Perturbation Sensitivity Analysis to Detect Unintended Model Biases

Vinodkumar Prabhakaran
Google Brain
San Francisco, CA, USA
vinodkpg@google.com

Ben Hutchinson
Google Brain
San Francisco, CA, USA
benhutch@google.com

Margaret Mitchell
Google Brain
Seattle, WA, USA
mmitchellai@google.com

Abstract

Data-driven statistical Natural Language Processing (NLP) techniques leverage large amounts of language data to build models that can understand language. However, most language data reflect the public discourse at the time the data was produced, and hence NLP models are susceptible to learning incidental associations around named referents at a particular point in time, in addition to general linguistic meaning. An NLP system designed to model notions such as sentiment and toxicity should ideally produce scores that are independent of the identity of such entities mentioned in text and their social associations. For example, in a general purpose sentiment analysis system, a phrase such as *I hate Katy Perry* should be interpreted as having the same sentiment as *I hate Taylor Swift*. Based on this idea, we propose a generic evaluation framework, *Perturbation Sensitivity Analysis*, which detects unintended model biases related to named entities, and requires no new annotations or corpora. We demonstrate the utility of this analysis by employing it on two different NLP models — a sentiment model and a toxicity model — applied on online comments in English language from four different genres.

1 Introduction

Recent research has shown ample evidence that data-driven NLP models may inadvertently capture, reflect and sometimes amplify various social biases present in the language data they are trained on (Bolukbasi et al., 2016; Blodgett and O'Connor, 2017). Such biases can often result in unintended and disparate harms to the users who engage with NLP-aided systems. For instance, when NLP algorithms are used to moderate online communication, e.g. by detecting harassment, although the net social benefits may be positive, the harms caused by incorrect classifications may be unevenly distributed, leading to disparate impact (Feldman et al., 2015). Some writers may find their contributions being disproportionately censored, while some readers may not be adequately protected from harassment (Dixon et al., 2018).

Sentence	Toxicity	Sentiment
I hate Justin Timberlake.	0.90	-0.30
I hate Katy Perry.	0.80	-0.10
I hate Taylor Swift.	0.74	-0.40
I hate Rihanna.	0.69	-0.60

Table 1: Sensitivity of NLP models to named entities in text. Toxicity score range: 0 to 1; Sentiment score range: -1 to +1.

Research into fairness in machine learning distinguishes two broad categories of unfair discrimination. First, unfairness for *individuals* exists when similar individuals are treated dissimilarly (Dwork et al., 2012). Second, a range of criteria define unfairness for *groups*, each in terms of statistical dependence between group membership, model score, and class label (see, e.g., (Chouldechova and Roth, 2018; Mitchell et al., 2018)). In both cases, what is "fair" or "unfair" is highly context-dependent, and judgments about fairness require consideration of the broader sociotechnical frame (Selbst et al., 2019).

This framework also poses some practical challenges: individual fairness requires knowing intricate details about an individual, while group fairness requires knowing how an individual can be categorized into legally and socially sensitive roles. The first runs into the ethical concerns of surveillance; the second runs into the ethical concerns of discrimination. Furthermore, texts are often not annotated with the social groups of their readers/writers (and for privacy reasons we may not wish to infer them), or whether two individuals are "similar" or not. Hence, fairness research in NLP has mostly focused on mentions of social identities (Dixon et al., 2018; Borkan et al., 2019; Garg et al., 2019), or on how social stereotypes impact semantic interpretation (Webster et al., 2018), and often rely heavily on annotated corpora.

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 5740–5745,
Hong Kong, China, November 3–7, 2019. ©2019 Association for Computational Linguistics

In this paper, we propose a general-purpose evaluation framework that detects unintended biases in NLP models around named entities mentioned in text. Our method does not rely on any annotated corpora, and we focus solely on application-independent *sensitivity* of models, which does not clearly fall under individual- or group- based fairness criteria. Our core idea is based on the assumption that an NLP system designed to be widely applicable should ideally produce scores that are independent of the identities of named entities mentioned in the text. For instance, the sentences *I hate Justin Timberlake* and *I hate Rihanna* both express the same semantics using identical constructions; however, the toxicity model used in our experiments gives a significantly higher score to the former (0.90) than the latter (0.69) (see Table 1 for more examples).

Mentions of such real-world entities are pervasive in data. Just as word co-occurrence metrics capture 'meaning representations' of words in the language,[1] co-occurrence patterns between entity mentions and other parts of the phrases they occur in influence their learned meaning. For example, if a person's name is often mentioned in negative linguistic contexts, a trained model might inadvertently associate negativity to that name, resulting in biased predictions on sentences with that name. If unchecked, this leads to undesirable biases in the model, violating tenets of both individual and group fairness as they are applied in context.

The primary contributions of this paper are: (i) a simple and effective general-purpose model evaluation metric, which we call *perturbation sensitivity analysis*, for measuring unintended bias; (ii) a large-scale systematic analysis of model sensitivity to name perturbations, on two tasks – sentiment and toxicity – across four different genres of English text; (iii) a demonstration of how name perturbation can reveal undesired biases in the learned model towards names of popular personalities; (iv) showing the downstream impact of name sensitivity, controlling for prediction thresholds.

2 Perturbation Sensitivity Analysis

We introduce *Perturbation Sensitivity Analysis (PSA)*, a general evaluation technique to detect unintended biases in NLP models towards real-world entities. Central to our approach is the notion of *perturbation*, where a reference to a real-world entity is replaced by a reference to another real-world entity of the same type (e.g., a person name replaced with another person name). PSA measures the extend to which a model prediction is sensitive to such perturbations, and is calculated w.r.t. a set of (unannotated) sentences X from the target domain and a set of names N (of the same entity type t) that the perturbations are performed over.

For simplicity, in this paper, we discuss text classification models that take in a piece of text and return a score for a target class. Similarly, we focus on perturbing over person names. However, our method is readily extendable to other kinds of models as well as to other entity types.

Our approach begins by first retrieving the set of sentences X such that each sentence contains at least one referring expression that refers to an entity of the type we are doing perturbation on (person, in our case). This referring expression could be a pronoun or a proper name. We select one such referring expression as the *anchor* for each sentence in X. We then "perturb" each sentence by replacing the anchor with named entities $n \in N$. We then measure the sensitivity of the model with respect to such perturbation by running it on the resulting set of $|X| * |N|$ perturbed sentences.

Formally, let x_n denote the perturbed sentence obtained by replacing the anchor word in $x \in X$ with n, and $f(x_n)$ denote the score assigned to a target class by model f on the perturbed sentence x_n. Formally, we define three metrics for the *perturbation sensitivity of model scores*:

Perturbation Score Sensitivity (*ScoreSens*) of a model f with respect to a corpus X and a name n is the average difference between $f(x_n)$ and $f(x)$ calculated over X, i.e. $\underset{x \in X}{E}[f(x_n) - f(x)]$.

Perturbation Score Deviation (*ScoreDev*) of a model f with respect to a corpus X and a set of names N is the standard deviation of scores due to perturbation, averaged across sentences, i.e., $\underset{x \in X}{E}[\underset{n \in N}{StdDev}(f(x_n)]$.

Perturbation Score Range (*ScoreRange*) of a model f with respect to a corpus X and a set of names N is the $Range$ ($max - min$) of scores, averaged across sentences, i.e., $\underset{x \in X}{E}[\underset{n \in N}{Range}(f(x_n)]$.

Whether a score difference caused by name perturbation results in a different label depends also on the threshold. Given a threshold, $0 \leq c \leq 1$,

[1] Often through word embeddings fed to or learned by the first layer of neural network based models

binary labels $y(x)$ can be obtained from the classifier f as $\mathbf{I}[f(x) \geq c] \in \{0, 1\}$, where $\mathbf{I}[\cdot]$ is the indicator function. Using this, we define a metric for the *perturbation sensitivity of model labels*:

Perturbation Label Distance (*LabelDist*) of a binary classifier y with respect to a corpus X and a set of names N is the Jaccard Distance between a) the set of sentences $\{x\}$ for which $y(x) = 1$, and b) the sentences $\{x\}$ for which $y(x_n) = 1$, averaged across names $n \in N$; i.e., $\underset{n \in N}{E}[Jaccard(\{x|y(x) = 1\}, \{x|y(x_n) = 1\})]$, where $Jaccard(A, B) = 1 - |A \cap B|/|A \cup B|$.

2.1 Assumptions

The underlying assumption of PSA is that the model should ideally be *not* sensitive to name perturbation. However, this assumption may not always hold true. Proper names *do* convey meaning akin to the linguistic meanings expressed in more general phrases, and thus perturbing names may sometimes remove critical semantic content that an NLP system should be modelling. For example, *he is like Hitler* vs. *he is like Gandhi* should have very different sentiment scores, since the sentences evoke the pragmatics associated with those referents. Whether the PSA assumption holds in individual sentences will depend on the sentential context; however, the corpus-level trends that we measure in the model scores/labels are still indicative of systemic biases in the model. This points to the importance of paying care to how the corpus X is constructed, and making sure that it captures a diverse set of sentential contexts.

2.2 Analysis Framework

The PSA framework described above is applicable to any text classification models, on any target corpus, to detect bias with respect to any type of named entities (i.e., perturbable among each other). In this paper, we focus on two text classification models, applied to 4 different corpora, to detect biases associated with person names.

Models: We use two text classification models: a) a toxicity model that returns a score between [0,1], and b) a sentiment model that returns a score between [-1,+1]. Both models were trained using state-of-the-art neural network algorithms, and perform competitively on benchmark tests.[2]

[2]To obtain information about the models, for instance to perform replication experiments, please contact the authors.

Corpora: We use four socially and topically diverse corpora of online comments released by Voigt et al. (2018): Facebook comments on politicians' posts (FB-Pol.) and on public figures' posts (FB-Pub.), Reddit comments, and comments in Fitocracy forums. For each corpus, we select 1000 comments at random that satisfy two criteria: at most 50 words in length, and contain at least one English 3rd person singular pronouns (i.e., anchors). We use these extracted comments to build templates, where the pronouns can be replaced with a person name to form a grammatically coherent perturbed sentence. We use pronouns as the anchors for multiple reasons. Pronouns are often *closed-class* words across languages,[3] making it a useful reference cross-linguistically. Using a list of names to query for anchors is an option; but it has the risk of resulting in a set of sentences biased towards the cultural/demographic associations of those names, a confound that the use of pronouns as anchors will avoid. We balance the representation of female and male pronouns in our extracted sentences so as to minimize the effect of skew towards one gender in particular within the test set. However future work should examine how to best account for non-binary genders in this step.

Names: We choose a list of controversial personalities, compiled based on Wikipedia page edit frequency.[4] Because of their controversial nature, these names are more likely to have social biases associated with them, which is helpful to demonstrate the utility of our analysis.

3 Results

Table 2 shows the results of perturbation sensitivity analysis on different corpora. Both models exhibit significant sensitivity towards name perturbation across all 4 corpora. On average, sentences subjected to name perturbation resulted in a wide range of scores; i.e., *ScoreRange* over 0.10 for toxicity, and 0.36-0.42 for sentiment. Similarly, *ScoreDev* values for the sentiment model is also higher (over 0.07 across board) compared to that of the toxicity model (around 0.02), suggesting that the sentiment model is much more sensitive to the named entities present in text than the toxicity model. We also observe that perturbation

[3]While the assumption that pronouns are a closed-class is useful for many languages, Japanese and Malay are example languages where this assumption does not hold.

[4]https://anon.to/x9PMYo

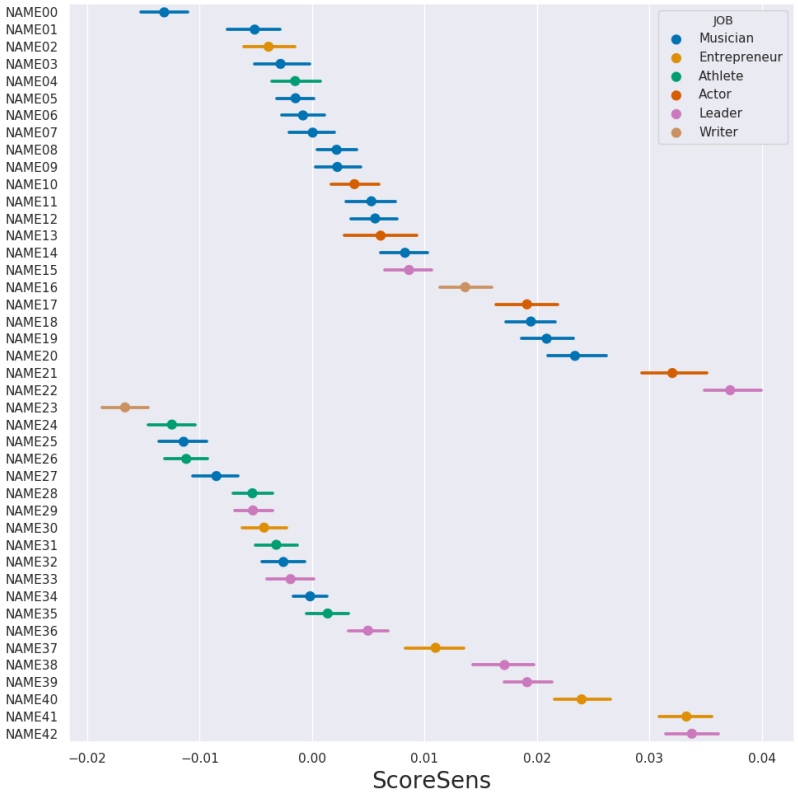

Figure 1: Name Perturbation Sensitivity (*ScoreSens*) for the toxicity model on the Reddit subcorpus, across names of controversial personalities. Female names are at the top; male names at the bottom; colors distinguish their career type. Names have been obfuscated due to their sensitive nature.

	Toxicity		Sentiment	
Corpus	*ScoreDev*	*ScoreRange*	*ScoreDev*	*ScoreRange*
FB-Pol.	0.022	0.107	0.070	0.360
FB-Pub.	0.025	0.118	0.083	0.420
Reddit	0.022	0.107	0.072	0.376
Fitocracy	0.022	0.103	0.071	0.364

Table 2: *ScoreDev* is the per-sentence standard deviation of scores upon name perturbation, averaged across all sentences. *ScoreRange* is the per-sentence range of scores (i.e., max - min) upon name perturbation, averaged across all sentences.

sensitivity is a function of the target corpus; comments on public figures had a much larger *ScoreDev* and *ScoreRange* for both tasks.

3.1 Bias Towards Specific Names

We now analyze the *ScoreSens* for specific names. Figure 1 shows the *ScoreSens* for each name in our list, for the Toxicity-Reddit combination. Names are obfuscated in the figure due to their sensitive nature, but their career type is distinguished. Replacing a pronoun with some names increases the toxicity scores by over 0.03 on average, while other names decrease the scores by almost 0.02 on average. It is also notable that leaders (politicians) and actors in our list have higher toxicity associations than musicians and athletes. Similar effects also occur in the sentiment analysis model.

3.2 Threshold Analysis

Whether a score difference caused by perturbation results in a different label or not depends also on the threshold. It is possible that a model would be more stable on sentences with highly toxic language, but the effect of perturbation is more prevalent in sentences that have fewer signals of toxicity. We verified this to be the case in our analysis: the average (over all names) value of the perturbation score sensitivity, i.e. $|f(x_n) - f(x)|$, has a significant moderate negative correlation (-0.48) with the original score of that sentence, $f(x)$. This finding is of importance to counter-factual token fairness approaches such as (Garg et al., 2019).

To further understand the impact of perturbation sensitivity, we calculate *LabelDist*, which takes into account the number of sentences that switch either from toxic to non-toxic or vice versa, when a pronoun is changed to a name. Figure 2 shows *LabelDist* values across different thresholds. As can be seen from the Figure, the name perturbation

results in a *LabelDist* of 0.10 – 0.40 across thresholds. This roughly suggests that around 10-40% of sentences (with third person singular pronouns) labeled as toxic at any given threshold could flip the label as a result of name perturbation. It is also interesting to note that despite the negative correlation between $|f(x_n) - f(x)|$ and $f(x)$, the *LabelDist* has high values at high thresholds.

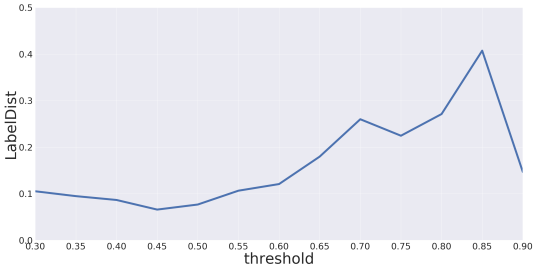

Figure 2: Even for high model thresholds, we see significant name perturbation sensitivity in classifications/labels. *LabelDist* measures the # of flips between *toxic* and *non-toxic*.

4 Related Work

Fairness research in NLP has seen tremendous growth in the past few years (e.g., (Bolukbasi et al., 2016; Caliskan et al., 2017; Webster et al., 2018; Díaz et al., 2018; Dixon et al., 2018; De-Arteaga et al., 2019; Gonen and Goldberg, 2019; Manzini et al., 2019)) over a range of NLP tasks such as co-reference resolution and machine translation, as well as the tasks we studied — sentiment analysis and toxicity prediction. Some of this work study bias by swapping names in sentence templates (Caliskan et al., 2017; Kiritchenko and Mohammad, 2018; May et al., 2019; Gonen and Goldberg, 2019); however they use synthetic sentence templates, while we extract naturally occurring sentences from the target corpus.

Our work is closely related to counter-factual token fairness (Garg et al., 2019), which measures the magnitude of model prediction change when identity terms (such as *gay*, *lesbian*, *transgender* etc.) referenced in naturally occurring sentences are perturbed. Additionally, De-Arteaga et al. (2019) study gender bias in occupation classification using names in online biographies. In contrast, we propose a general framework to study biases with named entities. Our work is also related to the work on interpreting neural network models by manipulating input text (Li et al., 2016); while their aim is to interpret model weights, we study the model outputs for biases.

5 Discussion and Conclusion

Social biases towards real-world entities are often reflected in the language that mentions them, and such biases can be unintentionally perpetuated to trained models. The focus of this paper is to introduce a simple method, Perturbation Sensitivity Analysis, to test for unwanted biases in an NLP model. Our method can be employed to study biases towards individuals (as demonstrated here), or towards groups (e.g., races, genders), and is flexible enough to be applied across domains.

We are motivated to provide solutions for end users of NLP systems, who are likely to use models outside of their original training/testing environments, e.g., on data from populations or platforms that the system was not explicitly trained on. The relative simplicity of the proposed approach suggests that the same method may be applied in different genres and across different languages, provided that a set of anchors are provided, such as pronouns in the target language. Pronouns' status cross-linguistically as closed-class – high frequency and easily listed as a small set of words – make them particularly amenable for serving as a starting point for open domain bias analyses.

After identifying unwanted biases in a model, a next logical step is to reduce these biases. Adapting the proposed approach to model training is straightforward, either by perturbing names in the training data directly, or by estimating the likelihood of given annotations as a function of sentence perturbation. Without access to model retraining, a simple solution could use post-processing to return system scores as a function of perturbed sentences, such as by averaging scores across perturbed sentences.

Future work could employ our method to study various group biases such as nationality, caste, and religion, since person names may function as significant markers for many such demographic associations. Our method could also be easily extended to other kinds of NLP models (beyond classification) as well as other types of entities (locations, organizations etc.).

Acknowledgements We would like to thank the anonymous reviewers for their helpful and constructive feedback. We also thank Dylan Baker, Emily Denton, Yoni Halpern, Ben Packer, Lucy Vasserman, Kellie Webster, and Simone Wu for their valuable discussions on this paper.

References

Su Lin Blodgett and Brendan O'Connor. 2017. Racial disparity in natural language processing: A case study of social media African-American English. *arXiv preprint arXiv:1707.00061*.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in neural information processing systems*, pages 4349–4357.

Daniel Borkan, Lucas Dixon, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2019. Nuanced metrics for measuring unintended bias with real data for text classification. In *Proceedings of the FATES 2019 Workshop on Fairness, Accountability, Transparency, Ethics, and Society on the Web*.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Alexandra Chouldechova and Aaron Roth. 2018. The frontiers of fairness in machine learning. *arXiv preprint arXiv:1810.08810*.

Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, and Adam Tauman Kalai. 2019. Bias in bios: A case study of semantic representation bias in a high-stakes setting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 120–128. ACM.

Mark Díaz, Isaac Johnson, Amanda Lazar, Anne Marie Piper, and Darren Gergle. 2018. Addressing age-related bias in sentiment analysis. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, page 412. ACM.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73. ACM.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd innovations in theoretical computer science conference*, pages 214–226. ACM.

Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. 2015. Certifying and removing disparate impact. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 259–268. ACM.

Sahaj Garg, Vincent Perot, Nicole Limtiaco, Ankur Taly, Ed H Chi, and Alex Beutel. 2019. Counterfactual fairness in text classification through robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, pages 219–226. ACM.

Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *Proceedings of NAACL*, pages 609–614, Minneapolis, Minnesota. Association for Computational Linguistics.

Svetlana Kiritchenko and Saif Mohammad. 2018. Examining gender and race bias in two hundred sentiment analysis systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 43–53, New Orleans, Louisiana. Association for Computational Linguistics.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*.

Thomas Manzini, Lim Yao Chong, Alan W Black, and Yulia Tsvetkov. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. In *Proceedings of NAACL*, pages 615–621, Minneapolis, Minnesota. Association for Computational Linguistics.

Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. In *Proceedings of NAACL*, pages 622–628, Minneapolis, Minnesota. Association for Computational Linguistics.

Shira Mitchell, Eric Potash, and Solon Barocas. 2018. Prediction-based decisions and fairness: A catalogue of choices, assumptions, and definitions. *arXiv:1811.07867*.

Andrew D Selbst, Danah Boyd, Sorelle A Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*. ACM.

Rob Voigt, David Jurgens, Vinodkumar Prabhakaran, Dan Jurafsky, and Yulia Tsvetkov. 2018. Rtgender: A corpus for studying differential responses to gender. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Kellie Webster, Marta Recasens, Vera Axelrod, and Jason Baldridge. 2018. Mind the gap: A balanced corpus of gendered ambiguous pronouns. *Transactions of the Association for Computational Linguistics*, 6:605–617.