Pronoun Resolution Based on Syntactic Features and Machine Learning Algorithms

Berta Franzluebbers
berta@uga.edu

Narinder Ghumman
ng04111@uga.edu

Shulin Zhang
sz38235@uga.edu

Abstract

In this project, we considered the problem of pronoun resolution from a statistical natural language processing perspective. We used the GAP dataset for our study, and devised some linguistically informed features for representing text and building machine learning classifiers. GAP poses a more difficult version of the pronoun resolution problem by proposing antecedents which already agree with the ambiguous pronoun in gender and number, rendering any methods for resolving between the two candidates based on these attributes unhelpful. Our results show that in terms of features, universal dependency tags and the depth in the dependency tree were the most valuable, and in terms of methods, multi-layer perceptron and random forest did the best. Overall our classifier improves on the GAP baseline by +4.4%. When the trained classifier was evaluated on new data from the Penn Treebank, comparable results were obtained.

1 Introduction

Anaphora describes the linguistic phenomenon of referring to a previously mentioned entity, the antecedent. Anaphora resolution refers to the process of locating that entity. This task is essential when comprehending language since it allows the listener to merge existing information about an entity with new information encountered. In many cases there are several objects to choose from, which can result in ambiguity, for example:

1. If an incendiary bomb drops next to you, don't lose your head. Put it in a bucket and cover it with sand.

In this sentence, from a British WWII leaflet, *it* can refer to either *bomb* or *head*. The correct antecedent, is of course *bomb*. However, this is only obvious based on semantics. In general, the English language would actually prefer the antecedent to be the more recently mentioned entity. This example is a short introduction to the difficulty of anaphora resolution from a computational perspective. Despite the difficulty of correct resolution, it is essential in order to collect all information about an object in order to expand the knowledge base and make as many new inferences as possible. Important applications which depend on correct anaphora resolution are machine translation, automatic abstracting, information abstraction, and question answering (Mitkov, 1999).

There are many types of anaphora, but this project will focus on pronoun resolution. The general procedure for pronoun resolution is quite standard: all noun phrases (NPs) preceding the pronoun are taken into consideration as potential candidates. Usually the scope is limited to NPs in the current and preceding sentence. Ideally, the system should extend the scope further, since there have been examples with antecedents 17 sentences away (Mitkov, 1995).

Once the potential antecedent candidates have been identified, a set of factors are used to make the resolution choice. The factors can be preferential, referred to as *preferences* or *proposers*, or eliminating (excluding certain NPs from consideration), referred to as *constraints*. Some authors claim all factors should be seen as preferential, with higher or lower weights (Preuss, 1992). Some examples of preferential factors are parallelism and salience, while some examples of constraints are gender and number constraints, c-command constraints, and semantic consistency (Mitkov, 1999).

Traditionally, these factors are incorporated into a statistical approach using the co-occurrence patterns observed in a corpus (Dagan and Itai, 1990). In 1994, Connolly, Burger, and Day used the idea of treating anaphoric reference as a classification problem using machine learning techniques. The clas-

sifier is applied to successive pairs of antecedent candidates, retaining the best candidate (Connolly et al.). Later work has focused on a knowledge-poor approach, since relying heavily on linguistic knowledge is more labor-intensive. An example, is Kennedy and Boguraev's approach which uses only part of speech tags and not full syntactic parsing (Kennedy and Boguraev, 1996). Recent work in anaphora resolution relies on contextual word embeddings, such as from Google's Bidirectional Encoder Representations from Transformer (BERT).

Our approach is a return to statistical machine learning techniques, using knowledge-rich linguistic features. This technique allows for an interesting comparison of the effectiveness among features. Our method is notable for being agnostic to both gender and number agreement. In addition, we explore the addition of features based on syntactic binding theory, using an approximation based on the dependency parse, removing the reliance on either hand-annotated syntactic trees or "noisy" machine-generated trees to generate this feature.

2 Task

This project uses the GAP dataset, which contains 8,908 coreference-labeled pairs of the form (ambiguous pronoun, antecedent name). These data are gathered from Wikipedia, and are notable for being gender-balanced, since most datasets have an overwhelming majority of male pronouns. When coreference accuracy is lower for female pronouns, the resulting error creates inequity in user experience for underrepresented groups in downstream applications (Webster et al., 2018).

The dataset is prepared with two names labeled A and B in each paragraph, and one ambiguous pronoun already selected for resolution. That is, the data contains the character offset for A, B, and pronoun from the beginning of the paragraph, as well as the string value for each (names can consist of a single or multiple word(s)). It is notable that both A and B always have the same gender within a paragraph, preventing the use of gender and number as distinguishing features. It is also possible that the ambiguous pronoun corefers with neither A nor B.

3 Method

Our method includes exploration of potential features generated from the sentence or paragraph to tackle the coreference relationship between a pronoun and a name. The features are then used to train machine learning algorithms, and the coreference labeling accuracy is used to reflect how helpful these features are. In this section, we will introduce the motivation for all the features and algorithms adopted in this study. The main feature generation steps are shown in Figure 1, which shows that we rely on tokenization and dependency parse results from spaCy, a toolkit equipped with pre-trained statistical language models. The English model we use (`en-core-web-sm`) is a convolutional neural network trained on data from OntoNotes 5 (Honnibal and Montani, 2017).

3.1 Feature Generation

3.1.1 Features Provided by the GAP Dataset

The GAP dataset provides the offset position for the aimed proper names A/B and the pronoun, shown in Table 2 as `Offset Position` (*'offset'*). Including the offset as a feature is motivated by recency - more recently mentioned words are more salient in the discourse, and more likely to be replaced with a pronoun (Jurafsky and Martin, 2019). Apart from this position, GAP also has a binary representation for whether A - Pronoun and B - Pronoun are coreferred, shown in Table 2 as `Coreference Label` (*'coref'*), which we use as our gold standard for training and testing.

3.1.2 Features Based On Dependency Structures

Features `Is Subject` (*'nsubj'*), and `Node depth in dependency tree` (*'depth'*) are generated with spaCy's dependency parsing results (Honnibal and Montani, 2017). An example sentence is parsed in Figure 2. The motivation for including the dependency tag as a feature is that grammatical role is also an indicator of saliency in the discourse. Subjects are more salient than objects, which are more salient than words in oblique roles (Jurafsky and Martin, 2019). In addition, the pronoun and its coreferred proper name might have a similar depth in the dependency tree.

3.1.3 Features Inspired By Hobbs Algorithm

The feature `Noun Phrase numbers in between` (*'NPs'*) is inspired by the Hobbs Algorithm. This classic corefrence algorithm traverses the syntactic tree starting at the position of the pronoun and orders potential noun phrases based on the rules of the algorithm (Hobbs and Shieber, 1987). A simplified version from Kehler

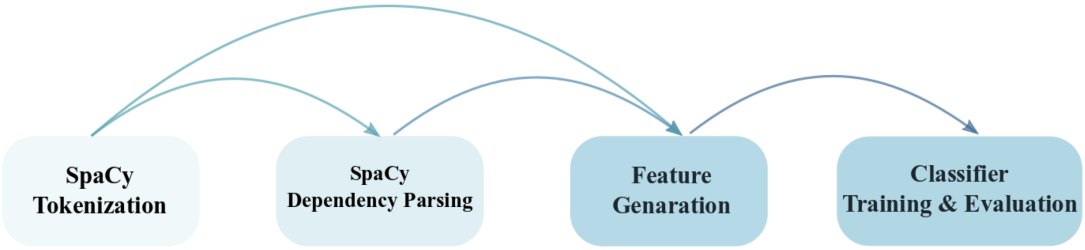

Figure 1: Feature generation process: With spaCy, a toolkit equipped with pre-trained statistical language models, we can obtain tokenization and dependency parsing results for each text instance in the GAP dataset. The generated features described in Table 2 are based on spaCy's parsing output, and they are fed into machine learning models (Section 3.2) to test how valuable they are for the pronoun resolution task.

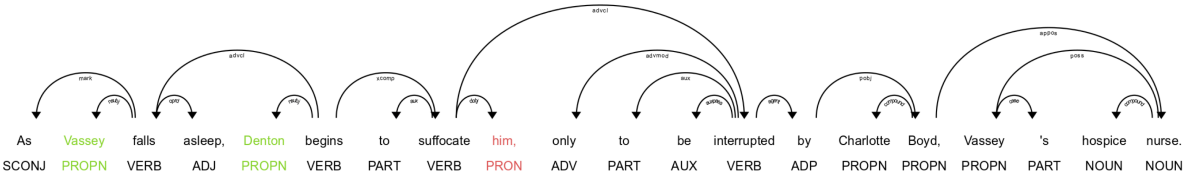

Figure 2: A sentence dependency parsed with spaCy: Each text instance in GAP has two aimed proper names and one target pronoun. Proper names (tagged above as 'PROPN', proper name) are labeled as 'A' and 'B' in GAP, and Pronouns (tag above as 'PRON', pronoun) will be assigned truth value for their coreference with A and B. In this parsing result, the green entities are proper names (*i.e.* A and B), and the red one is the pronoun, which is coreferred with "Vassey".

et al. describes the search order as follows: starting in the current sentence from right-to-left, then in previous sentences from left-to-right (finally, the current sentences is searched left-to-right for cataphora). The first noun phrase that agrees with the pronoun with respect to number, gender, and person is chosen as the antecedent (Kehler et al., 2004). While the lack of constituency parse prevents the application of this algorithm, we estimate the order in which A/B would have been found by the Hobbs algorithm by counting the number of noun phrases between A/B and the pronoun.

3.1.4 Features Based On Binding Theory

In linguistics, binding is the distribution of anaphoric elements. A pronoun usually has an antecedent (a "binder") in context. The goal of binding theory is to identify the syntactic relationship that can or must hold between a given pronoun or noun and its antecedent (or postcedent). The binding domain is where a reflexive or reciprocal pronoun should find its antecedent. (Carnie, 2012)

As shown in Table 1: (1) With Condition A, if an anaphor is bound with a proper name in a clause, they must corefer; (2) With Condition B, if a pronominal is bound with a proper name in a clause, they cannot corefer. Using these two rules, we generated features `A/B bound with Pronoun` (*'aim-bound'*) and `A/B bound with an anaphor` (*'anaphor-bound'*). If *aim-bound* or *anaphor-bound* is true, the proper name and the pronoun is bound in the same clause.

It should be noted that, for this study, we did not have a constituent structure parsing result, and it can be hard to define the clause boundary. The solution we took for clause boundary definition was to check whether the pronoun and the proper name's distances from the same ROOT in the dependency tree are both smaller than 3. If so, it is very likely that they are from the same clause.

3.1.5 Other Features

Features `Punctuation numbers in between` (*'punc'*) and `Mentioned times` (*'mention'*) are generated based on the tokenization result of spaCy. The *'punc'* feature reflects an estimate of the number of clauses between the proper name A/B and the pronoun, providing a

	Anaphors	Pronominals	R-expressions
Example	himself, herself	he/she/his/her	James/the man
Principles	Need a c-commanding coreference NP in their clause. Need to be bound in their clause.	Cannot have a c-commanding coreference NP in their clause. Cannot be bound, must be free in their clause.	Cannot have a c-commanding coreference NP. Cannot be bound, must be free.
Binding domain	NP/clause	NP/clause	No binding domain
Condition Name	Condition A	Condition B	Condition C

Table 1: Conditions in Binding Theory: With the restrictions of Binding Theory, an anaphor must have its coreferred proper name in the same clause, and a pronomial cannot have its coreferred proper name in the same clause.

measure of "syntactic distance" in combination with the linear distance provided by the offset. The *'mention'* feature indicates whether the proper name A/B is a "key word" in this paragraph, which is more likely to be referred to by a pronoun.

3.2 Machine Learning Algorithms

Machine learning algorithms are used for coreference relationship (represented by *'A-coref'* and *'B-coref'*) prediction based on the other features (*'offset', 'nsubj', 'NPs', 'punc', 'mention', 'aim-bound', 'depth', 'anaphor-bound'*). To test each feature's contribution, the *'offset'* feature is used as a baseline, and all other features are combined with it (represented as '+ *feature*' along the x-axis in the result Figure 3, 4, 5).

The algorithms include: SVM (support vector machine), decision tree, random forest, logistic regression, multi-layer perceptron. The training and testing data are described in Section 4.

4 Results

4.1 Main Results for GAP

Using the features described in Table 2, classifiers (shown on the y-axis in Figure 3) are trained with 80% of the GAP data (3562 rows), and tested with 20% of the GAP data (891 rows). *'offset'* is the baseline for feature performance, and the other features are added separately to *'offset'* to verify their contribution degree. Feature *'ALL'* uses all the features we generated to train and test the model.

Figure 3 shows the accuracy when [A-coref, B-coref] are used as labels. There are three possible outcomes: [1, 0] (proper name A is coreferred with the pronoun), [0, 1] (proper name B is coreferred with the pronoun), [0, 0] (neither proper name A nor B is coreferred with the pronoun).

Figure 4 shows the accuracy when *'A-coref'* is used as the label. There are two possible outcomes: [1] (proper name A is coreferred with the pronoun), [0] (proper name A is not coreferred with the pronoun).

Inferring from the results in Figure 3 and Figure 4, the feature combinations *"offset + depth", "offset + nsubj"* have the best performance for coreference prediction.

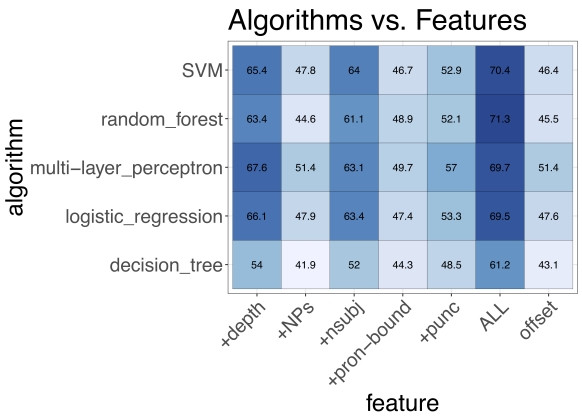

Figure 3: Accuracy(%) when different features(x-axis) are used to predict *'A-coref', 'B-coref'* with machine learning algorithms(y-axis). Training models: 80% of GAP data; Testing models: 20% of GAP data.

4.2 Results for Penn Treebank Data

To test the model's performance on a different text resource, 90 hand-annotated data from the WSJ in

Feature	Description	Example
`Coreference Label` (*'coref'*)	This binary feature provided in the GAP dataset indicates whether A/B and Pronoun corefer.	A-coref: 1 (True) B-coref: 0 (False)
`Offset Position` (*'offset'*)	This numerical feature is the offset position of A/B/Pronoun in the whole paragraph. The character offset is provided in the GAP dataset, and the token offset is generated in this project.	A-offset-char:234 B-offset-char:255 Pronoun-offset-char:282 A-offset-token:45 B-offset-token:49 Pronoun-offset-token:53
`Is Subject` (*'nsubj'*)	This binary feature is obtained from the dependency parsing result with spaCy, and indicates whether A/B/Pronoun is acting as a subject in its sentence.	A-nsubj:1 B-nsubj:1 Pronoun-nsubj:0
`Noun Phrase numbers in between` (*'NPs'*)	This numerical feature is the number of noun phrases between A/B and Pronoun in a paragraph, inspired by the Hobbs Algorithm.	nouns-A-Pronoun:1 nouns-B-Pronoun:0
`Punctuation numbers in between` (*'punc'*)	This numerical feature is the number of commas/periods between A/B and Pronoun in a paragraph.	commas-A-Pron:1 commas-B-Pron:0 period-A-Pron:0 period-B-Pron:0
`Mentioned times` (*'mention'*)	This numerical feature is the number of times A/B/Pronoun is mentioned in a paragraph.	A-Mentions:5 B-Mentions:2
`A/B bound with Pronoun` (*'aim-bound'*)	This binary feature is inspired by Binding Theory, and supported by the dependency parsing result using spaCy. A/B and Pronoun are bound if both of them are headed by the same verb ROOT and distance from the ROOT is smaller than 3.	aim-bound-A:0 aim-bound-B:1
`Node depth in dependency tree` (*'depth'*)	This numerical feature indicates A/B/Pronoun's depth in the dependency tree, i.e. depth from the main verb.	A-depth:2 B-depth:1 Pronoun-depth:2
`A/B bound with an anaphor` (*'anaphor-bound'*)	This binary feature is motivated by Binding Theory as well. If an entity is bound with an anaphor(himself/herself), its gender is clear and can be used to exclude the other gender pronouns' co-indexation .	anaphor-bound-A:0 anaphor-bound-B:0

Table 2: Feature Descriptions

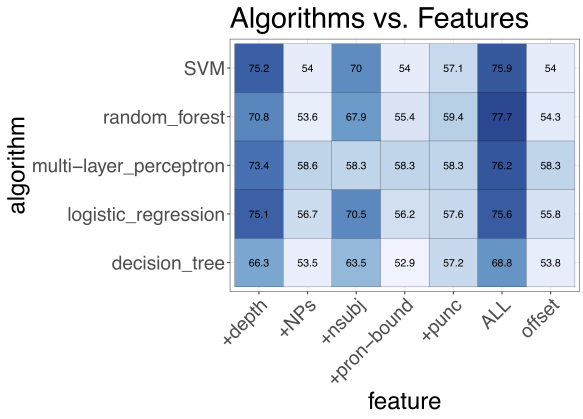

Figure 4: Accuracy(%) when different features(x-axis) are used to predict *'A-coref'* with machine learning algorithms(y-axis). Training models: 80% of GAP data; Testing models: 20% of GAP data.

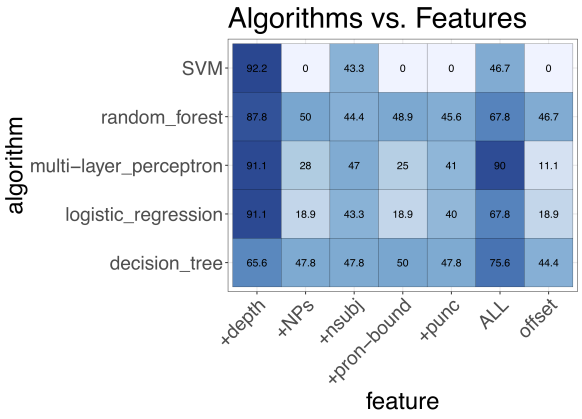

Figure 5: Accuracy(%) when different features(x-axis) are used to predict *'A-coref'* with machine learning algorithms(y-axis), using only A and Pronoun related features. Training models: 100% of GAP data; Testing models: 90 annotated sentences from Penn Treebank.

the Penn Treebank (Taylor et al., 2003) were used.

To obtain test data from the Penn Treebank, about 250 sentences containing the same third person pronoun as GAP were selected using regular expressions, which were also used to clean the text for feature generation using spaCy. The proper name which coreferred with each pronoun was hand-annotated in 90 instances.

The classifier was trained using all 4453 rows in the GAP dataset, and *'A-coref'* was used as label, and all pronoun-B-related features were excluded for this stage. As shown in Figure 5, the feature combination *"offset + depth"* has the best performance, and *"offset + nsubj"* gets higher accuracy than *"offset"* alone. Among all machine learning algorithms, multi-layer perceptron is the most robust classifier, and gets 90% accuracy while trained and tested with all features.

In addition, we obtained 1,090 pronoun antecedent pairs from WSJ text, automatically annotated in the BLLIP corpus (Charniak et al., 2000). When evaluating on these paragraphs, we observed the same trends for feature combinations and classifiers as on the 90 hand-annotated pairs, with accuracy generally improving a few percentage points, to a high value of 95.3% accuracy for SVM using *"offset + depth"* features.

As a comparison, the A-related and Pronoun-related features were also used to train a classifier using 80% of the GAP data, and testing was applied on 20% of the GAP data. The result is shown in Figure 6.

5 Discussion

In this study, features for coreference prediction relied heavily on tokenization and dependency parsing results from spaCy. However, we cannot guarantee that spaCy provides 100% accurate parsing results. For example, the 'depth' feature is the entities' depth in the dependency tree, and an inaccurate depth value might cause a wrong prediction.

At the beginning of the project, the features generated with binding theory were expected to be a powerful predictor. However, when they were applied on the GAP data, less than 10% of the rows were found to have a binding relationship between a proper name and a pronoun. It is obvious that these binding theory features will not play an essential role for prediction. However, we would argue that this is related to the nature of GAP data, since most entities are not labeled in the same sentence, let alone the same CP. With another dataset, we might encounter more circumstances with a proper name and a pronoun in the same CP so that binding theory related features could be more helpful. In contrast, the 'depth' feature was much more valuable than predicted, especially with the Penn Treebank data.

6 Related Work

When Webster et al. released the GAP dataset in 2018, they included benchmark results for the coreference task (Webster et al., 2018). The strongest "off the shelf system", developed by Lee

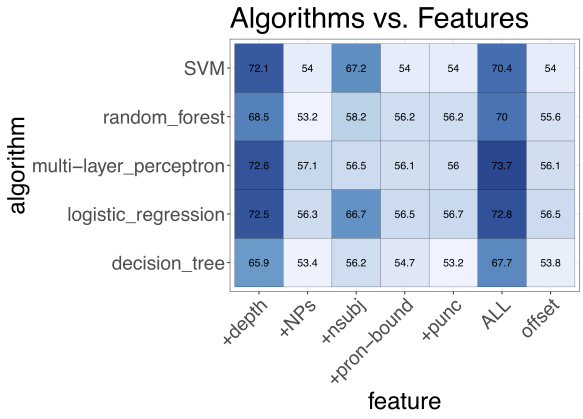

Figure 6: Accuracy(%) when different features(x-axis) are used to predict *'A-coref'* with machine learning algorithms(y-axis), using only A and Pronoun related features. Training models: 80% of GAP; Testing models: 20% of GAP.

et al. had an accuracy of 64.0% on the development section of the dataset (Lee et al., 2017). Webster et al. also developed a "Parallelism" method with an accuracy of 66.9%, which was the highest accuracy achieved without using extra context from the URL of the Wikipedia page. This "Parallelism" baseline prefers the possible antecedent with the same part of speech as the pronoun, and backs off to closest syntactic distance when this is not found.

State of the art results on this dataset were achieved using contextual word embeddings from BERT as the input to a neural network (Joshi et al., 2019), (Yang et al., 2019), but contextual word embeddings have not been included as features in our exploration. Instead, we focus on generating linguistically-informed features. This approach is similar to that of Ge et al., who used the WSJ Penn Treebank and included features based on the hand-annotated constituency trees (Ge et al., 1998). In contrast, we rely only on the machine-generated dependency parse to generate our trees. Another option is to use machine-generated constituency trees to generate features based on binding theory (Luo and Zitouni, 2005).

Ge et al. also use unsupervised learning to acquire gender information for noun phrases (Ge et al., 1998). Similarly, Bergsma et al. who introduce the use of a "dependency path" between two entities find that gender is the most "powerful probabilistic feature" in their model (Bergsma and Lin, 2006). Our model differs in taking a gender-agnostic approach, motivated by the structure of the GAP dataset which hopes to "close the gap" between performance between male and female coreference scores.

7 Conclusion

Based on the results, *'depth'* (an entity's depth in its sentence's dependency tree) is the most robust feature to predict the coreference relationship between a proper noun and a pronoun, and it has good performance when applied on a different text resource data.

When the model is tested and trained both with GAP data, the accuracy results are consistent for all classifiers: SVM, decision tree, random forest, logistic regression, and multi-layer perceptron. However, when the model is trained with GAP and tested with Penn Treebank data, only decision tree and random forest have stable performance. But when an appropriate shape is chosen for the multi-layer perceptron, it can also have good performance on *"offset + depth"* and *'ALL'* features.

Future work could combine some of the features generated in this project with contextual word embeddings, and POS (Part Of Speech) taggers can be included for embeddings as well. Although we explored using word vectors based on the language model from spaCy, using the cosine similarity between A/B and Pronoun as a feature had no effect on accuracy. We expect that incorporating contextual word embeddings from BERT will significantly improve accuracy in combination with our syntactic features.

References

Shane Bergsma and Dekang Lin. 2006. Bootstrapping path-based pronoun resolution. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 33–40. Association for Computational Linguistics.

Andrew Carnie. 2012. *Syntax: A generative introduction*, volume 16. John Wiley & Sons.

Eugene Charniak, Don Blaheta, Niyu Ge, Keith Hall, and Mark Johnson. 2000. Bllip 1987–89 wsj corpus release 1, ldc no. *LDC2000T43. Linguistic Data Consortium*.

Dennis Connolly, John D Burger, and S David. Day. 1994. In *A machine learning approach to anaphoric reference. In Proceedings of the International Conference on New Methods in Language Processing (NEMLAP)*.

Ido Dagan and Alon Itai. 1990. Automatic processing of large corpora for the resolution of anaphora references. In *Proceedings of the 13th conference on Computational linguistics-Volume 3*, pages 330–332. Association for Computational Linguistics.

Niyu Ge, John Hale, and Eugene Charniak. 1998. A statistical approach to anaphora resolution. In *Sixth Workshop on Very Large Corpora*.

Jerry R Hobbs and Stuart M Shieber. 1987. An algorithm for generating quantifier scopings. *Computational Linguistics*, 13(1-2):47–63.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7.

Mandar Joshi, Omer Levy, Daniel S Weld, and Luke Zettlemoyer. 2019. Bert for coreference resolution: Baselines and analysis. *arXiv preprint arXiv:1908.09091*.

Daniel Jurafsky and James H. Martin. 2019. *Speech and Language Processing*.

Andrew Kehler, Douglas Appelt, Lara Taylor, and Aleksandr Simma. 2004. The (non) utility of predicate-argument frequencies for pronoun interpretation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 289–296.

Christopher Kennedy and Branimir Boguraev. 1996. Anaphora for everyone: pronominal anaphora resoluation without a parser. In *Proceedings of the 16th conference on Computational linguistics-Volume 1*, pages 113–118. Association for Computational Linguistics.

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

Xiaoqiang Luo and Imed Zitouni. 2005. Multi-lingual coreference resolution with syntactic features. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 660–667. Association for Computational Linguistics.

Ruslan Mitkov. 1995. *Anaphora resolution in natural language processing and machine translation*. Institut der Gesellschaft zur Förderung der Angewandten Informationsforschung.

Ruslan Mitkov. 1999. *Anaphora resolution: the state of the art*. Citeseer.

Susanne Preuss. 1992. *Anaphora resolution in machine translation*. TU, Fachbereich 20, Projektgruppe KIT.

Ann Taylor, Mitchell Marcus, and Beatrice Santorini. 2003. The penn treebank: an overview. In *Treebanks*, pages 5–22. Springer.

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.

Kai-Chou Yang, Timothy Niven, Tzu Hsuan Chou, and Hung-Yu Kao. 2019. Fill the gap: Exploiting bert for pronoun resolution. In *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, pages 102–106.