Project Proposal: Sarcasm Detection

Esha Wang, M.Sc. Stanford University
Kelvin Fang, Ph.D. Nokia Bell Labs

1 Key Information

- Title: The Best Sarcasm Detector Ever Created
- Team member names: Esha Wang (eshawang@stanford.edu)
- External collaborators: Kelvin Fang (kfang11@gmail.com)
- Custom or default project: Custom
- Mentor: We have no particular mentor.

2 Research Paper Summary

2.1 Bibliographical Info

[Amir et al. 2016] Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 2016. Modelling context with user embeddings for sarcasm detection in social media. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, pages 167-177. Berlin, Germany, August. https://arxiv.org/pdf/1607.00976.pdf.

2.2 Background

Accurate detection and interpretation of figurative language is a difficult task for existing NLP systems. Especially in domains such as social media and politics, sarcasm is a commonly used, and often misinterpreted, communicative device, in which speakers say something other than, and usually opposite to, what they actually mean. Early computational models for sarcasm detection used mainly lexical clues (such as emoticons, heavy punctuation, and positive interjections), features based on word and character n-grams, or identification of contrasting sentiments in expressions to classify sarcastic statements. However, the exclusive use of features intrinsic to the text in question in determining figurative meaning is often insufficient - context is needed.

By way of the sarcasm device, the exact same sentence can be interpreted as literal or ironic, depending on the speaker and setting. Even for humans, knowing the context of a sentence is critical for interpretation, such as an author's political leanings or the expected outcome of a described situation. Recent attempts to incorporate contextual information into sarcasm detection have included detecting contrasts in sentiments expressed towards named entities, inferring behavioral traits of authors, and capturing relationships between authors and audiences. However, in order to model those contexts in a general way, it has previously been necessary to engineer extensive sets of features using profile information, post histories, and/or interactions within specific communities. The amount of manual effort required to derive these feature sets has been a major downside to these approaches.

2.3 Summary of Contributions

This paper proposes a novel approach to sarcasm detection on social media (Twitter) that obviates the need for extensive manual feature engineering. The authors instead develop a neural model that learns to represent and exploit embeddings of both content (words) and context (users). Inference concerning whether an utterance (tweet) was intended ironically or not is modelled as a joint function of lexical representations and corresponding author embeddings.

Vector representations for tweet authors or users are learned in a way that projects similar users into nearby regions of the embedding space. The authors hypothesize that the embeddings can encode latent aspects of users, naturally capturing homophily and other signals that are important indicators of sarcasm. (This is based on the principle of inferability, stating that sarcasm requires a common ground between parties to be understood.) To induce the user embeddings, the authors optimize the conditional probability of texts, given the vector representations of their authors. *i.e.* given a sentence $S = \{w_1, \ldots, w_N\}$ where w_i denotes a word drawn from a vocabulary V, they aim to maximize:

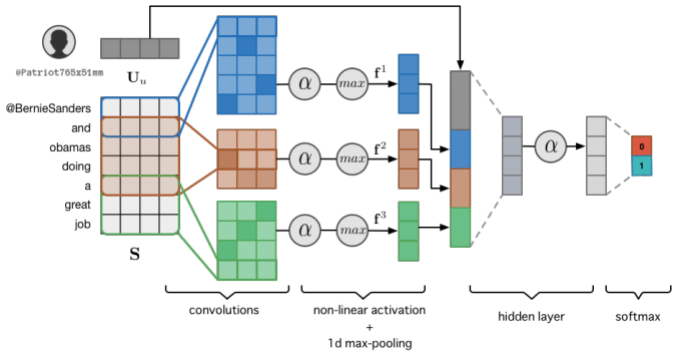
$$P(S|\text{user}_j) = \sum_{w_i \in S} \log P(w_i|\mathbf{u}_j) + \sum_{w_i \in S} \sum_{w_k \in C(w_i)} \log P(w_i|\mathbf{e}_k),$$

where $C(w_i)$ denotes the set of words in a pre-specified window around word w_i, and $\mathbf{e}_k$ and $\mathbf{u}_j$ denote the embeddings of word k and user j, respectively. The conditional probabilities are estimated by:

$$P(w_i|\mathbf{x}) = \frac{\exp(\mathbf{W}_i \cdot \mathbf{x} + \mathbf{b}_i)}{\sum_{k=1}^{Y} \exp(\mathbf{W}_k \cdot \mathbf{x} + \mathbf{b}_k)},$$

where $\mathbf{x}$ denotes a feature vector, and $\mathbf{W}_k$ and $\mathbf{b}_k$ are weight vectors and bias for class k. In this case, words are treated as classes to be predicted.

The proposed sarcasm detection model is composed of 3 sliding filters that comprise the convolution layer on the sentence embedding, a ReLU activation layer followed by 1d max-pooling, and concatenation with the corresponding user embedding before a final hidden activation layer and softmax layer. This approach, shown in the schematic below, is dubbed the *Content and User Embedding Convolutional Neural Network*, or CUE-CNN.



The CUE-CNN outperforms the strong baseline and previous state-of-the-art by more than 2% in absolute accuracy, and it is shown that the learned user embeddings can capture relevant user attributes.

2.4 Limitations and Discussion

The authors of this paper take great care in providing suitable baseline models, both for the prior state-of-the-art (SoA) (Bamman and Smith, 2015) and the proposed CUE-CNN, with increasing complexities for comparison. The experimental setup, including the dataset generation and feature set engineering, was meticulously replicated from Bamman and Smith, 2015. However, the dataset ultimately ended up being significantly smaller than that used in the prior SoA due to deleted tweets and new restrictions in the Twitter API. This discrepancy could potentially contribute to the paper's perceived strength, as baseline results were found to be worse than

those reported in the prior SoA across the board. On the other hand, the proposed CUE-CNN models achieved higher accuracy than the prior SoA reported, to more than 1 standard deviation over 10-fold cross-validation. Given the evident effort in creating a fair comparison and plausible explanation of baseline discrepancy, the findings of this paper are still convincing. One additional metric that could be provided is the total training times of the neural and baseline models, as well as an estimation of the manual effort dedicated to feature engineering.

2.5 Why This Paper?

This paper was chosen after doing an (decently) exhaustive search of the sarcasm detection literature. The topic of contextualized sentiment classification matched our own initial ideas for a sarcasm classification project. As we describe later in the Project Description, we hope to build upon this work in both classification method and scope of task. Having read the paper in depth, we have gained a better understanding of rigorous experimental setup as well as a solidified model to act as our starting point.

2.6 Wider Research Context

The broader story of NLP research is well-supported by this work. In all languages, context is a fundamental, yet highly intangible, factor in discerning the intended meaning of text or speech. While this paper focuses on the incorporation of contextual features into sarcasm detection models, the theory is widely applicable to general sentiment analysis, and beyond. A fuller understanding of connotation and interpretation in context motivates research in code-mixing (Mathur et al., 2018), machine translation, and even traditional linguistics (Campbell and Katz, 2012), among others.

2.7 References

[Bamman and Smith 2015] David Bamman and Noah A. Smith. 2015. Contextualized sarcasm detection on Twitter. In *Proceedings of the 9th International Conference on Web and Social Media*, pages 574-577. Menlo Park, CA, USA, May. https://www.aaai.org/ocs/index.php/ICWSM/ICWSM15/paper/view/10538.
[Campbell and Katz 2012] John D. Campbell and Albert N. Katz. 2012. Are there necessary conditions for inducing a sense of sarcastic irony? *Discourse Processes*, 49(6):459-480. https://www.tandfonline.com/doi/abs/10.1080/0163853X.2012.687863.
[Mathur et al. 2018] Puneet Mathur, Rajiv Shah, Ramit Sawhney, and Debanjan Mahata. 2018. Detecting offensive tweets in Hindi-English code-switched language. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 18-26. Melbourne, Australia, July. https://www.aclweb.org/anthology/W18-3504/.

3 Project Description

3.1 Main Goals

The main goal of this project is to develop a generalized sarcasm detection model for a variety of dataset types and to study the use and effectiveness of contextual features in identifying sarcasm with high accuracy. This project aims to build upon the work of Amir et al., 2016 and many others in expanding the target application beyond Twitter to not only other social media sites, but also news headlines and more generic dialogues. Since most datasets and real world situations have substantially less available contextual history than Twitter (usually a few interactions instead of up to thousands of tweets/posts), this project is motivated by the need for more robust context embedding. A critical objective is to determine how to efficiently encode embeddings for a multitude of datasets into a general purpose CUE-CNN-like model.

3.2 NLP Tasks

The principal NLP task addressed by this project is sentiment analysis, specifically that of sarcasm or irony. Given an input text phrase or sentence (along with supporting context), our model aims to identify and output whether the utterance was meant to be interpreted literally or sarcastically by the author. A secondary goal of this project is to improve our understanding of contextual embeddings in the context of language modeling in general.

3.3 Data

We aim to apply context-embedded sarcasm detection to a diverse collection of publicly available datasets. Oraby et al., 2016 provides a corpus of 1,630 quote-response pairs of generic dialogue, with responses labeled as sarcastic or not. The quote-response pairs are further categorized as belonging to one of general sarcasm, hyperbole, or rhetorical question subsets. Khodak et al., 2017 contains a collection of 1.3 million labelled comments from Reddit, along with what they responded to. The dataset has balanced and imbalanced versions, where the imbalanced version features the true distribution of sarcastic to non-sarcastic comments (about 1:100). Non-sarcastic comments are from the same threads as sarcastic ones. Abercrombie 2018 provides a corpus of author-audience Twitter conversations with manually annotated labels (not relying on the #sarcasm hashtag) that is disjoint from the Amir et al. 2016 and Bamman and Smith 2015 datasets. This collection contains 2241 tweet pairs, of which 449 are sarcastic. While this is a significantly smaller dataset than those trained on in the referenced papers, it is unique in that the authors of sarcastic comments did not label their tweets themselves, leading to potentially more colloquial or unintentional sarcastic sentiments. Finally, Misra and Arora, 2018 contains 28,619 news headlines (sarcastic titles from *The Onion* and non-sarcastic from *HuffPost*). Since the news headlines are not initially paired with any type of context, we plan to collect additional data from Google by searching each title and pairing it with the first result not from *The Onion* or *HuffPost*, which should result in high relevancy context. For all datasets, we will preprocess out any self-labeling (*e.g.* the /s tag on Reddit), as well as removing links, images, and emojis/emoticons. We will then utilize a TF-IDF vectorizer, a commonly used NLP algorithm for transforming text into vector embeddings, which can then be used for feature extraction and provide the starting point for our neural net training.

3.4 Neural Methods

As alluded to previously, we look to leverage CNN models similar to the CUE-CNN developed in the reference paper. However, in generalizing the contextual embedding method for the datasets described, we must re-evaluate how to properly encode 'user' embeddings. Without individual user profiles containing large post histories, the data scarcity issue touched upon by Amir et al., 2016 is further exacerbated. Thus, in selectively sampling negative examples for embedding approximation, we will use a unigram distribution estimated from all non-sarcastic comments in each training set. Following from this, instead of having multiple user embeddings, we hypothesize that a single generalized embedding based on the aggregate data will be sufficient for context encoding on an individual dataset-to-dataset basis. With a single context embedding, we will lose some predictive capability (accuracy) in polarizing discussions/threads (*e.g.* politics), but for most day-to-day dialogue, we expect people to be roughly equal in their semantic responses to a given situation (*e.g.* waiting at the airport). Finally, we will expand on the CUE-CNN model by incorporating an additional input, as all of our datasets include pairs of utterances (naïvely, we can just convolve the two).

3.5 Baselines

We plan to develop a few baseline models similar to those described in the reference paper for comparison with our CNN models on the provided datasets. These include 'UNIGRAMS', an l_2-regularized logistic regression classifier with binary unigrams as features, 'NBOW', logistic regression with neural word embeddings as features, 'NLSE', a non-linear embedding model that projects each word vector into a small subspace to capture the most discriminative encoded latent aspects before transforming through an element-wise sigmoid,

and baseline/shallow CNNs, which will disinclude combinations of context embedding, quote inputs, and pre-trained weights. If there is extra time, we will also manually engineer feature sets for implementing models from Bamman and Smith, 2015. Previously published results from the reference paper will serve as additional baselines to compare our work against, especially for the Twitter dataset.

3.6 Evaluation

Our main evaluation metric for our models will be test accuracy across the different datasets. This is for comparison with the reported results from the reference paper (87.2 and 86.4%) and Bamman and Smith, 2015 (84.9 and 85.1%). While we expect our method to achieve lower accuracy given the smaller datasets and generalized context embedding, this metric allows for direct interpretation of our results. Additionally, the referenced papers report accuracy values for a number of their own baseline models. For a slightly more meaningful evaluation of our models' performances, especially for datasets with large imbalance, we will also look at achieved F-scores. Overall, this project aims to derive a somewhat qualitative understanding of contextual embeddings from the results across all datasets.

3.7 References

[Abercrombie 2018] Gavin Abercrombie. 2018. Corpus of sarcasm in Twitter conversations. *Mendeley Data*. http://dx.doi.org/10.17632/fn2mmff85g.1.

[Khodak et al. 2017] Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. 2017. A large self-annotated corpus for sarcasm. *arXiv*. https://arxiv.org/abs/1704.05579.

[Misra and Arora 2018] Rishabh Misra and Prahal Arora. 2018. Sarcasm detection using hybrid neural network. *arXiv*. https://rishabhmisra.github.io/publications.

[Oraby et al. 2016] Shereen Oraby, Vrindavan Harrison, Lena Reed, Ernesto Hernandez, Ellen Riloff and Marilyn Walker. 2016. Creating and characterizing a diverse corpus of sarcasm in dialogue. In *The 17th Annual SIGdial Meeting on Discourse and Dialogue (SIGDIAL)*, pages 31-41. Los Angeles, CA, USA, September. https://nlds.soe.ucsc.edu/sarcasm2.