COMP4420 Project Report: Sarcasm Detection in Headlines

Bui, Nam (#01963609), Conners, Riley (#01943861), Zuk, Sam (#01642608)

1 Abstract

This project seeks to explore different sentiment analysis techniques in the task of sarcasm detection in newspaper headlines. We compare the performance of a Naive Bayes model and LSTM model, with and without pre-trained word2vec embeddings, on the task. Naive Bayes achieved roughly 80 % accuracy and the LSTM approaches achieved roughly 90 %. Different embeddings had a negligible effect on classification, but the switch from a Naive Bayes to an LSTM approach showed significant improvement.

2 Introduction

Sarcasm is defined as the use of words that convey the opposite meaning to cause or show irritation. [1]

Sarcasm is a feature of natural language that is notoriously difficult to define and identify in both the spoken and written word. The assumption that a statement will be recognized as sarcastic is typically contingent upon the listener/reader knowing some outside piece of contextual information beforehand. However, this external information isn't always known, and even when it is, the relationship between it and the statement at hand may not always be clear. When this happens, the meaning can be obscured as a result, often leading to avoidable scenarios involving miscommunication.

Recognizing sarcasm typically involves picking up on subtle cues and nuance that can be difficult to identify. This can often pose a challenge for populations who encounter greater difficulty when processing certain aspects of a language. For example, someone trying to interpret a language they don't speak natively will likely have to expend more mental effort to parse out meaning from words, which in turn makes it more difficult to pick up on nuance, including sarcasm. Being unfamiliar with the cultural norms, idioms, etc. that inform the established meaning of the locally spoken language can also be a source of confusion. In addition, many neurodivergent people, in particular those with autism, can struggle to recognize and/or communicate certain social cues in conversation due to differences between their cognitive experience of language and what is expected of them.

Finally, there are unique challenges faced in detecting sarcasm in the written word. It is often possible in practice to infer a statement is sarcastic, even without necessarily having the context to understand *why* by listening to changes in the tone of the speaker. However, when translated into the written word, some or all of this information is lost, making sarcasm even more difficult to detect when only text is given. With the Internet now being extremely important to modern infrastructure, and with text being the predominant medium for online communication, this problem has become increasingly apparent over the years. This project shall explore and contrast different approaches to disambiguating sarcasm by applying concepts from the fields of computational linguistics and machine learning.

3 Method

First we took the dataset described in section 4 and partitioned it into train, validation, and test sets with a 70/20/10 split respectively. The test set labels were then manually verified.

The steps for tokenizing the dataset were:

1. Tokenize hyphens.
2. Tokenize single quotes.

3. Transform contractions to canonical form.
4. NLTK word tokenize.
5. Address edge cases.

The vocabulary included all tokens with $count > 5$.

A Naive Bayes model was used to get initial performance baselines. Along with the tokenized dataset, lemmatization was used to group words with the same meaning, like 'says' and 'said', together. Hyperparameter search was done on the smoothing parameter of the model. We found that smoothing factor $\alpha = 1.5$ performed the best, although other parameters performed closely.

Word2vec embeddings pre-trained on the Google News dataset were then fine-tuned over the sarcasm dataset to better fit the dataset. [2] Embeddings for words that were common and unique to the dataset were also added. Since word2vec does not have an unknown token, we mapped the unknown token to the 0 vector, which is what Rishabh and Prahal did in their research. [3] An LSTM model was then trained with and without the pre-trained embeddings. The architecture of the LSTM model consisted of embedding layer, LSTM layer, and a fully-connected feedforward network. Gradient clipping, early stoppage, batching, dropout, and learning rate scheduling were used during training. Results can be found in section 5.

4 Data

The dataset used for this project is a collection of 28,619 tagged newspaper headlines – of which 13,635 are from the satirical publication *TheOnion*, the other 14,984 being from the non-satirical publication *The Huffington Post* (*HuffPost*). The data was collected from TheOnion's "News in Brief" and "News in Photos" sections and HuffPost's news archive page in 2019 [3].

For each headline, the dataset contains a JSON object with three attributes:

- `is_sarcastic` (integer): the headline's label – 1 if sarcastic, 0 if not.
- `headline` (string): the text of the headline, case-converted to be all lowercase.
- `article_link` (string): the URL of the referenced article.

This project used the contents of `headline` to predict the value of `is_sarcastic`. Values of `article_link` were not used to detect sarcasm because the labels for sarcasm were source-based, so detecting sarcasm from the article link in this dataset is trivial. However, for other experiments they may be useful when attempting to decipher why certain models made certain predictions.

This dataset has advantages over text that could be found on social media platforms because news text is formal in nature. This means there are less words outside of the word2vec vocabulary, less spelling mistakes, and little to no slang usage. Also, because The Onion is openly sarcastic by design, there is no ambiguity regarding if labels are correct.

However, there are downsides this dataset. In this case, there are only two news sources being used, and the model could pick up on writing styles or other details instead of sarcasm. There is another potential issue that stems from The Onion's obvious use of sarcasm; in cases where sarcasm is more subtle, a model trained on this dataset could do poorly.

The data was split 70/20/10 into training, validation, and testing sets, each with equal proportion of genuine and sarcastic articles. All articles labeled as genuine in the test dataset were manually reviewed to ensure there were no incorrect labels.

5 Results

To compare our results with the results of Rishabh and Prahal, we measured the accuracy of our models on the test set. [3] Additionally, since this is a binary classification task, we also used precision, recall, and F1 metrics.

6 Conclusion

In conclusion, we found that a neural network approach like the LSTM model was more powerful than a simpler concept like Naive Bayes. Sarcasm is a complex concept to identify, especially with only text input. Having a model that can hit near 90% accuracy is a good improvement on previous results and shows the power of an LSTM over Bayes.

This project could be extended in the future to work on further customized dictionaries with word embeddings for new words like Trump. Customized embeddings for names and/or events could lead to even better recognition of the meaning behind headlines. In terms of the model, an attention or dropout layer could also be added to see how it affects performance.

7 Contribution Chart:

Student Name & ID	Tasks/Subtasks	Commentary on Contribution
Bui, Nam (#01963609)	Tokenized Dictionary Created and Ran Bayes Model Created LSTM Model Debugged Models	
Conners, Riley (#01943861)	Split Data and Validated Tests Created Data Loader Ran Initial Runs of LSTM	
Zuk, Sam (#01642608)	Exploratory Data Analysis Tokenized Dictionary Created Custom Word Embeddings Ran Final Run of LSTM Model	

8 References

[1] Merriam-Webster. Sarcasm. `https://www.merriam-webster.com/dictionary/sarcasm`.

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[3] Rishabh Misra and Prahal Arora. Sarcasm detection using news headlines dataset. *AI Open*, 4:13–18, 2023.