The score will be calculated by summing the two spans' individual mention scores and their pairwise antecedent scores. The model will fix a dummy antecedent to have the score $\varepsilon = 0$, and assign an antecedent to a span if the score rises above the dummy antecedent and assign no antecedent if all possible antecedents have a negative score. The scores are calculated using a feedforward neural network (FFNN). A softmax function is then used to get a probability score for each span and its antecedents, to find the most likely antecedent. These are used to cluster the document to the entities in the text and their mentions[3].

Even though this state-of-the-art model often performs very well, it can potentially be very biased. The bias can be introduced through the commonly used OntoNotes training data and also through the word embeddings used in the model [4]. This corresponds with the findings in Webster et al. (2018). By specializing the model with respect to the GAP dataset, its performance might be improved.

3.2 RNN attention

This model is a simplification of the E2E neural coreference model presented by Lee et al. (2017). This model can be simplified to fit the challenge presented in Webster et al. (2018). Here each target pronoun will only have two possible antecedents that it can be linked to, which are determined beforehand. Therefore, both the pronoun mention and possible antecedents are already defined. This modifies the coreference task into a problem of determining which of the two antecedents is the correct one, or possibly neither. This model extracts the given sentence of the target pronoun and the surrounding 100 words (50 on each side). From here, the features are created together with the word embeddings and run through an RNN. The RNN will compute attention weights, and each mention score and pairwise score is calculated. A softmax function is used to find the highest probability between the two antecedents and neither of the two. The ability to know which are the two possible antecedents creates a more focused task, and this is expected to give better performance in the GAP challenge (Webster et al., 2018). The architecture of this model is seen in Appendix G.

Both models start by concatenating the created embeddings and passing them through an RNN (LSTM). Afterwards, the attention weights are used to focus on the important features. Both models then output a score for each mention pair, where the RNN attention model only focuses on the two entities in focus (A, B or neither).

4 Data

The purpose of this section is to introduce the GAP dataset used in our analysis and compare it in terms of gender bias with the OntoNotes dataset. OntoNotes is the largest coreference corpus and it is widely used for training both word embeddings and coreference models (Webster et al., 2018).

4.1 GAP data

GAP is a human-labeled dataset with 8,908 observations of ambiguous pronoun-name pairs derived from Wikipedia (Webster et al., 2018). This dataset is created to challenge biased systems and reward gender-fair ones. The data was provided for the Kaggle competition[5], where the dataset was organized in eight columns of text and metadata according to Table 5.

Previously constructed datasets have been of limited size and consisted of manually constructed examples, which generally differ from examples CR systems will meet in real applications. The examples posed in GAP contains two named entities of the same gender and a pronoun that may refer to either (or neither) as seen in Table 1. GAP will in this way complement the widely used OntoNotes data with naturally occurring ambiguous pronouns.

It is important to note that the GAP dataset is constructed in such a way that there are equally many male and female target pronouns, whereas the OntoNotes dataset is skewed in favor of male pronouns. Specifically, of the gendered pronouns in the OntoNotes training data, 83.7% are male (9321 male, 1821 female). In Figure 1, we compare the

[3] The model architecture is drawn in Appendix F

[4] See more about bias in word embeddings under Section 5.2

[5] `https://kaggle.com/c/gendered-pronoun-resolution`

GAP and OntoNotes dataset in terms of co-occurence between gendered pronouns and different entity types.

Document	Pronoun	Entity
Englishman Sir Thomas Herbert, in the 1638 edition of ***his*** travel book [...]	his	A
He believes Archie's relationship with Ronnie suffered when he realised that he could not control ***her***	her	B
Zara tells Husna about Nighat and ***her*** father	her	Neither

Table 1: Three examples from the GAP dataset. The dataset contains short text documents. For each document there is a highlighted pronoun that has to be linked with an entity. In the Figure, we show examples where the pronoun is linked together with entity A, B or none of them.

5 Mitigating Gender Bias

5.1 Pronoun gender neutralization

As Figure 1 illustrates, neither GAP is void of bias in terms of mentions of male and female pronouns, even though the pronouns which are subject to the coreference resolution task are evenly distributed. To mitigate this, we employ data augmentization - the alteration of textual content in order to remove bias. Zhao et al. (2018) employed a scheme were training data were duplicated and all gender-implicating references were swapped for its counterpart of the opposite gender, to create an auxiliary dataset. Because this encompasses not only pronouns, but common nouns and proper nouns, the augmentation by Zhao et al. (2018) relied on annotation by crowdsourcing through Amazon Mechanical Turk. Such measures are beyond the constraints of this project, and so we decided to augment the data by replacing gendered pronouns in the GAP dataset with gender-neutral ones, instead of removing all notions of gender as suggested by nouns and proper nouns, as well as pronouns.

Some relevant sets of gender neutral pronouns are described in Table 2. These can be subdivided into pronoun schemes which have precedence in the English language as neutral, singular pronouns, or gender-neutral pronouns which are completely engineered. *It* and *One* both have linguistic precedence, and while *They* can be used in a singular sense, this normally would require conjugation of all related verbs to plural form. *Who* allows for similar pronoun cases as *He/She*, but does otherwise not convey the same meaning (Remark: *Themself* less accepted than *Themselves* as reflexive form of *They*, but is well suited for our purposes). *Spivak* contains examples of engineered pronouns, found in some literature where authors prefer gender-neutral pronouns. Here, it is reproduced in Elverson's variant (Robinson, 2018). *Ne* is another engineered pronoun scheme often appearing in texts criticizing heteronormative grammar (Darr and Kibbey, 2016).

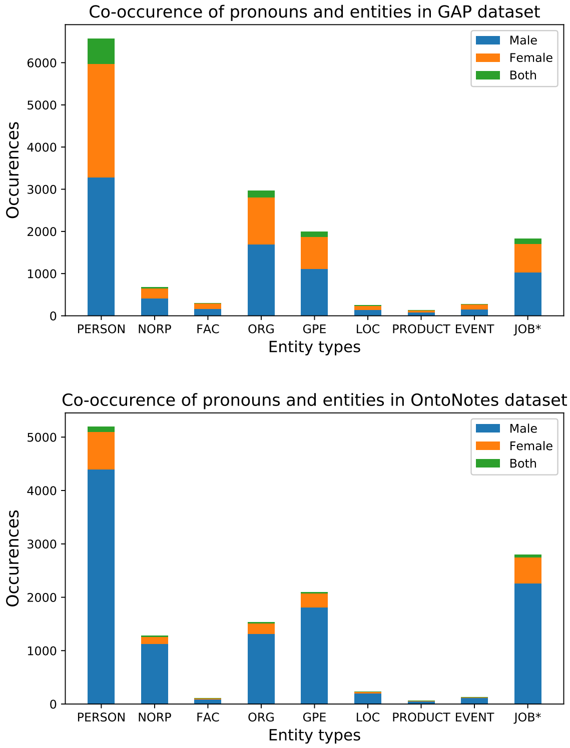


Figure 1: Distribution of co-occurences between gendered pronouns and different entity types across all single sentences. *Top:* GAP dataset, *bottom:* OntoNotes dataset. Explanation of entity types can be found in Table 6. Remark: JOB* is not a spaCy entity type, but refers to a list of job titles compiled by Berry (2018).

5.2 Bias in word embeddings

We also study the bias in word embeddings by studying the distance between the words `man` and `woman` to different job titles. From here we can see if `man`/`woman` is closer to their respective stereotypical jobs, and may assess whether a "debiased" embedding indeed is unbiased. Here we study five different word embeddings: GloVe (Pennington et al., 2014), GoogleNews Word2Vec (Kusner et al., 2015), Word Dependency (Levy and Goldberg, 2014), Turian (Turian et al., 2010) and Google Word2Vec debiased (Bolukbasi et al., 2016) embeddings . We find that GloVe and Word2Vec embeddings show biases with respect to select job titles. Turian and the Word Dependency embeddings show some pull towards either of the genders, but no systematic bias in the stereotypical job titles. The debiased Word2Vec embeddings from Bolukbasi et al. (2016) show virtually no bias, with `man` and `woman` having the same distance to all stereotypical job titles[6].

[6] The results are visualized in appendix Appendix C

Scheme	He/She	Him/Her	His/Her	His/Hers	Himself/Herself
It	It	It	Its	Its	Itself
One	One	One	One’s	One’s	Oneself
They (singular)	They	Them	Their	Theirs	Themself
Who	Who	Whom	Whose	Whose	Themself
Spivak (Elverson)	Ey	Em	Eir	Eirs	Emself
Ne	Ne	Nem	Nir	Nirs	Nemself

Table 2: Different schemes for neutralizing gendered pronouns. The top four schemes have precedence in the English language, while the bottom two are engineered pronouns to remove implicated gender in language.

In our experiments, we compared the bias in several commonly used word embeddings. As seen in Figure 2, the Dependency Word Embedding (Levy and Goldberg, 2014) and the debiased Google News Embedding (Bolukbasi et al., 2016) have very low bias, whereas the Common Crawl GloVe 840, Turian, and Google News embedding have rather high bias. We evaluated the bias based on a similarity measure between `man`/`woman` and common male/female jobs. In a gender-unbiased word embedding the distance between a job and `man`/`woman` should be equal. We tested five commonly used distance measures[7].

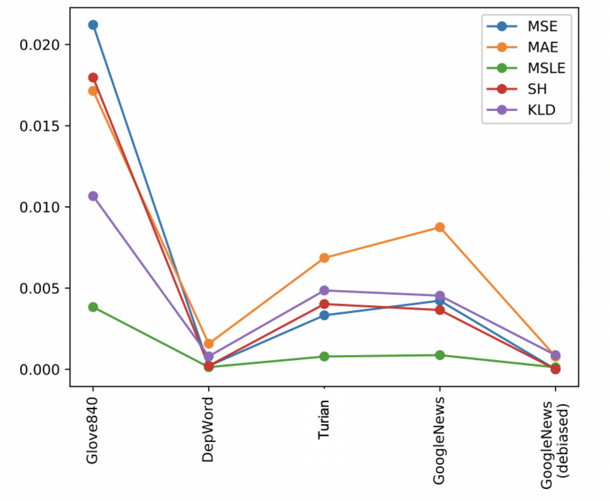


Figure 2: Average absolute distances between `man`/`woman` and various job titles. Lower means less bias.

6 Experiments

In this section, we describe the experiments we carried out and the motivation behind these.

6.1 Majority vote

We choose the simplest possible baseline model, where we simply make a majority guess. As the GAP dataset is not gender biased when considering only the target pronouns, our majority should not be biased either, performing equally good (or bad) for both genders.

6.2 E2E Coreference Model

We use the pre-trained E2E Coref Model explained in Section 3.1 to make predictions. More specifically, we use the AllenNLP implementation (Gardner et al., 2017). This is among the highest-performing models for most coreference tasks. In particular, Webster et al. (2018) report that the E2E coref model was the best performing out-of-the-box model on the GAP dataset. This model is pre-trained on the OntoNotes dataset, so we expect a certain amount of gender bias in the predictions.

6.3 RNN Attention Model

As the GAP dataset constitutes a simplified, tertiary coreference problem, we can drastically simplify the E2E model as explained earlier. We train the RNN Attention model on the Kaggle data. All experiments with this model are trained with early stopping with 5 epoch delay, sparse catagorial cross entropy loss, batch size 30, and the Adam optimizer. We also tried varying the loss-function to improve performance of the model. The results of tuning the loss function can be found in Appendix D.

6.4 Debiased Embedding

As GAP has equally many male/female target pronouns, the model bias is most probably introduced in the pretrained embedding. We found in Figure 2 that the Common Crawl GloVe Embedding (which is default in spaCy) has high gender bias. We train the RNN Attention Model with two unbiased embeddings, namely the Dependency-Based Word Embedding (Levy and Goldberg, 2014) and the Debiased Google News Embedding (Bolukbasi et al., 2016).

6.5 GAP data neutralization

We perform data augmentation by replacing all gendered pronouns with gender-neutral pronouns, either precedented pronouns or engineered ones, according to Table 2.

[7] Similarity measures used: MSE = Mean Square Error, MAE = Mean Absolute Error, MSLE = Mean Square Logarithic Error, SH = Squared Hinge, KLD = (scaled) Kullback Leibler Divergence.

Model	M precision	F precision	M recall	F recall	M F1	F F1	Bias
Majority vote	0.15	0.15	0.33	0.33	0.20	0.20	1
E2E Coreference Model	0.49	0.48	0.57	0.53	0.47	0.40	0.85
RNN Attention Model	0.63	0.60	0.60	0.57	0.60	0.58	0.96
GoogleNews debiased Embedding	0.60	0.60	0.56	0.58	0.57	0.59	1.03
Word Dependency Embedding	0.62	0.64	0.61	0.65	0.61	0.64	1.05
It pronoun scheme	0.63	0.64	0.58	0.60	0.59	0.60	1.02
Spivak pronoun scheme	0.64	0.62	0.58	0.61	0.60	0.61	1.02

Table 3: Gender bias: M = Male, F = Female, Bias = (female F1) / (male F1). Remark: For the gender-neutral pronoun schemes single trials values are reported, from the best performing models with respect to median bias, according to Table 8.

When employing data augmentation, we have freedom in which datasets we want to alter, as optimal performance is not necessarily achieved by treating training, validation and test data equally. We tested all combinations of pronoun schemes for the training data and for the test data (we'll refer to this as "crossing"). In total this amounts to 49 trails of predictions based on data augmentation. Validation data is treated the same way as training data. Single trials of the two best performing pronoun schemes with respect to median bias acquired from bootstraping are reported in Table 3. Performance from all schemes are reported in Table 8 (only trials where schemes are crossed with themselves or the original dataset are reported).
Interestingly, pronouns from both the *Spivak* and *Ne* schemes had existing word embeddings in the spaCy vocabulary. This justifies the fact that they had different performance, as the embeddings are differen, and might even exhibit some bias as well. For a set of pronouns without existing word embeddings, this pronoun-based bias would not come into play, but this would come at the cost of deteriorated performance owing to a generally harder coreference task.

Model	F1	logloss
Majority vote	0.20	18.15
E2E Coreference Model	0.43	17.01
RNN Attention Model	0.59	10.42
GoogleNews debiased Embed	0.57	11.20
Word Dependency Embed	**0.63**	**9.67**
It pronoun scheme	0.575	10.074
Spivak pronoun scheme	0.573	10.557

Table 4: Performance

7 Results & Discussion

In this section, we analyse the performed experiments. The pre-trained coreference systems' predictions are converted into the format of the Kaggle data by investigating whether the A, B or Neither of the coreference are in the same cluster as the pronoun. In Table 3 we report bias metrics for each experiments. In Table 4 we report the performance of each model. We use both the binary F1 score averaged over the predictions as well as the multiclass logorithmic loss function

$$\text{logloss} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M} y_{ij}\log\left(p_{ij}\right) \tag{1}$$

to compare the performance of the models[8]. From Table 4 we see that using the RNN Attention model with the Word Dependency embedding have the best performance according to both the F1 score and the logloss.

In Table 3, we compare the bias of the model predictions. We calculate the performance for both male and female pronouns. We expect an unbiased model to be equally good at linking males and female pronouns to their entities. Thus, we expect `Bias = Female F1 / Male F1 = 1` for an unbiased model. We see that training on the unbiased dataset removes most of the bias observed in the E2E Corefence model. We further show that we are able to mitigate gender bias by either replacing the GloVe word embedding with a gender-unbiased embedding such as the GoogleNews debiased Embedding, or using a gender-neutral set of pronouns. Through bootstrapping we found that we could not conclude that either of the five models were gender biased, and we found using a t-test that they did not have significantly different performance.

8 Conclusion

In this project, we studied gender bias in the task of pronoun resolution. We compared the gender bias for two datasets: OntoNotes and the Gendered Ambiguous Pronouns dataset (GAP). In the OntoNotes dataset, we found that less than 20% pronouns were female pronouns, and that when a person, organization or job occured together with a gendered pronoun in a sentence, 85% of them would be male pronouns. GAP on the other hand contains equally many examples of male and female ambiguous pronouns. We do find similar bias in terms

[8] We choose also to report the logloss as this is the official evaluation metric in the Kaggle competition.

of co-occurence between persons, organizations and entities and gendered pronouns, but to a much lesser degree. Through experimental work, we found that the state-of-the-art E2E coreference neural model (Lee et al., 2017) trained on the biased OntoNotes dataset made biased predictions. We investigated several methods to mitigate this bias.

One method to mitigate bias is to train on an unbiased dataset. We used a variation of the Lee et al. model, called the RNN attention model, that is targeted specifically towards the GAP challenge. We find that this model both improves performance and reduces gender bias. Another method to mitigate bias is to replace pronouns with a gender-neutral pronoun scheme. We compared seven different schemes and tested how they mitigated the problem of the model learning gendered stereotypical matters. We found that the *It* pronoun scheme, when applied to training and validation data and tested on un-altered data, resulted in the lowest bias. A third method to mitigate gender bias is to use an unbiased word embeddings. We found that several commonly used word embeddings have an underlying gender bias. The pronoun-resolution models rely on the word embeddings and it can therefore help to mitigate the bias when using unbiased word embeddings. We find that the debiased word embeddings from (Bolukbasi et al., 2016) or the Word Dependency embedding (Levy and Goldberg, 2014) show the lowest bias.

Thus, we find that training on an unbiased dataset, using an unbiased word embedding or using a gender-neutral pronoun scheme all reduce gender bias in model predictions. We find that the two first can be applied without performance drops. We urge practitioners and researchers to consider and test if their NLP systems are gender biased, and hope that the suggested methods will help mitigate these biases.

References

Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In In The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference, pages 563–566, 1998.

Dina Berry. Samples for the language understanding intelligent service (LUIS). https://github.com/Microsoft/LUIS-Samples, 2018.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. CoRR, abs/1607.06520, 2016. URL http://arxiv.org/abs/1607.06520.

Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. Science, 356(6334):183–186, 2017. ISSN 0036-8075. doi: 10.1126/science.aal4230. URL http://science.sciencemag.org/content/356/6334/183.

Kevin Clark. Neural coreference resolution, 2015.

Brandon Darr and Tyler Kibbey. Pronouns and thoughts on neutrality: Gender concerns in modern grammar. Pursuit - The Journal of Undergraduate Research at the University of Tennessee, 7(1), 2016.

Pradheep Elango. Coreference resolution: A survey. University of Wisconsin, Madison, WI, 2005.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. Allennlp: A deep semantic natural language processing platform. 2017.

Jerry R Hobbs. Resolving pronoun references. Lingua, 44 (4):311–338, 1978.

Svetlana Kiritchenko and Saif M. Mohammad. Examining gender and race bias in two hundred sentiment analysis systems. CoRR, abs/1805.04508, 2018. URL http://arxiv.org/abs/1805.04508.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In International Conference on Machine Learning, pages 957–966, 2015.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. arXiv preprint arXiv:1707.07045, 2017.

Omer Levy and Yoav Goldberg. Dependency-based word embeddings. In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), volume 2, pages 302–308, 2014.

Xiaoqiang Luo. On coreference resolution performance metrics. In Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT '05, pages 25–32, Stroudsburg, PA, USA, 2005. Association for Computational Linguistics. doi: 10.3115/1220575.1220579. URL https://doi.org/10.3115/1220575.1220579.

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), pages 1532–1543, 2014.

Brodie Robinson. From heo to zir: A history of gender expression in the english language. Senior Honors Theses, 750(4):24–26, 2018.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. Gender bias in coreference resolution. arXiv preprint arXiv:1804.09301, 2018.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. Word representations: A simple and general method for semi-supervised learning. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, ACL '10, pages 384–394, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1858681.1858721.

Kellie Webster, Marta Recasens, Vera Axelrod, and Jason Baldridge. Mind the gap: A balanced corpus of gendered ambiguous pronouns. In Transactions of the ACL, page to appear, 2018.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Gender bias in coreference resolution: Evaluation and debiasing methods. arXiv preprint arXiv:1804.06876, 2018.

A GAP dataset

Attribute	Description
Text	Text paragraph in question, with multiple pronoun antecedent choices.
Pronoun	Target for which we find an antecedent.
Pronoun-offset	Character offset of the pronoun, relative to the start of the paragraph.
A	The first name candidate in the text. Algorithm chooses between candidate A and B.
A-offset	Character offset of name candidate A.
B	Second name candidate pronoun in the text.
B-offset	Character offset of candidate B.
URL	Wikipedia page whence example originates.

Table 5: Annotation data in the GAP dataset.

B spaCy entity types

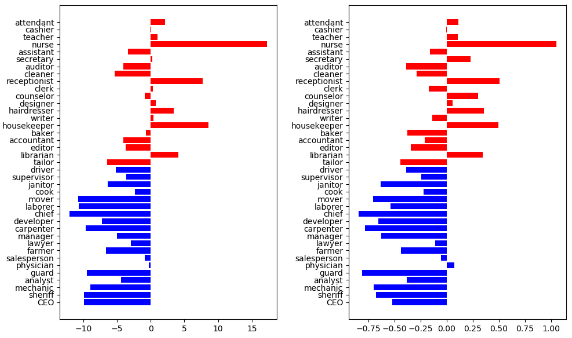
Type	Description
PERSON	People, including fictional.
NORP	Nationalities or religious or political groups.
FAC	Buildings, airports, highways, bridges, etc.
ORG	Companies, agencies, institutions, etc.
GPE	Contries, cities, states.
LOC	Non-GPE locations, mountain ranges, bodies of water.
PRODUCT	Object, vehicles, foods, etc. (Not services.)
EVENT	Named hurricanes, battles, wars, sports events, etc.

Table 6: Entity types recognized by spaCy.

C Bias in word embeddings

We plot each embedding distance with two distance measures that is L^1-norm, L^2-norm. The difference is measured as $d_i = ||job_i - man|| - ||job_i - woman||$ such that a negative distance means that it has the smallest distance towards males and a positive distance means a smaller distance towards females. The color bars indicates whether the job is a typical female (red) or male job (blue). In a gender unbiased embedding we expect the distance to the words `man` and `woman` are independent on the job being a typical man or female job.

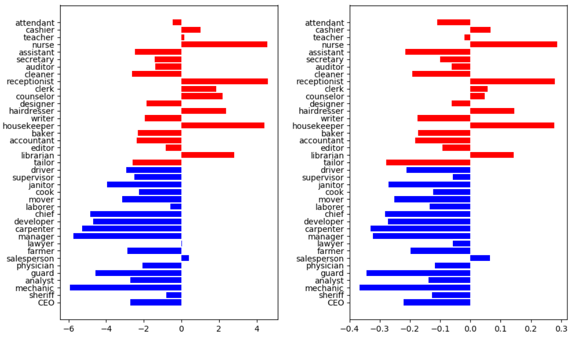
From Figure 3f - Figure 6 we find that the commonly used GloVe and Google News embeddings are biased as they are consistantly closer to `man` for stereotypical male jobs. We find that especially the debiased Google News and Word Dependency embeddings are less biased.



(a) L^1- norm

(b) L^2- norm

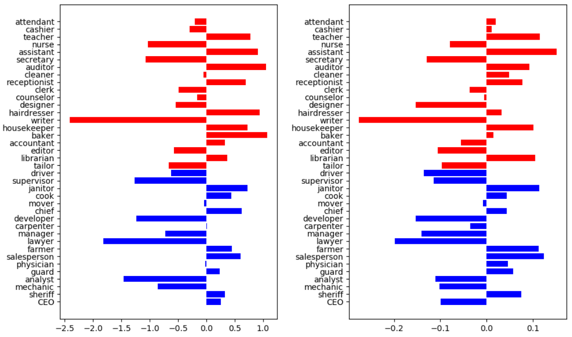
(c) GloVe embeddings trained on Common Crawl



(d) L^1- norm

(e) L^2- norm

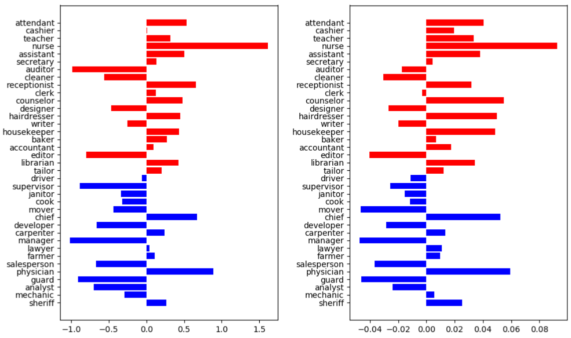
(f) Word2Vec embeddings trained on Google News



(a) L^1- norm

(b) L^2- norm

(c) Turian embeddings



(d) L^1- norm

(e) L^2- norm

(f) Word Dependency embeddings

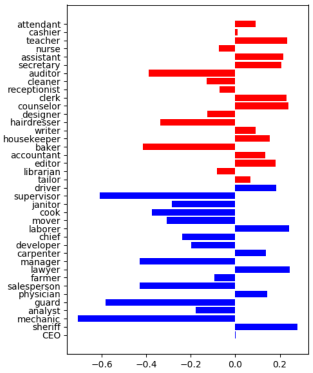


Figure 5: L^1- norm

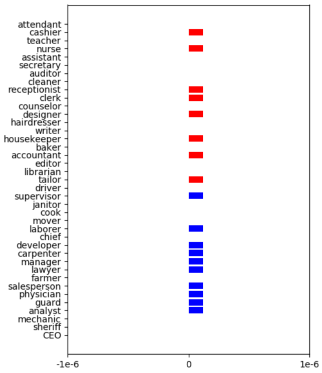


Figure 6: L^2- norm

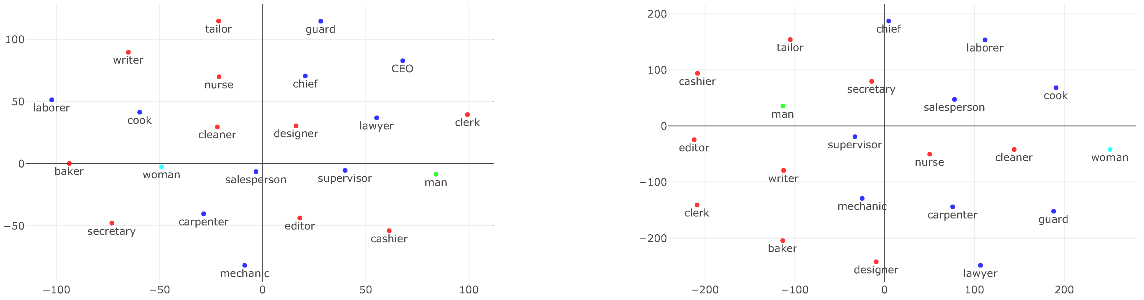
Figure 7: Word2Vec debiased embeddings trained on Google News

C.1 T-SNE visualization of embeddings

Figure 8a - Figure 10 show a 2D visualization of the different word embeddings. We places the typical female and male jobs in this 2D grid together with the words `man` and `woman`. We colored the typical male jobs blue and the typical female jobs red. It is expected that female and male jobs do not cluster together in an unbiased embedding and that the distances between `man` / `woman` is equal independent on male or female job.

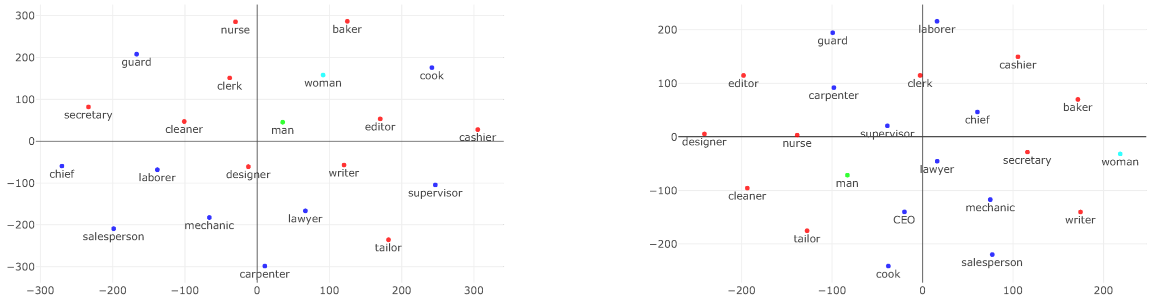
We find that in the embeddings with high bias from Appendix C the female and male job titles seem to cluster together. We also find that these embeddings have a large distance between `man` and `woman`. Contrary, in e.g. the Word Dependency

Embeddings (Figure 9a) that `man` and `woman` are very close. Comparing the GoogleNews Embedding Figure 9b with the Debiased Google News Embedding Figure 10 we see that the gender-based clusters are only apperant in the GoogleNews Embedding. These findings confirms that the both the Word Dependency Embeddings and the GoogleNews Debiased Embeddings are gender unbiased.



(a) GloVe Embedding

(b) HLBL Embedding



(a) Word Dependency Embedding

(b) GoogleNews Embedding

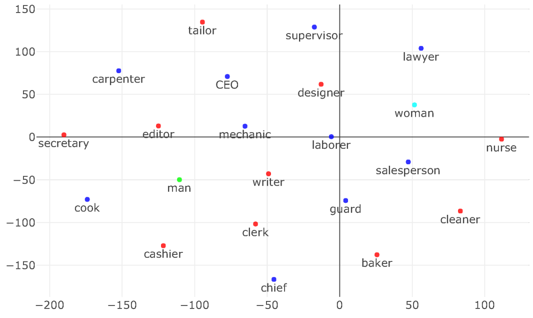


Figure 10: GoogleNews Debiased Embedding

D Parameter tuning

Here we tried altering the Loss function to see which resulted in the best performance for the model. Results for the different loss functions are reported for the RNN attention model described in Section 3.2. Results are reported with their bootstrap confidence intervals.

Loss function	**F1 [CI]**	**logloss [CI]**	**Bias [CI]**
Categorical Cross-entropy	0.558 [0.266; 0.732]	10.074 [9.429; 10.767]	0.964 [0.584; 1.629]
Binary cross-entropy	0.541 [0.240; 0.714]	10.573 [9.912; 11.283]	0.986 [0.850; 4.983]
Squared hinge	0.604 [0.351; 0.74]	9.784 [9.22; 10.428]	0.979 [0.819; 1.95]
Hinge	0.596 [0.351; 0.726]	10.283 [9.671; 10.928]	0.983 [0.771; 1.672]
Logcosh	0.584 [0.319; 0.735]	9.977 [9.332; 10.670]	1.024 [0.895; 3.559]
KL divergence	0.569 [0.308; 0.732]	10.090 [9.445; 10.783]	1.008 [0.88; 3.968]
Poisson	0.532 [0.206; 0.733]	10.348 [9.67; 10.993]	0.985 [0.834; 4.064]
Cosine similarity	0.504 [0.167; 0.734]	10.299 [9.655; 10.928]	0.960 [0.449; 2.131]

Table 7: Performance with respect to different loss functions

E Pronoun scheme performance

Training	**Test**	**F1 [CI]**	**logloss [CI]**	**Bias [CI]**
Original	Original	0.601 [0.360; 0.723]	10.074 [9.413; 10.735]	0.978 [0.838; 2.087]
	It	0.592 [0.352; 0.718]	10.203 [9.668; 10.880]	0.996 [0.859; 2.499]
	One	0.603 [0.361; 0.723]	10.154 [9.494; 10.815]	0.991 [0.842; 2.023]
	They	0.583 [0.354; 0.713]	10.235 [9.574; 10.896]	0.987 [0.835; 2.202]
	Who	0.580 [0.354; 0.716]	10.267 [9.590; 10.928]	0.975 [0.830; 2.065]
	Spivak	0.601 [0.381; 0.716]	10.203 [9.542; 10.880]	0.982 [0.835; 1.983]
	Ne	0.563 [0.321; 0.707]	10.477 [9.832, 11.170]	0.979 [0.837; 2.789]
It	Original	0.575 [0.328; 0.725]	10.074 [9.413; 10.735]	0.999 [0.851; 2.151]
	It	0.570 [0.311; 0.723]	10.171 [9.510; 10.831]	0.997 [0.848; 2.316]
One	Original	0.547 [0.282; 0.724]	10.203 [9.542; 10.880]	0.975 [0.840; 2.591]
	One	0.581 [0.332; 0.720]	10.090 [0.429; 10.735]	0.971 [0.827; 2.215]
They	Original	0.565 [0.303; 0.709]	10.573 [9.945; 11.234]	1.006 [0.833; 2.262]
	They	0.548 [0.292; 0.629]	10.880 [10.235; 11.573]	0.987 [0.819; 2.370]
Who	Original	0.576 [0.289; 0.727]	10.251 [9.606; 10.928]	0.994 [0.824; 2.303]
	Who	0.597 [0.340; 0.730]	9.993 [9.348; 10.638]	0.984 [0.811; 1.885]
Spivak	Original	0.573 [0.330; 0.713]	10.557 [9.880; 11.234]	0.990 [0.815; 1.907]
	Spivak	0.547 [0.281; 0.714]	10.541 [9.897; 11.218]	0.984 [0.708; 1.871]
Ne	Original	0.533 [0.253; 0.696]	10.654 [9.993; 11.331]	0.949 [0.781; 2.771]
	Ne	0.555 [0.299; 0.715]	10.122 [9.477; 10.767]	0.984 [0.844; 2.959]

Table 8: Performance with respect to different gender-neutral pronoun schemes used for data augmentation. Confidence intervals where calculated using the bootstrap algorithm, 3000 iterations. *Training* indicates which scheme was applied to training and development data, *Test* indicates the scheme applied to test data. Best performance respectively among precedented and engineered pronoun schemes are highlighted.

F (Lee et al., 2017) E2E Coreference model architecture

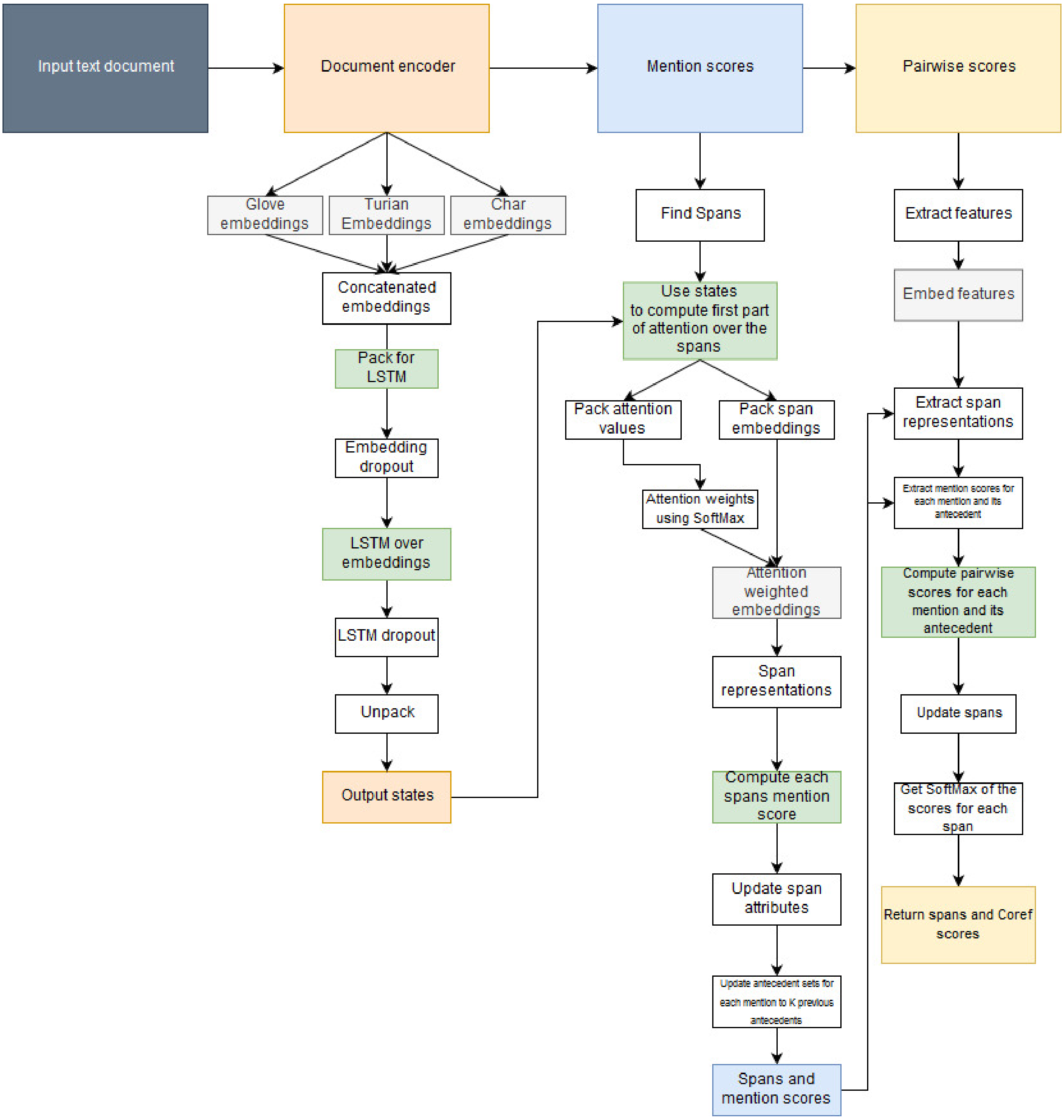
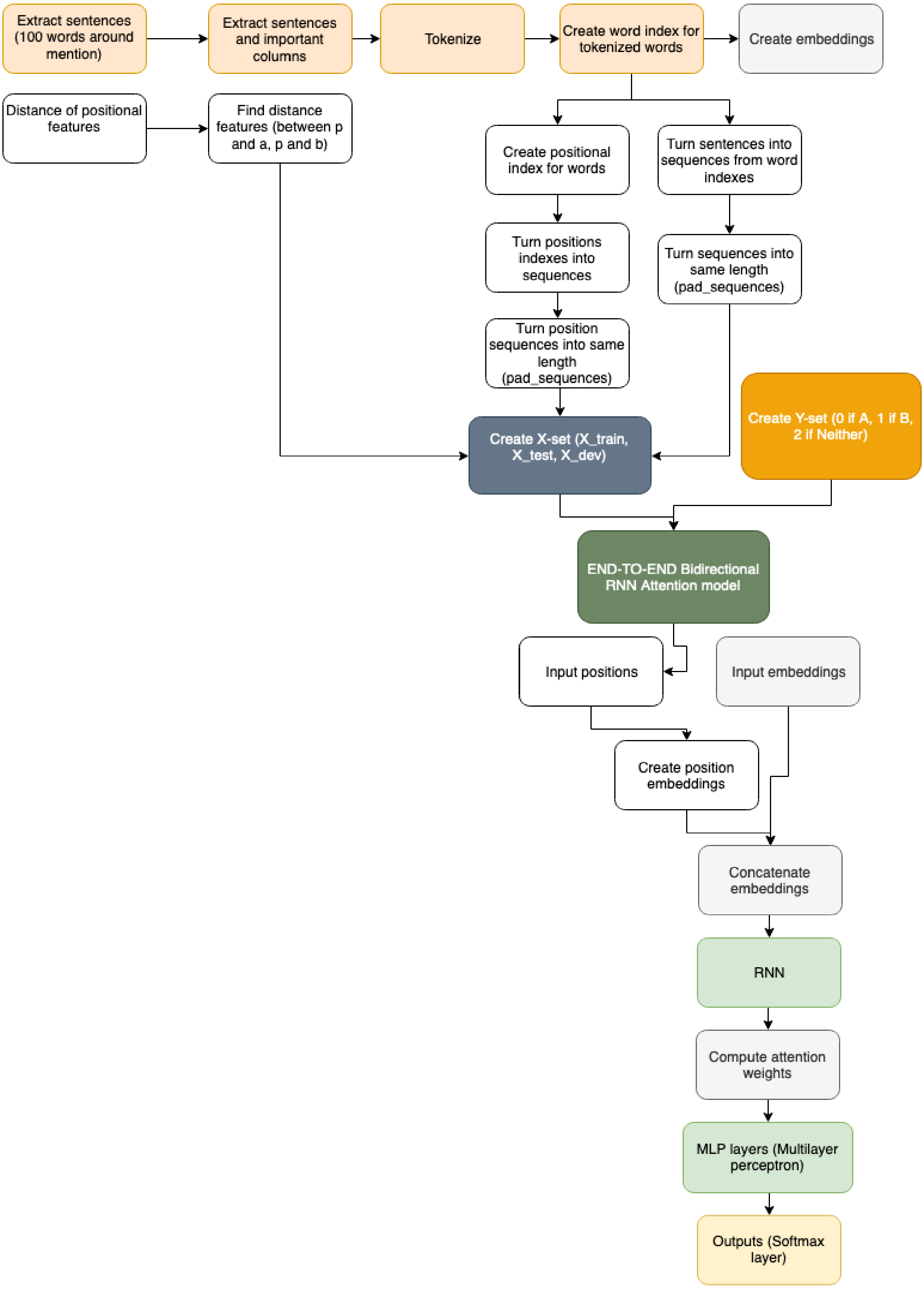

Input text document
Document encoder
Mention scores
Pairwise scores
Glove embeddings
Turian Embeddings
Char embeddings
Concatenated embeddings
Pack for LSTM
Embedding dropout
LSTM over embeddings
LSTM dropout
Unpack
Output states
Find Spans
Use states to compute first part of attention over the spans
Pack attention values
Pack span embeddings
Attention weights using SoftMax
Attention weighted embeddings
Span representations
Compute each spans mention score
Update span attributes
Update antecedent sets for each mention to K previous antecedents
Spans and mention scores
Extract features
Embed features
Extract span representations
Extract mention scores for each mention and its antecedent
Compute pairwise scores for each mention and its antecedent
Update spans
Get SoftMax of the scores for each span
Return spans and Coref scores

Figure 11: (Lee et al., 2017) model

G RNN Attention model architecture



Extract sentences (100 words around mention)
Extract sentences and important columns
Tokenize
Create word index for tokenized words
Create embeddings
Distance of positional features
Find distance features (between p and a, p and b)
Create positional index for words
Turn sentences into sequences from word indexes
Turn positions indexes into sequences
Turn sequences into same length (pad_sequences)
Turn position sequences into same length (pad_sequences)
Create Y-set (0 if A, 1 if B, 2 if Neither)
Create X-set (X_train, X_test, X_dev)
END-TO-END Bidirectional RNN Attention model
Input positions
Input embeddings
Create position embeddings
Concatenate embeddings
RNN
Compute attention weights
MLP layers (Multilayer perceptron)
Outputs (Softmax layer)

Figure 12: RNN attention model