SSN_NLP@IDAT-FIRE-2019 : Irony Detection in Arabic Tweets using Deep Learning and Features-based Approaches

S. Kayalvizhi, D. Thenmozhi, B. Senthil Kumar and Chandrabose Aravindan

Department of CSE, SSN College of Engineering, Chennai
{kayalvizhis,theni_d,senthil,aravindanc}@ssn.edu.in

Abstract. The disparity between the statement and its intended meaning is referred to as irony. Detecting this disparity in Arabic tweets is a challenging task. We presented three approaches namely deep learning using transformers, deep learning using Recurrent Neural Networks (RNN) and a features-based approach for detecting the irony in Arabic tweets. Among these approaches, the deep learning approach using transformers scores better F1-score of 0.816 than the deep learning approach using RNN which has 0.719 and the features-based approach which scores 0.709 on the IDAT@FIRE-2019 dataset.

Keywords: Deep neural network · Deep Learning · Irony Detection· LSTM · Transformers.

1 Introduction

Irony is a complex linguistic phenomenon widely studied in philosophy and linguistics. Irony can be defined as an incompatibility between the literal meaning and its conveyed meaning. Irony detection has gained relevance recently, due to its importance in various NLP applications such as sentiment analysis, hate speech detection, author profiling, fake news detection, and crisis management. The irony is detected in various language namely Italian [3], Czech [14], Spanish [13], French [7] and other languages [8]. Various methods have been used to detect irony in English tweets. The methods include SVM classifier [18], LSTM and word embeddings architecture [11], ensemble method of word-based and character based LSTM ensemble method of Logistic Regression(LR) [4] and SVM [15], etc. In Arabic, sarcasm, a special form of irony has been detected by creating a word cloud of tweets and classifying the words by weka tool [1] and deep neural networks [19] have also been used to classify the sarcasm. A survey has also been done on state of art of irony detection in Arabic language [16]. IDAT@FIRE-2019 [6] aims at detecting irony in Arabic tweets. Given a tweet, systems have to classify it as either ironic or not ironic.

2 Dataset Description

The dataset is from IDAT@FIRE-2019. The dataset includes short documents taken from Twitter related to different political issues and events related to the Middle East that hold during the years 2011 to 2018. The training set contains 4024 instances with two classes namely ironic and non-ironic. The test set contains 1006 instances.

3 Proposed Methodology

We propose two deep learning approaches and a features-based approach for detecting irony in Arabic tweets.

3.1 Deep learning approach using transformers architecture

In this method, Tensorflow based bi-directional transformers [5] are made use. Bi-directional Encoder Representation from Transformers (BERT) makes use of transformers. Transformer is an architecture that converts input sequence to output sequence, based on self-attention without using sequence RNN or convolution. Attention mechanism is the one that decides which part of the sentence is important. BERT uses masking mechanism in which 15% of input sequence are masked and then first those masked words by running the sequence.

In general, BERT predicts whether the given two sentence are adjacent sentences (i.e. Is the next sentence are not). The architecture includes the parameters such as number of layers, hidden nodes and attention heads. For binary classification, BERT is fine tuned by adding extra parameters to a classification layer W–>(K*H), where 'K' is the number of classifier labels and 'H' is the number of final hidden states which is at the top. The input data for the BERT model is prepared by removing the extra lines and special characters. The preprocessed data is given to the model for training. The output of the classifier for the test data is submitted as SSN_NLP Run 1.

3.2 Deep learning approach using Recurrent Neural Network architecture

In this method, sequence to sequence model [9, 10] is made use. In general, seq-to-seq model learns the long and short term dependencies. In our work, the model is used to predict the classes by learning the dependencies. For implementation we have utilized the Tensorflow based tutorial code by Neural Machine Translation[1].

Initially, the given training set is split into training and testing set and then analysed. During that analysis, training set had 3500 instances and testing had 524 instances. The tweets along with their classes are given to the model for

[1] https://github.com/tensorflow/nmt

training. The tweets are encoded into an intermediate representation which are then filtered by deciding whether they are important or not by attention and then decoded. Different models are constructed by varying the recurrent units and attention. In our work, two attentions namely Scaled Luong [9] and Normed Bahdanau[2] and recurrent units namely LSTM, GRU and GNMT are made use. Then, the class labels are predicted and projected as output by projection layer where loss is also calculated and reduced by back propagation. The result of analysis is shown in table below.

Recurrent Unit	**Attention**	**Accuracy**
LSTM	Scaled Luong	68.5
LSTM	Normed Bahdanau	72.3
GRU	**Scaled Luong**	**74.8**
GRU	Normed Bahdanau	70.99
GNMT	Scaled Luong	74.4
GNMT	Normed Bahdanau	74.6

Table 1. Comparison of different models

From the Table 1, it is clear that the model with GRU as recurrent unit and Scaled Luong attention performs well than the other architectures and it is submitted as SSN_NLP Run 2.

3.3 Features-based Approach

In this method, the full corpus given are pre-processed and vectorized using AraVec-1.0 [17]. It is an open source pre-trained word embedding for Arabic content. There are many models which vary by the number of vocabulary and techniques. We have used Twitter t-CBOW with 204,448 vocabulary size and 300 dimension for our work.

Classifiers	Accuracy		
	minimum	maximin	max-min
Multi-Layer Perceptron	**0.759**	0.733	0.746
Support Vector Classifier	0.732	0.712	0.693
Decision Tree	0.680	0.669	0.668
KNN	0.716	0.730	0.704
Random Forest	0.755	0.732	0.743
Adaboost	0.727	0.709	0.722
Quadratic DA	0.720	0.700	0.723
Gaussian NB	0.645	0.650	0.618

Table 2. Comparison of different machine learning classifiers

After vectorization, every tweet is represented as a single vector by either considering the minimum vector of 300 dimension, maximum of 300 dimension and minimum-maximum vectors of 600 dimension. Then the vectors are classified using different machine learning classifiers namely Multi-Layer Perceptron(MLP), Support Vector Classifier, Decision Tree, KNN, Random Forest, Adaboost and Quadratic DA.
Different traditional classifiers are trained by considering various forms of vector representation. Three vector representations namely minimum, maximum and minimum-maximum are considered whose results are shown in the Table 2. From the table, MLP with minimum vector seems to perform better than other classifiers with different vector representation. Thus, it was submitted as SSN_NLP Run 3.

4 Results

Approaches	**F1-score**
Deep learning approach using transformers architecture	**0.816**
Deep learning approach using Recurrent Neural Network architecture	**0.793**
Features-based Approach	**0.709**

Table 3. F1-scores of approaches for test data

Table 3 shows the F1-score of our approaches for the test data and Table 4 shows the score obtained by different teams. Three approaches namely deep learning using transformer architecture, deep learning using GRU as recurrent unit in neural network architecture with scaled luong attention and a features-based approach of using MLP are submitted as three runs SSN_NLP Run 1, SSN_NLP Run 2 and SSN_NLP Run 3 respectively. The first run seems to perform better than the other two approaches.

5 Error Analysis

Word embeddings have been used in the deep learning approach for detecting the irony in tweets. The significant phrases or words present in the tweets of ironic class may fall under the non-ironic class in the training set or vice versa, which may lead to misclassification [12]. In the features based approach, we have used pre-trained word embeddings. Due to out-of-vocabulary problem, many words have been skipped which may be a reason for misclassification.

6 Conclusions

Irony detection in Arabic tweets have gained a good relevance nowadays in which the incompatibility between the comment and considered statement can

Teams and Runs	F1-score
SSN_NLP Run 1	**0.816**
SSN_NLP Run 2	**0.793**
SSN_NLP Run 3	**0.709**
YOLO Run 1	0.844
chiyu_zhang_ubc Run 2	0.824
BENHA Run 3	0.821
rgcl Run1	0.818
Ali Allaith Run1	0.818
PITS Run1	0.807
Tha3aroon-ft Run1	0.794
kinmokusu Run1	0.695
Amrita_CEN Run1	0.687

Table 4. Final evaluation for Test Data

be found. We have presented three different approaches in which a deep learning approach using transformers have achieved 0.816 as F1-score for the test data which is better than deep learning approach of using LSTM and features-based approach of MLP classifiers, with 0.793 and 0.709 as F1-scores respectively. The performance can further be improved by changing the value of parameters in transformers architecture.The probability scores of different deep learning architectures and traditional classifiers can also be ensembled to improve the performance.

Acknowledgement

We would like to thank the Science and Engineering Research Board (SERB), Department of Science and Technology for funding the GPU system (EEQ/2018/000262) where this work is being carried out.

References

1. Al-Ghadhban, D., Alnkhilan, E., Tatwany, L., Alrazgan, M.: Arabic sarcasm detection in twitter. In: 2017 International Conference on Engineering & MIS (ICEMIS). pp. 1–7. IEEE (2017)
2. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473 (2014)
3. Barbieri, F., Ronzano, F., Saggion, H.: Italian irony detection in twitter: a first approach. In: The First Italian Conference on Computational Linguistics CLiC-it. vol. 28 (2014)
4. Baziotis, C., Nikolaos, A., Papalampidi, P., Kolovou, A., Paraskevopoulos, G., Ellinas, N., Potamianos, A.: NTUA-SLP at SemEval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive RNNs. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 613–621. Association for Computational Linguistics,,

New Orleans, Louisiana, (jun 2018). https://doi.org/10.18653/v1/S18-1100, https://www.aclweb.org/anthology/S18-1100.
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
6. Ghanem, B., Karoui, J., Benamara, F., Moriceau, V., Rosso, P.: IDAT@FIRE2019: Overview of the Track on Irony Detection in Arabic Tweets. In: Metha P., Rosso P., Majumder P., Mitra M. (Eds.) Working Notes of the Forum for Information Retrieval Evaluation (FIRE 2019). CEUR Workshop Proceedings. In: CEUR-WS.org, Kolkata, India, December 12-15 (2019)
7. González Fuente, S., Prieto Vives, P., Noveck, I.A.: A fine-grained analysis of the acoustic cues involved in verbal irony recognition in french. Barnes J, Brugos A, Shattuck-Hufnagel S, Veilleux N, editors. Speech Prosody 2016; 2016 May 31-June 3; Boston, United States of America.[place unknown]: International Speech Communication Association; 2016. p. 902-6. DOI: 10.21437/SpeechProsody. 2016-185 (2016)
8. Karoui, J., Zitoune, F.B., Moriceau, V.: Soukhria: Towards an irony detection system for arabic in social media. Procedia Computer Science **117**, 161–168 (2017)
9. Luong, M., Brevdo, E., Zhao, R.: Neural machine translation (seq2seq) tutorial. https://github.com/tensorflow/nmt (2017)
10. Luong, M.T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. arXiv preprint arXiv:1508.04025 (2015)
11. Mohammad, S.M., Bravo-Marquez, F.: Emotion intensities in tweets. arXiv preprint arXiv:1708.03696 (2017)
12. Nguyen, T.Q., Chiang, D.: Improving lexical choice in neural machine translation. arXiv preprint arXiv:1710.01329 (2017)
13. Ortega-Bueno, R., Rangel, F., Hernández Farıas, D., Rosso, P., Montes-y Gómez, M., Medina Pagola, J.E.: Overview of the task on irony detection in spanish variants. In: Proceedings of the Iberian Languages Evaluation Forum (IberLEF 2019), co-located with 34th Conference of the Spanish Society for Natural Language Processing (SEPLN 2019). CEUR-WS. org (2019)
14. Ptáček, T., Habernal, I., Hong, J.: Sarcasm detection on czech and english twitter. In: Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers. pp. 213–223 (2014)
15. Rohanian, O., Taslimipoor, S., Evans, R., Mitkov, R.: Wlv at semeval-2018 task 3: Dissecting tweets in search of irony. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 553–559 (2018)
16. Rosso, P., Rangel, F., Farías, I.H., Cagnina, L., Zaghouani, W., Charfi, A.: A survey on author profiling, deception, and irony detection for the arabic language. Language and Linguistics Compass **12**(4), e12275 (2018)
17. Soliman, A.B., Eissa, K., El-Beltagy, S.R.: Aravec: A set of arabic word embedding models for use in arabic nlp. Procedia Computer Science **117**, 256–265 (2017)
18. Wu, C., Wu, F., Wu, S., Liu, J., Yuan, Z., Huang, Y.: Thu_ngn at semeval-2018 task 3: Tweet irony detection with densely connected lstm and multi-task learning. In: Proceedings of The 12th International Workshop on Semantic Evaluation. pp. 51–56 (2018)
19. Xiong, T., Zhang, P., Zhu, H., Yang, Y.: Sarcasm detection with self-matching networks and low-rank bilinear pooling. In: The World Wide Web Conference. pp. 2115–2124. WWW '19, ACM, New York, USA (2019). https://doi.org/10.1145/3308558.3313735