The Best Sarcasm Detector Ever Created

Esha Wang
Department of Statistics
Stanford University
eshawang@stanford.edu

May 23, 2023

Abstract

The aim of this project is to develop a content embedding CNN (CE-CNN) for generalized sarcasm detection over a wide range of human dialogue corpuses. For robust learning, we plan to adapt a Twitter user embedding model to instead create a singular generic context-aware human language profile. We will train our CE-CNN over quote-response pairs generated from Twitter, Reddit, and human dialogue sarcasm datasets and study the behavior of the neural net relative to a number of baseline models. We envision that our embedding model will be sufficient in achieving overall high accuracy in the sarcasm detection NLP task while maintaining robust context encoding for corpuses with little available user history.

1 Key Information

- External collaborators: Kelvin Fang (kfang11@gmail.com)

2 Approach

2.1 CE-CNN Architecture

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. We look to leverage CNN models similar to the CUE-CNN described in [1]. However, without individual user profiles containing large post histories, the data scarcity issue touched upon by [1] 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). Our content embedding CNN (CE-CNN) will incorporate two inputs, as our datasets will include contextual features in the form of quote-response pairs. This can be done as a concatenation or convolution, among other techniques. Equations describing the context embedding are given in the project proposal and [1], and a figure of our CE-CNN is TBD. All code for the CE-CNN will be written ourselves using the NLTK [2], Gensim [3], and PyTorch [4] Python packages.

2.2 Baselines

We develop a number of baseline models similar to those described in [1] for comparison with the CE-CNN 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 embedding summations 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 (TBD), which will disinclude combinations of context embedding, quote inputs, and pre-trained weights. Previously published results from the reference paper serve as additional baselines to compare our work against, especially for the Twitter dataset. All code for the baseline models is written ourselves using NLTK, Gensim, and PyTorch. Gensim word2vec is used to train independent word embeddings on each dataset, with nominal window size of 5 and dimensionality 300, for NBOW and NLSE. The NLSE subspace hidden layer has nominal dimensionality 10, and a representative figure of the neural architecture is given in [5].

3 Experiments

3.1 Data

We aim to apply context-embedded sarcasm detection to a diverse collection of publicly available datasets. [6] 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. [7] 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. We parse each thread into quote-response pairs, with responses being labelled comments and quotes being concatenated chains of their top level posts/comments, using ijson [8].

[9] provides a corpus of author-audience Twitter conversations with manually annotated labels (not relying on the #sarcasm hashtag) that is disjoint from the dataset in [1]. This collection contains 1105 tweet pairs, of which 193 are sarcastic (after discarding inaccessible/deleted tweets). While this is a significantly smaller dataset than that of [1], 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. Tweet contents are retrieved via their IDs through the Twitter API using Twython [10].

3.2 Evaluation Method

The main evaluation metric for our models is test accuracy across the different datasets. This is for comparison with the reported results from [1] (87.2 and 86.4%). 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, [1] reports 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.

3.3 Experimental Details

Baseline models are trained independently on the aforementioned datasets, including both balanced and unbalanced versions of [7]. The Dialogue and Twitter datasets are partitioned into training and test sets via a random 75%/25% split, and the Reddit dataset is already split. UNIGRAMS logistic regression is run for a maximum of 200 iterations and scaled to unit variance, with both binary and non-binary (frequency-based) implementations. NBOW is trained up to 1000 iterations and normalized with centered mean and unit variance in CBOW and skip-gram variations. Both baselines use l_2 regularization to reduce overfitting. The NLSE neural net is constructed with one subspace embedding hidden layer with randomly initialized weights, a logistic sigmoid layer, BOW summation layer, and softmax output. Model parameters such as learning rate, batch size, dimensionality, momentum, and early stopping are still being fine-tuned at the time of this milestone report. Development of the CE-CNN has also begun with the implementation of negative sampling of quote words for generalized context embedding.

3.4 Results

Table 1 shows accuracy results for the described baseline experiments. We observe that skip-gram NBOW performs the best for the Dialogue and balanced Reddit datasets. Accuracy metrics across the Twitter and unbalanced Reddit datasets are roughly equivalent due to data imbalance and models predicting

'Non-sarcastic' for each example. NBOW takes significantly longer to train on the large unbalanced Reddit dataset, and final accuracy results are not yet available. Additionally, further tuning of the maximum iterations and standardization should improve the overall accuracies on the balanced Reddit data. Confusion matrices for the experiments are shown in Figures.

Dataset	Model	Accuracy
Dialogue [6]	Binary UNIGRAMS	0.676
Dialogue [6]	Frequency UNIGRAMS	0.650
Dialogue [6]	Continuous NBOW	0.702
Dialogue [6]	Skip-gram NBOW	0.719
Twitter [9]	Binary UNIGRAMS	0.791
Twitter [9]	Frequency UNIGRAMS	0.789
Twitter [9]	Continuous NBOW	0.790
Twitter [9]	Skip-gram NBOW	0.790
Reddit (Balanced) [7]	Binary UNIGRAMS	0.524
Reddit (Balanced) [7]	Frequency UNIGRAMS	0.530
Reddit (Balanced) [6]	Continuous NBOW	0.505
Reddit (Balanced) [6]	Skip-gram NBOW	0.549
Reddit (Unbalanced) [6]	Binary UNIGRAMS	0.973
Reddit (Unbalanced) [6]	Frequency UNIGRAMS	0.973

Table 1: Accuracy results of baseline experiments.

4 Future work

Further work on this project throughout the remainder of the term will focus on the continued development of the proposed CE-CNN model and evaluation of its performance on the various datasets. To better understand the role of context embedding in sarcasm detection, we will train baseline variations of the architecture disincluding combinations of context embedding, quote inputs, and pre-trained weights. We would also like to perform more substantive data preprocessing and cleaning (*e.g.* expanding acronyms, correcting misspellings, etc.) and analyze model performances over aggregates of the datasets. Finally, if time allows, we will include the sarcastic news dataset from [11], generating context for each headline through a Google search algorithm, in our evaluation.

References

[1] Silvio Amir, Byron C. Wallace, Hao Lyu, Paula Carvalho, and Mário J. Silva. 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 2016.

[2] NLTK project, August 2019.

[3] Radim Řehůřek. Gensim, November 2019.

[4] PyTorch, March 2020.

[5] Ramón Astudillo, Silvio Amir, Wang Ling, Mário Silva, and Isabel Trancoso. Learning word representations from scarce and noisy data with embedding subspaces. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1074–1084, Beijing, China, July 2015.

[6] Shereen Oraby, Vrindavan Harrison, Lena Reed, Ernesto Hernandez, Ellen Riloff, and Marilyn Walker. 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, September 2016.

[7] Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. A large self-annotated corpus for sarcasm. *arXiv*, March 2018.

[8] ijson, March 2020.

[9] Gavin Abercrombie. Corpus of sarcasm in Twitter conversations. *Mendeley Data*, February 2018.

[10] Ryan McGrath. Twython, 2013.

[11] Rishabh Misra and Prahal Arora. Sarcasm detection using hybrid neural network. *arXiv*, August 2019.

Figures

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	350	217	567
	Sarcastic	163	443	606
	Total	513	660	1173

Figure 1: Confusion matrix for binary UNIGRAMS on Dialogue dataset (F_1 = 0.70) ([6]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	455	112	567
	Sarcastic	299	307	606
	Total	513	660	1173

Figure 2: Confusion matrix for frequency UNIGRAMS on Dialogue dataset (F_1 = 0.60) ([6]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	443	0	443
	Sarcastic	117	0	117
	Total	560	0	560

Figure 3: Confusion matrix for binary UNIGRAMS on Twitter dataset (F_1 = 0.00) ([9]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	442	1	443
	Sarcastic	117	0	117
	Total	559	1	560

Figure 4: Confusion matrix for frequency UNIGRAMS on Twitter dataset (F_1 = 0.00) ([9]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	31897	436	32333
	Sarcastic	30329	2004	32333
	Total	62226	2440	64666

Figure 5: Confusion matrix for binary UNIGRAMS on balanced Reddit dataset ($F_1 = 0.12$) ([7]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	31897	436	32333
	Sarcastic	30329	2004	32333
	Total	62226	2440	64666

Figure 6: Confusion matrix for frequency UNIGRAMS on balanced Reddit dataset ($F_1 = 0.12$) ([7]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	1637718	0	1637718
	Sarcastic	44637	0	44637
	Total	1682355	0	1682355

Figure 7: Confusion matrix for binary UNIGRAMS on unbalanced Reddit dataset ($F_1 = 0.00$) ([7]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	1637704	14	1637718
	Sarcastic	44637	0	44637
	Total	1682341	14	1682355

Figure 8: Confusion matrix for frequency UNIGRAMS on unbalanced Reddit dataset ($F_1 = 0.00$) ([7]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	360	207	567
	Sarcastic	143	463	606
	Total	503	670	1173

Figure 9: Confusion matrix for continuous NBOW on Dialogue dataset ($F_1 = 0.72$) ([6]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	369	198	567
	Sarcastic	132	474	606
	Total	501	672	1173

Figure 10: Confusion matrix for skip-gram NBOW on Dialogue dataset ($F_1 = 0.74$) ([6]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	334	0	334
	Sarcastic	89	0	89
	Total	423	0	423

Figure 11: Confusion matrix for continuous NBOW on Twitter dataset ($F_1 = 0.00$) ([9]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	334	0	334
	Sarcastic	89	0	89
	Total	423	0	423

Figure 12: Confusion matrix for skip-gram NBOW on Twitter dataset ($F_1 = 0.00$) ([9]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	13710	18623	32333
	Sarcastic	13378	18955	32333
	Total	27088	37578	64666

Figure 13: Confusion matrix for continuous NBOW on balanced Reddit dataset ($F_1 = 0.54$) ([7]).

		True values		
		Non-sarcastic	Sarcastic	Total
Predicted values	Non-sarcastic	20813	11520	32333
	Sarcastic	17657	14676	32333
	Total	38470	26196	64666

Figure 14: Confusion matrix for skip-gram NBOW on balanced Reddit dataset ($F_1 = 0.50$) ([7]).