Improving Generalization in Coreference Resolution via Adversarial Training

Sanjay Subramanian
Advisor: Prof. Dan Roth

Abstract

Coreference resolution, the task of finding and clustering mentions of entities (e.g. people, locations, and organizations) in text, is important for several applications in natural language processing (NLP), such as summarization of news articles and automated reading comprehension of stories. In order for coreference resolution systems to be useful in practice, they must be able to generalize to (i.e. work well on) new text. In particular, it is important that systems work well with unfamiliar names and with styles of text that differ from those seen in training. In this work, we demonstrate that the performance of the state-of-the-art system decreases when the names of person (PER) and geopolitical location (GPE) named entities in the CoNLL dataset are changed to names that do not occur in the training set. Then we use the technique of adversarial gradient-based training to retrain the state-of-the-art system and demonstrate that the retrained system achieves higher performance on the CoNLL dataset (both with and without the change of named entities) and the recently introduced GAP dataset, which balances the occurrences of male and female names.

Motivation

Entities play a central role in most meaningful texts. In a story, the characters, their development, and their involvement in events are perhaps the most important aspects. When a student or social scientist is gathering information from articles in the news or scholarly journals, he or she is often interested in a single or small set of people or organizations. Hence, the ability to recognize references to entities is important for automated systems designed for natural language processing (NLP). Beyond simply recognizing such mentions of entities, a reader should also understand which mentions refer to the same entity. Only then can the reader piece together different parts of the text to synthesize a complete understanding of the story or exposition. Determining whether mentions refer to the same entity is complicated by the variety of ways in which text can be used to refer to a single entity. In particular, names, pronouns, and nominal phrases (phrases that are neither names nor pronouns) can all be used to refer to the same entity. Therefore, this task of finding and clustering mentions of entities in a document, which is called *coreference resolution*, is not only important in the world of NLP, but is also difficult.

Since it has intuitive importance of entities in text as argued above, coreference resolution has been shown to be useful for several applications in NLP, such as question answering (Dhingra et al., 2018), reading comprehension (Wang et al., 2016), summarization (Steinberger et al., 2016), and sentiment analysis (Sukthanker et al., 2018). In accordance with the utility of coreference resolution, NLP researchers have invested a great deal of effort into developing and improving systems for the task. Through the use of neural networks, performance on the task of coreference resolution has increased significantly over the last few years. Still, neural systems trained on the standard coreference dataset have issues with generalization, as shown by Moosavi and Strube (2018).

One way to improve the understanding of how a system overfits a dataset is to study the change in the system's performance when the dataset is modified slightly in a focused and relevant manner. We take this approach by modifying the test set so that each PER and GPE (person and geopolitical entity) named entity is different from those seen in training. In other words, we ensure that there is no leakage of PER and GPE named entities from the training set into the test set. We demonstrate that the performance of the Lee et al. (2018) system, which is the current state-of-the-art, decreases when the named entities are replaced. An example of a replacement that causes the system to make an error is given in Table 1.

Motivated by these issues of generalization, this paper aims to improve the training process of neural coreference systems. Various regularization techniques have been proposed for improving the generalization capability of neural networks, including dropout (Srivastava et al., 2014) and adversarial training (Goodfellow et al., 2015; Miyato et al., 2017). The model of Lee et al. (2018), like most neural approaches, uses dropout. In this work, we apply the adversarial fast-gradient-sign-method (FGSM) described by Miyato et al. (2017) to the model of Lee et al. (2018), and show that this technique improves the model's generalization even when applied on top of dropout.

We demonstrate that the system of Lee et al. (2018) retrained with adversarial training achieves state-of-the-art perfor-

Original: But **Dirk Van Dongen , president of the National Association of Wholesaler - Distributors** , said that last month 's rise " is n't as bad an omen " as the 0.9 % figure suggests . " If you examine the data carefully , the increase is concentrated in energy and motor vehicle prices , rather than being a broad - based advance in the prices of consumer and industrial goods , " **he** explained .
Replacement: Replace *Dick Van Dongen* with *Vendemiaire Van Korewdit*.

Table 1: An excerpt from the CoNLL test set. The coreference between the two highlighted mentions is correctly predicted by the Lee et al. (2018) system, but after the specified replacement, the system incorrectly resolves "he" to a different name occurring outside this excerpt.

mance on the original CoNLL-2012 dataset (Pradhan et al., 2012) as well as the CoNLL-2012 dataset with changed named entities. Furthermore, the system trained with the adversarial method exhibits state-of-the-art performance on the GAP dataset (Webster et al., 2018), a recently released dataset focusing on resolving pronouns to people's names in excerpts from Wikipedia.[1]

Related Work

Moosavi and Strube (2017, 2018) also study generalization of neural coreference resolvers. However, they focus on transfer and indicate that the ranking of coreference resolvers (trained on the CoNLL training set) induced by their performance on the CoNLL test set is not preserved when the systems are evaluated on a different dataset. They use the Wikicoref dataset (Ghaddar and Langlais, 2016), which is limited in that it consists of only 30 documents.

They then show that the addition of features representing linguistic information improves the performance of a coreference resolver on the out-of-domain dataset.

The adversarial fast-gradient-sign-method (FGSM) was first introduced by Goodfellow et al. (2015) and was applied to sentence classification tasks through word embeddings by Miyato et al. (2017). Gradient-based adversarial attacks have since been used to train models for various NLP tasks, such as relation extraction (Wu et al., 2017) and joint entity and relation extraction Bekoulis et al. (2018).

Our replacements of named entities can also be viewed as a way of generating adversarial examples for coreference systems; it is related to the earlier method proposed in Khashabi et al. (2016) in the context of question answering and to Alzantot et al. (2018), which provides a way of generating adversarial examples for simple classification tasks.

Technical Approach

Adversarial Training for Coreference

In coreference resolution, the goal is to find and cluster phrases that refer to entities. We use the word "span" to mean a series of consecutive words. A span that refers to an entity is called a mention. If two mentions i and j refer to the same entity and mention i occurs before mention j in the text, we say that mention i is an antecedent of mention j. For a given mention i, the candidate antecedents of i are the mentions that occur before i in the text. In Figure 1, each line segment represents a mention and the arrows are directed from one mention to its possible antecedents.

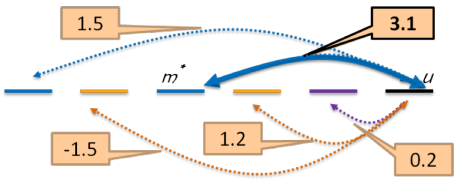

Figure 1: For each mention, the model computes scores for each of the candidate antecedent mentions and chooses the candidate with the highest score to be the predicted antecedent. This image was created by the authors of (Chang et al., 2013).

We now review the model architecture of Lee et al. (2018) and describe how we apply the fast-gradient-sign-method (FGSM) of Miyato et al. (2017) to the model. Using GloVe (Pennington et al., 2014) and ELMo Peters et al. (2018) embeddings of each word and using learned character embeddings, the model computes contextualized representations $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$ of each word x_i in the input document using a bidirectional LSTM Hochreiter and Schmidhuber (1997). For candidate span i, which consists of the words at indices $start_i, start_i + 1, ..., end_i$, the model constructs a span representation $\mathbf{g}_i$ by concatenating $\mathbf{x}_{start_i}$, $\mathbf{x}_{end_i}$, $\frac{1}{\sum_{j=start_i}^{end_i} \beta_j} \sum_{j=start_i}^{end_i} \beta_j \mathbf{x}_j$, and $\phi(end_i - start_i)$, where the β_j's are learned scalar values and $\phi(\cdot)$ is a learned embedding representing the width of the span (Lee et al., 2017). The span representations are then used as inputs to feedforward networks that compute mention scores for each span and that compute antecedent scores for pairs of spans. In Figure 1, the number associated with each arrow is the antecedent score for the associated pair of mentions. The coreference score for the pair of spans (i, j) is the sum of the mention score for span i, the mention score for span j, and the antecedent score for (i, j). For each span i, the antecedent span predicted by the model is the span j that maximizes the antecedent score for (i, j).

Let $\mathbf{g} = \{\mathbf{g}_i\}_{i=1}^N$ denote the set of the representations of all N candidate spans. Let $\mathcal{L}(\mathbf{g})$ denote the original model's loss function. (Note that the model's predictions and the loss depend on the input text only through the span representations.) For each $i \in \{1, ..., N\}$, let $\mathbf{g}_i^{adv}(\mathbf{g}) = \nabla_{\mathbf{g}_i} \mathcal{L}\left(\{\mathbf{g}_i\}_{i=1}^N\right)$ denote the gradient of the loss with respect to the span embeddings. Then the adversarial loss with the FGSM is

$$\mathcal{L}_{adv}(\{\mathbf{g}_i\}_{i=1}^N) = \mathcal{L}\left(\left\{\mathbf{g}_i + \epsilon \frac{\mathbf{g}_i^{adv}(\mathbf{g})}{||\mathbf{g}_i^{adv}(\mathbf{g})||}\right\}_{i=1}^N\right).$$

[1] A significant portion of this report is taken from a recently accepted publication based on the same work (Subramanian and Roth, 2019).

The total loss used in training is

$$\mathcal{L}_{total}(\mathbf{g}) = \alpha\mathcal{L}(\mathbf{g}) + (1-\alpha)\mathcal{L}_{adv}(\mathbf{g}).$$

In our experiments, we find that $\alpha = 0.6$ and $\epsilon = 1$ work well. A key difference between our method and that employed by Miyato et al. (2017) is that the latter applies the adversarial perturbation to the input embeddings, whereas we apply it to the span representations, which are an intermediate layer of the model. We found in our experiments that applying the FGSM to the character embeddings in the initial layer was not as effective as applying the method to the span representations as described above. Another difference between our method and that of Miyato et al. (2017) is that we do not normalize the span embeddings before applying the adversarial perturbations.

No Leakage of Named Entities

Named entities are an important subset of the entities a coreference system is tasked with discovering. Agarwal et al. (2018) provide the percentages of clusters in the CoNLL dataset represented by the PER (person), ORG (organization), GPE (geopolitical entity), and DATE (date) named entity types – 15%, 11%, 11%, and 4%, respectively. It is important for generalization that systems perform well with names that are different from those seen in training. We found that in the CoNLL dataset, roughly 34% of the PER and GPE named entities that are the head of a mention of some gold cluster in the test set are also the head of a mention of a gold cluster in the train set. Therefore, there is considerable overlap, or leakage, between the names in the train and test sets. In this section, we describe a method for evaluating on the CoNLL test set without leaked name entities.

We focus on PER and GPE named entities because they are two of the three most common entity types and because in general when replacing a PER or GPE name with another name, it is easy to not change the true coreference structure of the document. In particular, changing the name of an organization while ensuring that it is compatible with nominals in the cluster is nontrivial without a finer semantic typing. By contrast, we describe below how we control for gender and location type when replacing PER and GPE names, respectively. We also ensure that the capitalization of the first letter in the replacement name is the same as in the original text. Finally, we note that the diversity of PER and GPE entities exceeds that of other named entity types; this increases the importance of generalization to new names and, at the same time, enables us to find matching names to use as replacements.

Replacing PER entities For replacing PER entities, we utilize the publicly available list of last names from the 1990 U.S. Census and a gazetteer of first names that has the proportion of people with this name who are males. The gazetteer was collected in an unsupervised fashion from Wikipedia. We denote the list of last names by $\mathcal{L}$, the list of male first names (i.e. first names with male proportion greater than or equal to 0.5 in the gazetteer) by $\mathcal{M}$, and the list of female first names (i.e. first names with male proportion less than or equal to 0.5 in the gazetteer) by $\mathcal{F}$. We remove all names occurring in training from $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{F}$. We use the spaCy dependency parser (Honnibal and Johnson, 2015) to find the heads of each mention. We say that a mention is a person-mention if the head of the mention is a PER named entity, and we say that the name of the person-mention is the PER named entity that is its head. We use the dependency parser and the gold NER to identify all of the person-mentions. For each gold cluster containing a person-mention, we find the longest name among the names of all of the person-mentions in the cluster. If the longest name of a cluster has only one token, we assume that the name is a last name, and we replace the name with a name chosen uniformly at random from the remaining last names in $\mathcal{L}$. Otherwise, if the longest name has multiple tokens, we say that the cluster is male if the cluster contains no female pronouns ("she", "her", "hers") and one of the following is true: the first token does not appear in $\mathcal{M}$ or $\mathcal{F}$, if the token appears in $\mathcal{M}$, or the cluster contains a male pronoun ("he", "him", "his"). We say that the cluster is female if it is not male. Then we (1) replace the last token with a name chosen uniformly at random from the remaining last names in $\mathcal{L}$, and (2) replace the first token with a name chosen uniformly at random from the remaining first names in $\mathcal{M}$ if the cluster is male or from the remaining first names $\mathcal{F}$ if the cluster is female. Note that our sampling from each of $\mathcal{L}$, $\mathcal{M}$, and $\mathcal{F}$ is without replacement, so no last name is used as a replacement more than once, no male first name is used more than once, and no female first name is used more than once.

Replacing GPE entities Our approach to replacing GPE entity names is very similar to that used for PER names. We use the GeoNames[2] database of geopolitical names. In addition to providing a list of GPE names, this database also categorizes the names by the type of entity to which they refer (e.g. city, state, county, etc.). The data includes the names and categories of more than $11,000,000$ locations in the world. We restrict our attention to GPE entities that satisfy the following requirements: (1) they occur in the GeoNames database and (2) they are not countries. We say that a mention is a GPE-mention if its head (as given by the dependency parser) is a GPE named entity satisfying these three requirements. (Again, we use the gold NER to identify GPE names in the CoNLL text.) We remove all GPE names occurring in the training set from the list of replacement GPE names for each location category. Then for each cluster containing a GPE-mention, we find the GeoNames category for the mention's GPE name and replace the name with a randomly chosen name from the same category. As with PER names, we sample names from each category without replacement, so each GPE name is used for replacement at most once.

Evaluation

In this section, we study both how our No Leakage modifications affect the performance of the state-of-the-art system and how adversarial training improves the generalization of the state-of-the-art system. For a qualitative understanding

[2]http://www.geonames.org/

of the No Leakage modifications, we provide in Table 2 examples of text in the original CoNLL-2012 dataset and the corresponding text after our modifications. In total 81 GPE entities and 560 PER entities in the test set were modified by our process; the total number of gold entities in the test set is 4532. We trained the Lee et al. (2018) model architecture with the adversarial approach on the CoNLL training set for 355000 iterations (the same number of iterations for which the original model was trained) with the same training hyperparameters used by original model. For comparing with the Lee et al. (2017) and Lee et al. (2018) systems, we use the pretrained models released by the authors.[3] The datasets used for evaluation are the CoNLL and GAP datasets.

CoNLL-2012 Dataset

The CoNLL-2012 Shared Task dataset (Pradhan et al., 2012) has been the standard dataset used for both training and evaluating English coreference systems since the dataset was introduced. The dataset includes seven genres that span multiple writing styles and multiple nationalities. Performance on the CoNLL-2012 dataset is traditionally measured via CoNLL F1, which is the arithmetic mean of three F1 metrics – MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), and $CEAF_e$ (Luo, 2005).

Moosavi and Strube (2016) provides a detailed description of each of these three metrics; we provide a brief overview here. In each case, we describe what the recall of the metric means; the precision is conceived by switching the role of the gold and predicted entities in the recall computation. For each gold entity, the penalty assigned by MUC recall to the predicted entity set is commensurate with the number of predicted entities that have non-zero intersection with the gold entity. B^3 recall measures for each gold entity the proportion of mentions in the entity that is captured by each predicted entity. For $CEAF_e$, a one-to-one mapping between the gold and predicted entities is computed based on a pairwise similarity function between two entities (one gold and one predicted) that is based on the fraction of mentions in the two entities that belong to both entities. Given this mapping, the $CEAF_e$ recall is given by the average similarity value across all gold entities.

GAP Dataset

The GAP dataset (Webster et al., 2018), recently introduced by Google, focuses on resolving pronouns to named people in excerpts from Wikipedia. The dataset, which is gender-balanced, consists of examples in which the system must determine whether a given pronoun refers to one, both, or neither of two given names. Thus, the task can be viewed a binary classification task in which the input is a (pronoun, name) pair and the output is True if the pair is coreferent and False otherwise. Performance is evaluated using the F1 score in this binary classification setup. Tables 3 and 4 display the results on these two datasets.[4]

[3] Available at https://lil.cs.washington.edu/coref/final.tgz and http://lsz-gpu-01.cs.washington.edu/resources/coref/c2f_final.tgz

[4] The results that we report for the Lee et al. (2017) system differ slightly from those reported in Table 10 of Webster et al. (2018) due to a difference in the parser and potentially small differences in the algorithm for converting the system's output to the binary predictions necessary for the GAP scorer.

Discussion of Findings

The modifications made by our No Leakage procedure are qualitatively reasonable and seem to achieve the goal of swapping names while retaining the coreference structure of the document. With respect to the total number of entities in the test set (4532), the number of replacements for PER entities (560) represents about 12%, which is relatively close to the overall proportion of entities that are of the PER type. On the other hand, the number of replacements for GPE entities (81) represents only about 1.8% of all entities, which is far lower than expected. This lower proportion can be attributed to (1) the fact that many GPE entities are countries (which are excluded from our procedure) and (2) searching within the GeoNames database, in which the primary name of an entity is frequently different from the name used to reference the entity in texts.

The drop in performance between the original and No Leakage CoNLL-2012 test sets shows that the state-of-the-art system has a generalization issue with respect to unseen names. With adversarial training, the system is able to recover about 40% of the performance drop in the No Leakage setting. Moreover, adversarial training also increases performance on the original CoNLL test set and improves transfer from the CoNLL-2012 training set to the GAP test set, providing further evidence for the view of adversarial training as a regularization technique.

Ethical Considerations and Societal Impact

Given that this project is centered on a research problem rather than a product, the societal impact is highly dependent on how the methods used in this study are applied to software used at scale by the companies and the public. However, this project also addresses a specific ethical issue that has garnered interest within the NLP community, namely the bias of coreference systems. **?** and **?** both study the gender bias of coreference resolution systems. Instead, we study the bias of coreference systems with respect to names seen in training; in particular, our method of replacing names utilizes names from many ethnicities and world regions and can help assess whether a system is robust to non-Western names. Moreover, our work extends the study of bias beyond people's names to names of locations (geopolitical entities). Therefore, our work is tied to ethical concerns and improves understanding with respect to such issues by introducing a method (controlled replacement of names) of evaluating a certain kind of bias (favoring familiar names) and a method that can help to decrease this bias (adversarial training).

We also accounted for gender bias in our study by including results on the GAP dataset. This dataset was motivated by the goal of evaluating gender bias. One specific observation regarding our work is that the increase in performance on the GAP dataset due to adversarial training does not come at the expense of additional gender bias. Using the values from Table 4, we find that the ratio of female performance to male performance is approximately 0.931 for the Lee et al. (2018)

Original	No Leakage
We asked **Judy Muller** if she would like to do the story of a fascinating man . She took a deep breath and said , okay .	We asked **Sallie Kousonsavath** if she would like to do the story of a fascinating man . She took a deep breath and said , okay .
The last thing President **Clinton** did today before heading to the Mideast is go to church – appropriate , perhaps , given the enormity of the task he and his national security team face in the days ahead .	The last thing President **Golia** did today before heading to the Mideast is go to church – appropriate , perhaps , given the enormity of the task he and his national security team face in the days ahead .
In theory at least , tight supplies next spring could leave the wheat futures market susceptible to a supply - demand squeeze , said Daniel Basse , a futures analyst with AgResource Co. in **Chicago** .	In theory at least , tight supplies next spring could leave the wheat futures market susceptible to a supply - demand squeeze , said Daniel Basse , a futures analyst with AgResource Co. in **Machete** .

Table 2: Excerpts from the CoNLL-2012 test set and their versions after we have replaced PER and GPE names to avoid name leakage.

	Original	**No Leakage**
Lee et al. (2018)	72.96	71.86
+Adv. Training	**73.23**	**72.36**

Table 3: Results (CoNLL F1) on the CoNLL Test Set. "Original" refers to the original test set, and "No Leakage" refers to the test set modified with the replacement of named entities described in Section . For each dataset, highest score for each dataset is **bolded** and is underlined if the difference between it and the other model's score is statistically significant ($p < 0.20$ per a stratified approximate randomization test similar to that of Noreen (1989)).

	M	**F**	**O**
Lee et al. (2017)	68.7	60.0	64.5
Lee et al. (2018)	75.8	70.6	73.3
+Adv. Training	**77.3**	**72.1**	**74.7**

Table 4: Results (F1 metric defined by Webster et al. (2018)) on the GAP Test Set. **M** refers to male pronouns, **F** refers to female pronouns, and **O** refers to the full evaluation data. For each category, highest score is **bolded** and is underlined if difference between it and next-highest score is statistically significant ($p < 0.05$ per the McNemar test (McNemar, 1947)).

system and approximately 0.933 for the adversarially trained system.

Business Analysis

Although this project is motivated by research questions, the methods and knowledge at the core of the project have strong potential for forming the basis of a company. We propose to commercialize this research by developing two core products: an API for coreference resolution in documents and an API for automatic summarization of documents.

Overview of the need, product, and targeted customers

Currently, news websites primarily rely on a simple keyword search to identify relevant documents. Some websites also offer an "Advanced Search" feature that allows users to manually specify information in multiple categories (e.g. author, title). While a simple keyword search is usually enough to identify which documents are relevant to the user's interest, it would be useful, especially for long documents, if the website could also identify the individual sentences that are relevant to the user's interest. For instance, if the user is searching for information about President Obama in a long news article, it would be useful to the user to know which sentences mention President Obama (though perhaps not by name) without reading the full document. Our **value proposition** is enabling users to synthesize information from news documents quicker and easier through the use of automated NLP tools. The **stakeholders** for our products include publishers of online news articles, other websites that a vast collection of documents that is searched by users, and consumers of online news and other documents.

The proposed coreference resolution API is meant to address this issue when the user searches for a named person, place, or organization. When the user enters a query in a search box, the coreference API would be called on the top document or top few documents in the search results. It is worthy of note that coreference resolution performance on English news documents has become quite strong in the last year and would provide real value to users through this search process. A secondary customer segment for the coreference API is quantitative trading firms, which could use coreference to identify sentences involving companies of interest and proceed to use the information or sentiment of these sentences to inform their trades.

While the development of a coreference resolution API could be initiated immediately, a summarization API would not make sense without further advancement of research in automated summarization. As mentioned earlier, coreference resolution is useful for summarization, so we could leverage knowledge of the former to improve summarization performance. The target customers for the summarization API would also be news websites, as summarization is

another step toward streamlining the user's process of synthesizing an article's information. Since summarization is a difficult task, a strong summarization system would differentiate us from competitors, allowing us to set a high price point and derive greater profit. Thus, together, our portfolio of the coreference API and the summarization API includes both a product that can be monetized quickly and a long-term investment with the potential for high payoff.

Competitive Analysis

Our immediate competitors are companies offering a coreference resolution or summarization API, and more broadly any company concerned with pursuing NLP research and translating it into products would be our competitor. Two companies that offer a coreference resolution API or a similar product are Algorithmia (Algorithmia) and Google (through its Cloud NL service, it offers custom entity extraction – Google Cloud (a)). In addition, Plasticity AI is planning to release a coreference API soon (Plasticity AI, a). One way in which we will differentiate ourselves is the performance of our system. The Algorithmia API is based on the Stanford deterministic coreference resolution system (Recasens et al., 2013), whose performance is lower than that of our system by more than 10 F1 points on the CoNLL-2012 dataset. Google's API seems to focus on entity extraction and not on coreference, so the API will not output pronouns that refer to a name of interest. However, given that Plasticity AI and Google will likely release high-performing coreference APIs in the near future, we will take steps to further differentiate our product by tailoring it to the news industry. One straightforward way to tailor our coreference API would be to use word representations that are pre-trained only on news text rather than on other kinds of text; doing so could improve the system's performance within the news domain. More importantly, we will work to make sure to design our API in a manner that makes integration with the search engines of news websites as streamlined as possible.
There are existing products for summarization, such as Algorithmia's API (Algorithmia), but these tools are limited in their scope and performance, since research on summarization has still not reached a mature stage.
As mentioned earlier, companies doing research and development in natural language processing may be competitors with us at some point. There are many such companies in the software industry, such as Google, Microsoft, and Facebook. Our aforementioned focus on the news industry will help to differentiate us from other companies that decide to offer similar products. Another such competitor that is very relevant to our objectives is Bloomberg. Though the strength of Bloomberg's NLP research group would make it difficult for us to sell the product to Bloomberg News, Bloomberg is unlikely to sell this technology to other news websites given its own interests. Therefore, Bloomberg would not be a competitor with respect to other news websites.
We also view the large companies mentioned above as opportunities for a potential acquisition. Several NLP companies have been acquired recently by larger firms. For instance, in the last three years, API.ai was acquired by Google, Maluuba was acquired by Microsoft, and Semantic Machines was acquired by Microsoft (Kumparak; Shum; Ku).

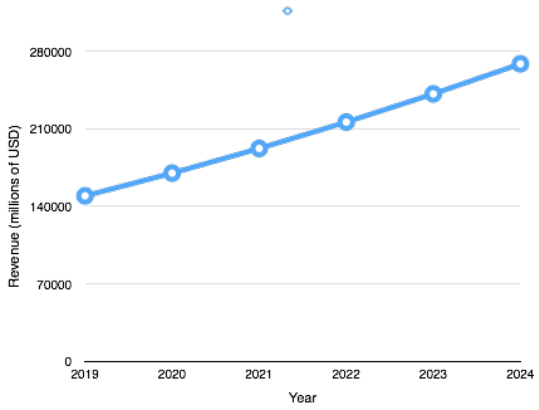

Figure 2: Projected revenue of the Internet content and publishing industry during the 2019-2024 period. Projections are those given by Hadad

Customer Segment Analysis

The primary customer segment for our APIs are publishers of online news. The newspaper industry has been declining in revenue in recent years due to the immense popularity of online content, and this decline is expected to continue into the near future (McGinley). However, newspaper publishers who were traditionally focused on the print medium have increasingly shifted focus to online publishing, and Garnett, the publisher with the highest circulation, has more than 117 million unique monthly visitors as of November 2018 (McGinley). The online-only news sector also shows promising signs of growth. The number of employees of "digital-native newsrooms" grew from approximately 7400 in 2008 to 13260 in 2017 (Pew Research Center). Moreover, revenue for the overall Internet content industry (which also includes publishers of other content and Internet companies like Alphabet and Facebook) is projected by IBISWorld to grow rapidly over the next five years; Figure 2 shows the projected growth according to Hadad. Therefore, we can expect that the health of the online news segment will be strong for the foreseeable future. Industry analysts also predict that Internet publishers will increasingly introduce access fees for customers (Freedonia). News publishers who decide to introduce access fees will need to convince their users that their websites provide value beyond the free articles offered by other websites. The faster synthesis of information enabled by our APIs would make it easier for news publishers to make this argument.

Revenue Model and Costs

For our revenue model, we will adopt a structure in which we charge the customer (e.g. a news company) for each API call that they make. This model makes sense because we will incur a variable cost for each API call (for the necessary computing power), and similar products (e.g. the APIs of Google Cloud NL and Plasticity AI) have this model. Our

pricing model for the coreference API is as follows. If the number of monthly requests is below $10,000$, then our price will be $\$2.00$ per 1000 requests. If the number of monthly requests is at least $10,000$, then our price will be $\$1.50$ per 1000 requests. The prices for our summarization API, if and when it is released in the future, will be twice the corresponding price for the coreference API (so $\$4.00$ and $\$3.00$). This model is based in part on the model of Plasticity AI's Sapien Language Engine (Plasticity AI, b) and also on the estimated cost of computing power. The cost of using a GPU for one hour on Google Cloud's Compute Engine is 0.95 (Google Cloud, b). We estimate that we can process at least $2,000$ documents in one hour using a single GPU.

Aside from computing power, our other major cost would be compensation for employees. We estimate that we will require 4 engineers, one of whom will serve as the engineering manager, and 4 researchers (focused on research in summarization), one of whom will be a senior researcher playing a supervisory rule. Each non-lead engineer will be paid about $\$120,000$, and the lead engineer will be paid about $\$140,000$. Each non-senior researcher will be paid about $\$150,000$, and the senior researcher will be paid about $\$175,000$. Furthermore, we estimate that we will need to add two (junior) engineers each year from Year 3 onward in order to scale the product. We estimate that aside from the computing costs and employment expenses discussed above, we will incur overhead of about $\$30,000$ per year (including costs for renting office space, purchasing and maintaining computers for development purposes, and other necessary costs).

Table 5 shows our projected revenues and costs for the first five years of the company. In order to compute the number of Monthly API calls for Y2, we used an estimate of $150,000,000$ total unique monthly users on news websites, estimated that 5% of such users would use the search engine on one of our client websites, estimated that each such user would make 5 searches in a month, and assumed that the client website would call our coreference API to process the top three documents resulting from the search. Note that the estimate of $150,000,000$ users in the market is conservative given that Garnett alone (which represents about 11.3% of the newspaper market) has $117,000,000$ unique monthly users. In order to compute the annual contribution margin, we divided the number of API calls by 1000 (since our price is on a per-1000-calls basis), multiplied by $\$1.5 - \frac{\$0.95}{2}$ (which is the difference between our price and our estimate for the variable cost for computing power), and multiplied by 12 to obtain the annualized estimate. To compute the revenues for years 3 through 5, we simply assume 150% growth over the previous year. Although this growth rate might seem too optimistic, it is reasonable given the conservative estimates of the size of the overall user base, the estimated initial portion of the user base that searches news websites, and the number of searches per user. These estimates yield net profit projections that are quite healthy by year 5 and do not take into account any potential revenue from the summarization product within this time frame. Beyond year 5, we expect that growth from the coreference API would decrease significantly but hope that the summarization API can drive growth at that point.

Conclusion

In this project, we introduced a technique for evaluating the robustness of a coreference resolution system to unseen names and an adversarial training method for a state-of-the-art neural coreference resolution system. We observed that replacing names in the evaluation text decreases the system's performance significantly and that the adversarial training method improves generalization as measured by multiple datasets.

Regarding future work, we observe both our method of replacing names and our method of adversarial training can be applied easily to other NLP tasks. In particular, the application of these techniques to Named Entity Recognition (NER) is an interesting direction. We conducted a preliminary experiment using the BERT-NER system (as described by Devlin et al. (2018)) and the popular CoNLL-2003 dataset Sang and De Meulder (2003) and found that the system is reasonably robust to replacements of PER names (drop of about 0.5 F1). However, it would be interesting to evaluate robustness to LOC (location) names in a manner similar to that described in this paper, and it would also be interesting to apply the adversarial training method described in this paper to a neural NER system.

References

Oshin Agarwal, Sanjay Subramanian, Ani Nenkova, and Dan Roth. Named person coreference in english news. *arXiv preprint arXiv:1810.11476*, 2018.

Algorithmia. Algorithms. https://algorithmia.com/algorithms. Accessed: 2019-04-24.

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2890–2896, 2018.

Amit Bagga and Breck Baldwin. Algorithms for scoring coreference chains. In *The first international conference on language resources and evaluation workshop on linguistics coreference*, volume 1, pages 563–566. Granada, 1998.

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. Adversarial training for multi-context joint entity and relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2830–2836, 2018.

Kai-Wei Chang, Rajhans Samdani, and Dan Roth. A constrained latent variable model for coreference resolution. In *EMNLP*, 2013. URL http://cogcomp.org/papers/ChangSaRo13.pdf.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

	Y1	Y2	Y3	Y4	Y5
Number of Monthly API calls	0	112.5	168.75	253.13	379.69
Annual Contribution Margin	0	$1.38	$2.08	$3.11	$4.67
Employee Expenses	$1.65	$1.65	1.89	$2.13	$2.37
Overhead	$0.03	$0.03	$0.03	$0.03	$0.03
Net Profit	−$1.68	−$0.30	$0.16	$0.95	$2.27

Table 5: Revenue and cost projections for the first five years of the company. All quantities are in millions.

Bhuwan Dhingra, Qiao Jin, Zhilin Yang, William Cohen, and Ruslan Salakhutdinov. Neural models for reasoning over multiple mentions using coreference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 42–48, 2018.

Freedonia. Internet content search: United states. Accessed: 2019-04-24.

Abbas Ghaddar and Philippe Langlais. Wikicoref: An english coreference-annotated corpus of wikipedia articles. In *LREC*, 2016.

Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. URL http://arxiv.org/abs/1412.6572.

Google Cloud. Natural language. https://cloud.google.com/natural-language/#natural-language-api-pricing, a. Accessed: 2019-04-24.

Google Cloud. Google compute engine pricing. https://cloud.google.com/compute/pricing, b. Accessed: 2019-04-24.

Jonathan Hadad. Internet publishing and broadcasting in the us. Accessed: 2019-04-24.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Matthew Honnibal and Mark Johnson. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, 2015.

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Peter Clark, Oren Etzioni, and Dan Roth. Question answering via integer programming over semi-structured knowledge. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, 2016. URL http://cogcomp.org/papers/KKSCER16.pdf.

David Ku. Microsoft acquires semantic machines, advancing the state of conversational ai. https://blogs.microsoft.com/blog/2018/05/20/microsoft-acquires-semantic-machines-advancing-the-state-of-conversational-ai/. Accessed: 2019-04-24.

Greg Kumparak. Google acquires api.ai, a company helping developers build bots that arent awful to talk to. https://techcrunch.com/2016/09/19/google-acquires-api-ai-a-company-helping-developers-build-bots-that Accessed: 2019-04-24.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 188–197, 2017.

Kenton Lee, Luheng He, and Luke Zettlemoyer. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692, 2018.

Xiaoqiang Luo. On coreference resolution performance metrics. In *Proceedings of the conference on human language technology and empirical methods in natural language processing*, pages 25–32. Association for Computational Linguistics, 2005.

Devin McGinley. Newspaper publishing in the us. Accessed: 2019-04-24.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

Takeru Miyato, Andrew M. Dai, and Ian Goodfellow. Adversarial training methods for semi-supervised text classification. *ICLR*, 2017. URL https://arxiv.org/abs/1605.07725.

Nafise Sadat Moosavi and Michael Strube. Which coreference evaluation metric do you trust? a proposal for a link-based entity aware metric. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 632–642, 2016.

Nafise Sadat Moosavi and Michael Strube. Lexical features in coreference resolution: To be used with caution. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 14–19, 2017.

Nafise Sadat Moosavi and Michael Strube. Using linguistic features to improve the generalization capability of neural coreference resolvers. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 193–203, 2018.

Eric W Noreen. *Computer-intensive methods for testing hypotheses*. Wiley New York, 1989.

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.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, 2018.

Pew Research Center. Digital news fact sheet. https://www.journalism.org/fact-sheet/digital-news/. Accessed: 2019-04-24.

Plasticity AI. Documentation. https://www.plasticity.ai/api/docs#sapien-core-coreference, a.

Plasticity AI. Pricing. https://www.plasticity.ai/api/pricing, b. Accessed: 2019-04-24.

Sameer Pradhan, Alessandro Moschitti, Nianwen Xue, Olga Uryupina, and Yuchen Zhang. Conll-2012 shared task: Modeling multilingual unrestricted coreference in ontonotes. In *Joint Conference on EMNLP and CoNLL-Shared Task*, pages 1–40. Association for Computational Linguistics, 2012.

Marta Recasens, Marie-Catherine de Marneffe, and Christopher Potts. The life and death of discourse entities: Identifying singleton mentions. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 627–633, 2013.

Erik F Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050*, 2003.

Harry Shum. Microsoft acquires deep learning startup maluuba; ai pioneer yoshua bengio to have advisory role. https://blogs.microsoft.com/blog/2017/01/13/microsoft-acquires-deep-learning-startup-maluuba-ai-pioneer-yoshua-bengio-advisory-role/. Accessed: 2019-04-24.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Josef Steinberger, Mijail Kabadjov, and Massimo Poesio. Coreference applications to summarization. In *Anaphora Resolution*, pages 433–456. Springer, 2016.

Sanjay Subramanian and Dan Roth. Improving generalization in coreference resolution via adversarial training. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM)*, 2019.

Rhea Sukthanker, Soujanya Poria, Erik Cambria, and Ramkumar Thirunavukarasu. Anaphora and coreference resolution: A review. *arXiv preprint arXiv:1805.11824*, 2018.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. A model-theoretic coreference scoring scheme. In *Proceedings of the 6th conference on Message understanding*, pages 45–52. Association for Computational Linguistics, 1995.

Hai Wang, Takeshi Onishi, Kevin Gimpel, and David McAllester. Emergent logical structure in vector representations of neural readers. *arXiv preprint arXiv:1611.07954*, 2016.

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

Yi Wu, David Bamman, and Stuart Russell. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783, 2017.