Text Analytics: Business Insider

Anand Raman

March 2020

1 Executive Summary

Using spaCy and supervised learning methods, CEO names, percentages, and company names were extracted from news articles published by Business Insider between the years 2013 and 2014.

1.1 Percentages

The percentage extraction was a built in feature with Spacy. Using this feature, the articles referred to percentages in 9942 distinct ways. The extracted list of percentages is available in the repo.

1.2 CEOs and Companies

CEOs and company names were extracted by filtering sentences for person and org respectively, then training a model to predict whether a sentence contained a CEO or Company. Persons and Organizations were extracted from the sentences predicted to have a CEO or company name in them. In total, 9374 distinct CEOs and 31955 companies were extracted. The lists of unique CEOs and companies are available in the repo.

2 Introduction

Business Insider is a news platform that publishes articles on a wide variety of industry verticals. The data used this project is 730 Business Insider articles – one for each day in 2013 and 2014. The goal of the project was to extract all percentages, CEO names and company names from these articles.

3 Data Preprocessing

Articles were initially contained in separate text files. To ease processing, all were appended as separate items in a list. Premliminary NER was performed with spaCy, all articles were transformed to Docs (a spaCy datatype).

3.1 Percentages

For percentages, preprocessing and percentage extraction concludes here. spaCy recognizes percentages with it's built in NER. So after NER was performed, matching percentages were extracted and deduplicated.

3.2 Companies and CEOs

3.2.1 Labeling Data

Training labels for CEOs and companies were deduplicated and filtered. This was necessary because some CEOs appeared in the list of companies and vice versa. For example, in the CEO labels, American Apparel was present.
Articles were broken into sentences using spaCy's sentencizer. NER features were used to filter the sentences. Two separate lists were created: one with sentences containing organizations and the other with sentences containing people. These lists were converted to separte DataFrames with one column each titled 'sentences'. From here, an indicator column was created in each DataFrame. In the organizations DataFrame, a function checked if any sentences contained a company. In the DataFrame of sentences containing people, the function checked which sentences contained CEO names from the training labels. Since the data is comprised of all people and organizations, there was no need to create additional negative samples because most sentences with people/organizations are not CEOs and companies.

3.2.2 Feature Engineering

Five features for each dataset were engineered "by hand".

- Number of capital letters in the sentence
- Number of words in the sentence
- Indicator for whether the sentence contains a quote
- Feature that indicated whether the sentence contained a top company/CEO, respectively for the org and person dataframes.
  - Used list of top 10 CEOs and companies from 2013
- Indicator for whether the sentence contained company (for org dataframe) and CEO or chief executive (for person dataframe).

After these features were extracted, the train-test split was performed. The test data was comprised of a random sample of 25% of the data. Following the train-test split, TF-IDF was fit on the training data's sentences. After fitting, both the train and test sentences were transformed. The final training dataset included 10000 features from TF-IDF and the five features extracted by hand. The data was stored as a sparse matrix.

4 Modeling

Distinct modeling methods were used in finding companies and CEOs.

4.1 Companies

Logistic regression provided the best result for predicting the presence of a CEO within a sentence. The max iteration counter was increased and the class weight parameter was set to 'balanced'. The class weight parameter shifts the decision boundary for logistic regression as the classes were imbalanced for the companies data (only 38% of sentences included a company name). Two other model types were implemented: gradient boosted trees and random forests. Both performed less well than the logistic regression model.

4.2 CEOs

Gradient boosted trees provided the best result for predicting the presence of a CEO in a sentence. The number of trees was set to 100 and the learning was increased to 0.6 (default=0.1). Classes were highly imbalanced for CEO data – only 12% of sentences with people in them contained the name of a CEO. Logistic regression, even with the class weight parameter set to 'balanced' did not perform as well as the gradient boosted classifier.

5 Results

5.1 CEOs

5.1.1 Descriptives

In total, 199,827 sentences contained the name of a person. CEO names were present in 12.7% of these sentences. The average number of capital letters in a sentence containing a person's name was 7.37 with a standard deviation of about 10. The average number of words in a sentence was 25.8 with a standard deviation of 21.7. Approximately 49% of sentences contained a quote. 0.7% of sentences contained the name of one of the 'best' CEOs from 2013. And just 2.7% of sentences contained the phrase 'chief executive officer' or 'CEO'.

5.1.2 Model Results

The gradient boosting classifier successfully classified 96% of sentences. For sentences with CEOs, the precision was 0.91 and the recall was 0.80. The F1 score was 0.85. I selected the GBC over logistic regression even though the recall of the logistic regression model was higher. On the other hand, F1 and precision were higher for the GBC. I considered using the logistic regression model because we were told that some CEOs do not appear in the list of training labels. But since we do not know the proportion of CEOs that do not exist in the training labels, I reasoned that it would be irresponsible to assume that logistic regression had more successfully identified unlabeled CEOs. The full results of the model on the test data are available below.

	precision	recall	f1-score	support
0	0.97	0.99	0.98	43512
1	0.91	0.80	0.85	6445
accuracy			0.96	49957
macro avg	0.94	0.89	0.91	49957
weighted avg	0.96	0.96	0.96	49957

5.2 Companies

5.2.1 Descriptives

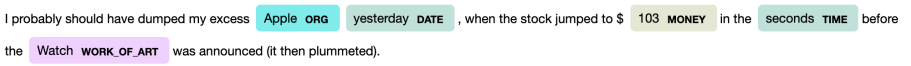
In total, the dataset of sentences containing an organization name was 303707 rows. The name of a company appeared in 38.07% of sentences. The average number of capital letters was 6.82. The average number of words in a sentence was 25.13 with a standard deviation of 18.28. 40% of sentences had a quote in them. 2% of sentences had the name of one of the "best" companies from 2013 in them. 4.2% of sentences included the word company.

5.2.2 Model Results

Logistic regression successfully classified 93% of sentences in the test data. The precision for the sentences with company names was 0.94. The recall was 0.88. This resulted in an F1 of 0.91. Several other methods were attempted include gradient boosting classifiers and random forests, but the best model was logistic regression. The full results of the model are reported below.

	precision	recall	f1-score	support
0	0.93	0.96	0.95	47022
1	0.94	0.88	0.91	28905
accuracy			0.93	75927
macro avg	0.93	0.92	0.93	75927
weighted avg	0.93	0.93	0.93	75927

Below is an example of a sentence containing a true positive and the company that was extracted.



5.3 False Positives

For both CEOs and companies I selected rows that had a confidence of 2.0 or greater but had a 'true' label of 0. I thought these might be good candidates for data that was not in the labels but did in fact contain a CEO or company. Inspecting these manually, it's really a mixed bag, but the models did manage to correctly identify a substantial number of CEOs and companies not included in the training labels. However, I am not capable of manually inspecting all of the candidates. There were 839 candidates in the list of 'false' positives for companies and 300 candidates in the list of 'false' positives for CEOs. I've included these in the repo because they could become useful when building the Q&A system.