A deep learning framework to detect sarcasm targets

Jasabanta Patro, Srijan Bansal, Animesh Mukherjee

Indian Institute of Technology Kharagpur, India – 721302

Objective

To identify statistically significant socio-linguistic features to characterise sarcasm targets and propose a deep learning framework for sarcasm target detection in pre-defined sarcastic texts.

1. Introduction

- **Sarcasm target** is defined as the entity or situation that is being mocked or ridiculed at in the sarcastic text. For example, ***I** love to be ignored.*
- **Multiple candidate phrases**: There can be multiple target candidate phrases present in the sarcastic text. For example, *The **laptop** heats up so much that I strongly recommend chefs to use it as a cook-top*. The target candidates could be 'chefs','cook-top' and 'laptop'.
- **Multiple sarcasm targets**: There can be multiple sarcasm target phrases present in the sentence. For example, ***I** used to be a middle-of-the-road kid, but now with **my freaky looks** I'm definitely an outsider. Hooray.*
- **Absence of any target**: It is also possible that no sarcasm target is present at all in the sarcastic text. For example *Oh, and I suppose the apples ate the cheese.*. In this case, sarcasm targets are labelled as **outside**.

2. Dataset and Preprocessing

- 224 sarcastic book snippets with average snippet length of 27.74 & average sarcasm target length of 1.6.
- 506 sarcastic tweets with average tweet length of 12.97 & average sarcasm target length of of 2.08.
- **Outside** if annotators do not find specific words in the text that correspond to a target.

3. Socio-linguistic features

Various socio-linguistic features that show statistically significant differences between the words corresponding to the sarcasm targets and the rest of the words in the sarcastic text.

- The distribution of **location (LOC) and organisation (ORG) named entities** are significantly different for the sarcasm target words compared to the other words ($p < 0.001$).
- The distribution of some of the **POS tags** (**nouns**, **verbs**, **adjectives** & **modifiers**) are significantly different for the target words compared to the other words.
- Fraction of certain **LIWC** & **Empath** categories across the target and the other words are significantly different.

4. Methodology

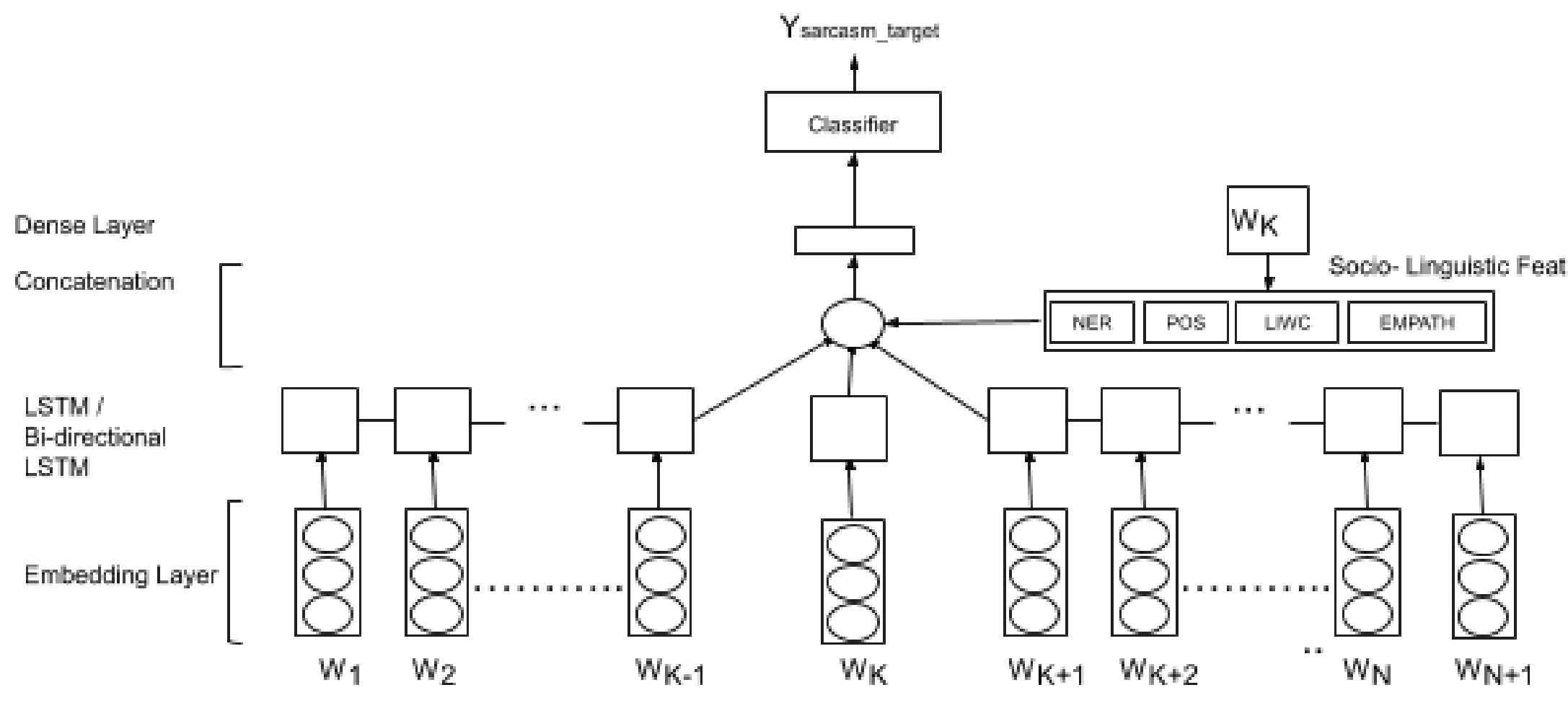

Figure 1:Architecture of the proposed system.

- The input to our system is a sarcastic text $[w_1, w_2 ... w_N]$ concatenated with a dummy word w_{N+1} at the end of the sentence.
- Start token and end token are appended at the beginning and the end of this sentence respectively. These two tokens are never considered as center word, but act as the left context for the first word w_1 and the right context for last dummy word w_{N+1} respectively.
- We proceed with the hypothesis that each word is a potential candidate to be a sarcasm target.
- Thus for each word in the sentence we create three components, (i) left context $[< start > w_1 : w_{K-1}]$, (ii) right context $[w_{K+1} : w_{N+1} < end >]$, and (iii) the center word w_K is to be classified as target or not.
- Each word in the left context, right context and the center word are passed through an embedding layer to initialize them through pre-trained embeddings.
- The word representations are then passed through an unidirectional LSTM (simple LSTM) or bidirectional LSTM (Bi-LSTM) layer or target dependent LSTM (TD-LSTM) layer.
- Next, the hidden vectors of the rightmost LSTM cell in left context, the central word LSTM cell hidden vector and the hidden vector of leftmost LSTM cell in right context are concatenated and passed to a dense layer.
- The dense representation is then concatenated with the socio-linguistic features that we obtained for the word w_K, and passed through a linear layer with sigmoid activation function, for the classification of the center word as sarcasm target or not.

5. Baseline

The baseline consists of two extractors joined by an integrator. The two extractors are rule based and statistics based. While the rule based extractor extracts candidate words for sarcasm target based on nine syntactic rules, the statistical extractor takes features such as lexical, POS tag, polarity, pragmatic features etc., and passes them to a classifier for the candidate word selection. The selected candidate words are then given as input to the integrator module, which is a hybrid 'AND' or 'OR' module, to select the final set of words as sarcasm targets.

6. Results

Model	EM_T	DS_T	EM_S	DS_S
Baseline: AND	13.45	20.82	16.51	21.28
Baseline: OR	9.09	39.63	7.01	32.68
LSTM layer	26.01	82.84	23.37	87.57
Bi-LSTM layer	29.48	84.04	30.14	87.66
TD-LSTM layer	26.35	82.27	25.97	87.71
Bi-LSTM layer + slf	**30.12**	**84.11**	**31.17**	**88.16**

The preceding table compares our models with the baseline. T: tweets, S: snippets, slf: socio-linguistic features Exact Match (EM), Dice Score (DS) over the training data for five-fold cross validation; $F1_{test}$ is the micro-F1 score over the test set.

6. Conclusion

- Presented a deep learning model for sarcasm target identification and outperformed the only available baseline by a large margin.
- Identified various socio-linguistic features that differentiate the target text from the rest of the snippet/tweet.
- When these additional socio-linguistic features are fused into our deep learning framework they seem to improve performance for both snippets and tweets establishing new a state-of-the-art for this problem.

References

[1] Aditya Joshi, Pranav Goel, Pushpak Bhattacharyya, and Mark Carman.
Sarcasm target identification: Dataset and an introductory approach.
In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, 2018.

Acknowledgements

We thank Microsoft Research India for travel support.

Contact Information

- **Email:** srijanbansal97@iitkgp.ac.in
- **Phone:** +91 7478 050 888
- **Code:** https://github.com/Srijanb97/ $Sarcasm_Target_Detection - EMNLPattention-$
- **CNeRG:** http://www.cnergres.iitkgp.ac.in/