Towards Improved Deep Contextual Embedding for the identification of Irony and Sarcasm

Usman Naseem*, Imran Razzak†, Peter Eklund‡ and Katarzyna Musial§
*§School of Computer Science, University of Technology Sydney, Australia
†‡School of Information Technology, Deakin University, Australia
Email:*usman.naseem@student.uts.edu.au,{†imran.razzak,‡peter.eklund}@deakin.edu.au, §katarzyna.musial-gabrys@uts.edu.au

***Abstract*—Humans use tonal stress and gestural cues to reveal negative feelings that are expressed ironically using positive or intensified positive words when communicating vocally. However, in textual data, like posts on social media, cues on sentiment valence are absent, thus making it challenging to identify the true meaning of utterances, even for the human reader. For a given post, an intelligent natural language processing system should be able to identify whether a post is ironic/sarcastic or not. Recent work confirms the difficulty of detecting sarcastic/ironic posts. To overcome challenges involved in the identification of sentiment valence, this paper presents the identification of irony and sarcasm in social media posts through transformer-based deep, intelligent contextual embedding – T-DICE – which improves noise within contexts. It solves the language ambiguities such as polysemy, semantics, syntax, and words sentiments by integrating embeddings. T-DICE is then forwarded to attention-based Bidirectional Long Short Term Memory (BiLSTM) to find out the sentiment of a post. We report the classification performance of the proposed network on benchmark datasets for #irony & #sarcasm. Results demonstrate that our approach outperforms existing state-of-the-art methods.**

***Index Terms*—Sarcasm; Irony, Twitter, Deep Contextual Embedding, Sentiment Analysis.**

I. Introduction

The explosive production of data via social media channels gives rise to the need for better solutions for the management, analysis and interpretation of semi-structured and unstructured big-data from social media. Social media challenges contemporary computational linguistic and text-analytic methods, in general toward the transformation of unstructured excerpts into some kind of structured data via the identification of embedded characteristics, such as emotional content [31]. In this context, the role of Sentiment Analysis (SA) comes into play, focusing on the development of efficient algorithmic processes for the automatic extraction of a writer's sentiment as conveyed in text excerpts. Relevant efforts focus on tracking the sentiment valence (or polarity) of single utterances, in the form of short text posts, which in most cases, are loaded with subjectivity and uncertainty [28].

Adding further complexity is that the forms of language used in social media and not normalised, utterances tend to breach vocabulary and grammar rules; they are unstructured, syntactically eccentric and often very informal. Users write using their own introduced words and jargon; they use abbreviations, non-standard punctuation, incorrect spelling, emoticons, slang words, idioms, abbreviations and often include URLs in posts. As with traditional text-analytics, utilising the context of the text in terms of polysemy, semantics, syntax, and having words sentiments, are crucial for detecting Figurative Language (FL): metaphors, similes, personification, hyperbole, and symbolism. The use of FL also intervenes to communicate the intent of the message. In such circumstances, the sentiment underlying the literal content conveyed may significantly differ from its figurative context, making sentiment identification even more confounding. Consider the following examples;

1) *It is wonderful feling to waste hrs in traffic jams!*
2) *You are a wonderful person*
3) *Oh how I luv being ignored*
4) *I love you*
5) *wow! flight delayed due to bad weather #badservice*
6) *I cudnt help myself.. I am a bad person...*

The examples indicate the discord between the actual sentiment valence and utterance content; they also emphasize the noisy and uncertain nature of social media. The contrast and shifting sentiment valence in sarcastic expression validates sarcasm as a unique instance of SA. Moreover, the absence of facial expressions, visual and tone-of-voice cues, means that context-aware approaches to leverage the ambiguity are ineffective. All of these language imperfections introduce noise, and one of the main tests is to handle this raw and informal text by employing useful pre-processing techniques [19]. In this paper, figurative expressions, especially *ironic or sarcastic* expressions, are considered as a way to express indirect meaning.

As shown in the above examples, words shown in **red** are those with inverse valence to dictionary word meaning and those in **green** are words whose valence follows dictionary word meaning. We see that the meaning of the words changes according to their context (considered a type of polysemy). In these cases, traditional methods are not able to capture meaning because they assign the same literal interpretation to a word regardless of its content and intention. Further, current word representation methods fail to capture words sentiments like "wonderful" and "bad", the use of which in content results in same word representations with opposite valences. Lastly, the unstructured and low-fidelity text, those words shown in **blue**, add to the out-of-vocabulary (OOV) word-list and reduce

978-1-7281-6926-2/20/$31.00 ©2020 IEEE

performance. Thus, ignoring polysemy induced from context and sentiment, based on the inverted valence of word meaning, and the injection of this new class of OOV words, naturally reduces the performance. However, directly applying state-of-the-art NLP models like Word2Vec [17], ELMo [22] and BERT [4] (and others) has limitations, such as the inability to identify and handle words with opposite valence (polysemy) and noise within the posts.

None of the models mentioned address all the language ambiguities earlier highlighted. Therefore, we propose an end-to-end methodology which is able to improve the quality of social-media text and efficiently handling language ambiguities defined earlier. Our research focuses on: (i) capture words with opposite valence; (ii) represent complex characteristics of word-usage, including both semantics and syntax; (iii) identify the sentiments of the words; and (iv) identify OOV words. We evaluate our system on several benchmark Twitter [27], [33] and Reddit datasets [11]. The performance for the identification of irony and sarcasm using the proposed method is better than other published state-of-the-art methods. The **key contributions** of this work are:

- a novel transformer-based deep intelligent contextual embedding (called T-DICE) that is able to capture words with opposite valence, semantics, syntax, words sentiments and OOV words within their context;
- an end-to-end methodology to capture deeper contextual word-relationships using an attention-based BiLSTM network for the detection of irony and sarcasm;
- extensive experimental results on different real-world datasets to assess and validate our method. These results show that the proposed model beats other published, state-of-the-art methods.

The remainder of the paper is structured as follows, Section II presents and summarises related work and prior art in the literature. Section III explains our proposed methodology and experimental results are presented in Section IV. Finally, conclusions are drawn in Section V.

II. Related Work

The detection of Figurative Language (FL), of which sarcasm and irony form a part, in social media platforms, has attracted much interest in the research community because of intrinsic problems related to differentiating them from the text that is sarcasm and irony-free. This recent trend breeds new challenges and opportunities for analysing text in order to detect sentiment and emotion [25]. Many attempts have been made targeting this problem, in particular, a challenge related to the valence detection of FL was presented at the SemEval workshop in 2015 and some graph based approaches were also presented in previous studies [29], [30]. Furthermore, similar challenges for different languages were also conducted, which indicates the growing interest in the detection of FL across the NLP community.

In a study conducted by Reyes et al. [26], the authors attempt to detect irony and sarcasm where they proposed concepts of unexpectedness and contradiction that frequently appear in FL. The concept of unexpectedness was also adopted in other studies, such as Barbieri et al. [2], where sarcastic Tweets were compared with other topics, and the measure of unexpectedness is measured using RF and DT classifiers. Similarly, Ghosh et al. [5], used a measure of unexpectedness to calculate the emotional imbalance between words in the text, and contradictions identified within Tweets, as a feature within a SVM.

In follow-up studies, researchers focused on context and context-based methods which employed features that reveal information about the content. Gonzlez-Ibez et al. [6] utilised a LIWC corpus and compared the Tweets with sarcasm – with positive and negatives examples – using an SVM-classifier. Farías et al. [8] utilise effective and structural features to predict FL language with traditional machine learning classifiers. A combination of lexical, semantic and syntactic features as input to an SVM-classifier for the detection of FL, which achieved a significant result [33]. Capturing global context proved to be more useful than only capturing local context for the detection of FL.

Deep-learning (DL) based architectures have gained enormous popularity in different NLP tasks since they are able to handle data sparseness in unbalanced datasets. Kumar et al. [13] combined word embeddings and convolutional neural networks (CNNs) for detecting sarcasm. Ghosh et al. [5] built a model combining a CNN with a recurrent neural network (RNN) followed by a deep neural network. It is noticed that this method demonstrates improvement. In separate studies conducted by Huang et al. [9] and Zhang et al. [35], attentive RNNs are employed with pre-trained Word2vec embeddings and contextual information for FL detection. Tay et al. [32] presented an LSTM-based intra-attention method that achieved excellent performance. Using pre-trained embeddings along with user representations structured from previous posts [1] and personality representations, passed through CNNs (CASCADE) [7] have also been employed by researchers for sarcasm detection in on-line discussion forums. CASCADE performs context and content-driven sarcasm detection on social media. It also takes into account the variation in the nature of sarcasm and its use from person-to-person. To address this issue, the study uses user embeddings and personality traits along with content-based features extracted using a CNN [16] using multi-tasking learning for sentiment and sarcasm classification. Ilic et al. [10] employed Embeddings from language model (ELMo) [22] in their study for the detection of irony and sarcasm. Different hybrid models such as DICE [19], DICE+ [18], hybrid contextual word representation [20] and transformer based Deep Intelligent Contextual Embedding [21] were also presented to capture complex language ambiguities in previous studies. Recently, Potamias et al. [23] presented an ensemble deep-learning (DESC) model, which can capture content and semantic information. The authors employed an extensive feature set from content along with TF-IDF features. In addition, a BiLSTM with attention model was trained with pre-trained GloVe to structure an ensemble classifier processing different text representations. In extended work,

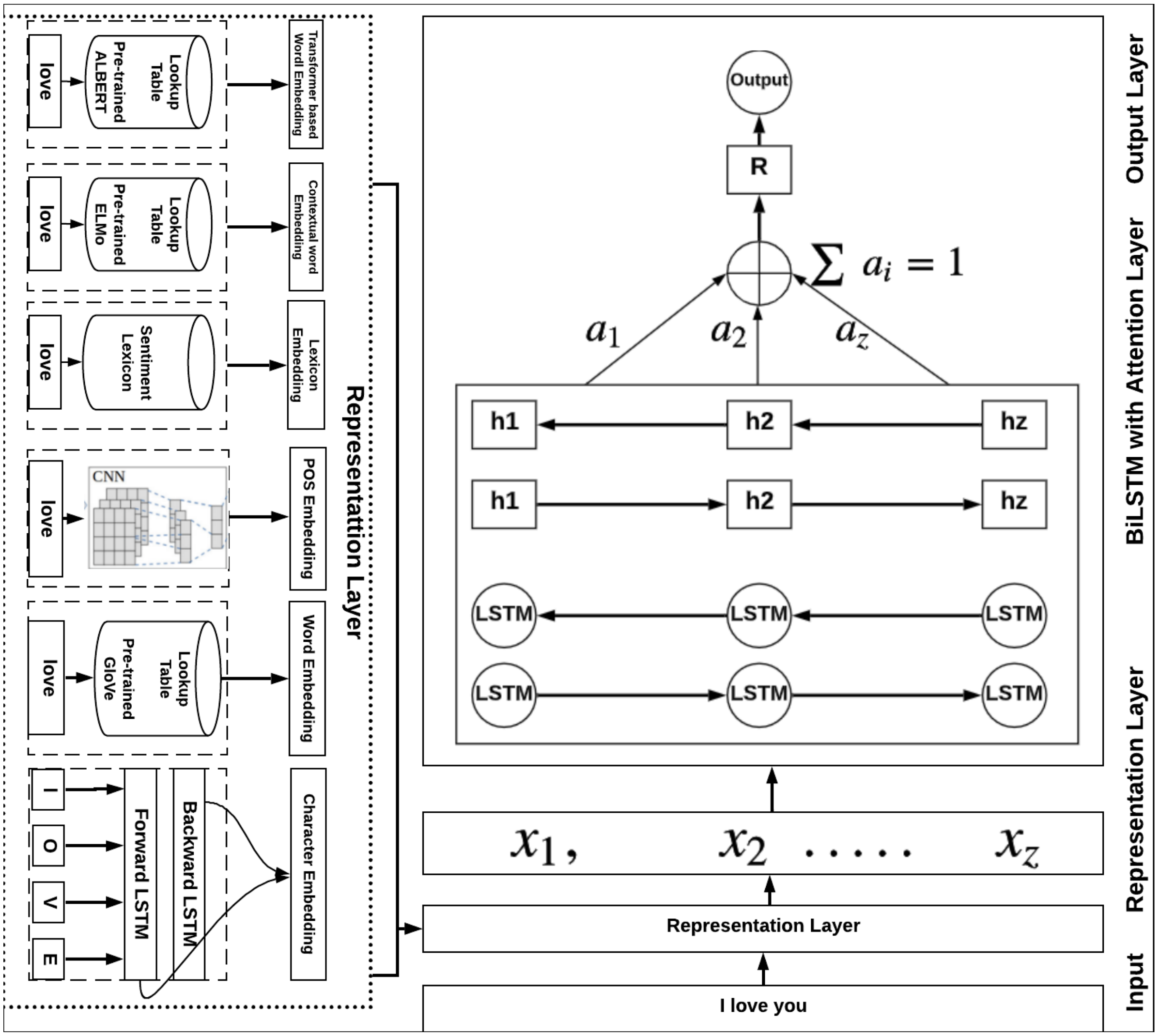

Fig. 1. Overall architecture of proposed model.

Potamias et al. [24] presented a transformer-based approach for the detection of FL, such as irony and sarcasm. In that work, the authors leverage the pre-trained RoBERTa model combined with a recurrent CNN (RCNN) to tag FL in social media. Contrast to previously mentioned literature, our proposed model is context-sensitive, which considers complex characteristics of words for the detection of irony and sarcasm posts in social media.

III. Proposed Model

In this section, we present the proposed Transformer based Deep Intelligent contextual embedding (T-DICE) with attention-based BiLSTM. The model is based on (a) Representation layer and; (b) Bidirectional Long Short Term Memory (BiLSTM) with attention layer. Our model's framework includes four components: Input (Corpus), Representation layer, BiLSTM with Attention and output layer. The overall architecture of the proposed model is given in Fig. 1. In the following discussion, we describe each of these components in detail.

A. Representational Layer

Given a **Post** T_i with a sequence of tokens $(t_1, t_2, t_3, ...t_k)$, i represents the number of a **Post** and k depicts the number of tokens in a **Post**. The aim is to identify whether the post is ironic/sarcastic or non-ironic/sarcastic. To do so, we fuse the text representations from transformers such as (ALBERT), contextual word representations (ELMo), traditional word representations (GloVe), Part-of-speech word representations, Lexicon word representation and Character representations at the representational layer. As a result, the representational layer is powerful enough to cover, not only context-level, but also word and character-level aspects of a post, and can thus handle the language complexities, being able to capture complex language attributes such as same words with different meanings (polysemous words), semantics, word sentiment and syntactical information. Detail of representation layer is given in the following sections.

*1) A Lite **BERT** (ALBERT):* Lan et al. [15] proposed the ALBERT model, a modified, improved variant of BERT (Bidirectional Encoder Representations from Transformers). ALBERT consolidates two-parameter reduction methods that relieve the significant hurdles in mounting pre-trained models. Similar to BERT-large, ALBERT has 18 times fewer parameters and is 1.7 times faster in training. The parameter reduction techniques also act as a form of regularization that stabilizes the training and helps with generalization. Our architecture is based on the pre-trained ALBERT model[1], trained on Bookcorpus and Wikipedia for 125,000 steps. ALBERT captures context information from both directions and gives us the transformer-based representation of the text.

2) Embedding from Language Model (ELMo): ELMo [22] considers various aspects of words according to its context. The use of context-based word embedding helps with polysemy and also to capture words with different sentiment valence, and are therefore better when dealing with sarcastic texts. These embeddings are formed on the representation from Bi-language model (BiLM). Unlike traditional word embeddings, ELMo deals with various prospects of words according to their usage in context, thus, in this work, we have used ELMo to identify the same words but with different meanings in a post. Our architecture is based on pre-trained ELMo embeddings[2]. ELMo gives us word representation which is able to capture words with different sentiment valence, syntax information of the context, assign them different vectors and is given by eqn. 1.

$$\mathit{ELMo}_n^{task} = E(M_n; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} h_{h,j}^{LM} \tag{1}$$

where s^{task} are weights which are softmax normalized for the combination of different layer representations and γ^{task} is a hyper-parameter for optimization and scaling of the Elmo representation.

3) GloVe Embedding: Concatenating ELMo word representations with conventional words representation models such as Word2Vec and GloVe has been shown to achieve better performance [22] in determining sentiment valence. Thus in our proposed methodology, we have utilized GloVe[3] pre-trained embeddings of 300 dimensions, this achieves good outcomes as compared to Word2Vec. GloVe outputs a vector, which consists of the semantics information.

4) Part-of-Speech (POS) Embedding: POS tagging is an effective step in which every word is allocated with a suitable POS tag. Utilizing POS tags has demonstrated positive results. POS tagging provides information about a word; it's adjoin and the different part-of-speech types of words. In this work we used Stanford parser for POS[4] tagging which generates POS tags. Each POS tagged token is passed through CNN proposed by Kim et al. [12] to capture syntactical information of words.

5) Lexicon Embedding: The lexicon-based word representations extract sentiment scores from sentiment specific lexicons. Each lexicon contains a pair of the word and its associated sentiment where each word has its sentiment score. There are plenty of lexicons available, so it is crucial to chose an appropriate (one or a combination of multiple) lexicons. We used a combination of lexicons, used in our previous work [18]. A score of zero is assigned to a word not available in any of lexicon. Our lexicon-based word representation gives a vector which captures the sentiment of words.

6) Character Embedding: To have deeper representation between words of a similar type, the suffix and prefix information of any words gives character level features. This helps to address the challenge of out-of-vocabulary words but also mitigates against previously unseen words. In this paper, we have implemented character-level representations using BiLSTMs to generate a character-enhanced embedding for every word in a post [14]. The maximum length of character was considered in our experiments is 25 and set both LSTMs (forward and backward) to 25, this gives in 50-dimensional vector useful to capture and handle OOV issues.

Eventually, we concatenate the above vectors to obtain one vector which is able to address the language ambiguities mentioned earlier.

B. BiLSTM with Attention layer

A BiLSTM layer is placed on top of the representation layer with the attention layer to gather information from both directions. An input vector with a sequence of x_z tokens is forwarded to BiLSTM which outputs a hidden representation h_i at a given time i, by combining the hidden representations from both forward $\overrightarrow{h_i}$ and backward $\overleftarrow{h_i}$ LSTM, as given by eqn. (2).

$$h_i = [\overrightarrow{h_i} \parallel \overleftarrow{h_i}] \tag{2}$$

where $\parallel$ represents the combination of outputs from both (forward and backward) LSTMS.

Not all words contribute equally to understand the meaning of a sentence. We have utilized an attention mechanism to reinforce the contribution of important words. We have allocated a weight a_i to every token through a softmax function, and finally, a representation R which is the weighted sum of all tokens is computed as shown in eqn. (3).

$$R = \sum_{i=1}^{z} a_i h_i, \tag{3}$$

where,

$$a_i = \frac{\exp(e_i)}{\sum_{t=1}^{z} \exp(e_t)}, \quad \sum_{i=1}^{z} a_i = 1$$

[1] ttps://github.com/google-research/ALBERThttps://github.com/google-research/ALBERT

[2] https://tfhub.dev/google/elmo/3

[3] https://nlp.stanford.edu/projects/glove/

[4] https://nlp.stanford.edu/software/tagger.shtml

TABLE I
DATASET DISTRIBUTION

Dataset	Total	Train	Valid	Test	Source
Twitter SemEval-2018-Task 3	3,834	3,067	306	784	Twitter
SARC 2.0 politics	17,074	10,934	2,734	3,406	Twitter
Riloff Sarcastic Dataset	1,897	3,284	469	939	Reditt

$$e_i = \tanh(W_h h_i + b_h)$$

where W_h and b_h are learned parameters, h_i is the concatenation of the representations of both (forward and backward) LSTM, introduced in eqn. (3).

C. Output layer

Representation R created from the attention layer and forwarded to the softmax layer to get the class probability distribution. We minimized the binary cross-entropy loss function L in which loss increases as the predicted probability p diverges from the actual label y, given by eqn. (4).

$$L = -(y\log(p) + (1-y)\log(1-p)) \tag{4}$$

IV. EXPERIMENTAL ANALYSIS

In this section, we first present the experimental settings, the datasets used in this study, and the experimental evaluations to show the effectiveness of our proposed model. For preprocessing, we used *Ekphrasis*[5] tool improve the quality of text. A 10-fold cross-validation technique is used to evaluate the classification results. Accuracy and F1-Score are used for the evaluation of our proposed model.

A. Datasets

- **Twitter SemEval-2018-Task 3**: Our first dataset is a Twitter dataset which was provided for the SemEval 2018 (Task 3)- Irony Detection in English Tweets [33]. The dataset is manually annotated using binary labels.
- **Self-Annotated Reddit Corpus (SARC)**: The Self-Annotated Reddit Corpus (SARC) was introduced by Khodak et al. . [11] in 2013. It contains more than a million sarcastic and non-sarcastic statements retrieved from Reddit, with some contextual information, such as author details, score and parent comment (corresponding to the quote text in Sarcasm V2[6]). In our experiments, we use only their large dataset containing political comments from Reddit.
- **Riloff Sarcastic Dataset:** Lastly, We also use the dataset by Riloff et al. [27], which is manually annotated for sarcasm. Table I shows the distribution of each dataset.

B. Parameters

In order to avoid the overfitting, we use zero-centred Gaussian noise ($\sigma = 0.3$) at our input layer. Also, a dropout (0.25) at connections of the network is used to turn off the neurons in the network randomly. Moreover, to avoid over-fitting and make our method robust, we applied the ($L_2 = 0.0001$) regularization technique to decrease large weights. The dimension of each hidden layer used is 150.

To train the model, we used the rectified linear unit (ReLu) with a batch size of 128. To tune the learning rate (0.001), we used the Adam optimizer. For optimization, we utilized the grid search technique to find hyper-parameters.

TABLE II
COMPARISON OF THE PROPOSED METHOD:IRONY/SEMVAL DATASET

Model\Dataset	Irony/SemVal-2018-Task 3.A	
	Accuracy	F1-Score
ELMo [24]	0.66	0.66
USE [24]	0.69	0.67
NBSVM [24]	0.69	0.69
FastText [24]	0.69	0.69
Xlnet [24]	0.71	0.70
Bert-Cased [24]	0.70	0.69
BERT-Uncased [24]	0.69	0.68
Roberta [24]	0.79	0.78
Illi et al. [10]	0.71	0.70
THU_NGN [34]	0.73	0.71
NTUA-SLP [3]	0.73	0.67
DESC [23]	0.74	0.73
Zhang et al. [35]	-	0.71
Potamias et al. [24]	0.82	0.80
Proposed	**0.84**	**0.82**
Δ Compared to Previous best	**2.43%**	**2.50%**

TABLE III
COMPARISON OF THE PROPOSED METHOD:SARC2.0 POLITICS DATASET

Model\Dataset	Reddit SARC2.0 politics	
	Accuracy	F1-Score
ELMo [24]	0.70	0.70
USE [24]	0.75	0.75
NBSVM [24]	0.65	0.65
FastText [24]	0.63	0.63
Xlnet [24]	0.76	0.76
Bert-Cased [24]	0.76	0.76
BERT-Uncased [24]	0.77	0.77
Roberta [24]	0.77	0.77
Illi et al. [10]	0.79	-
CASCADE [7]	0.74	0.75
Khodak et al. [11]	0.77	-
Potamias et al. [24]	0.79	0.78
Proposed	**0.81**	**0.80**
Δ Compared to Previous best	**2.53%**	**2.56%**

C. Experimental Results

This section presents the baseline models used to compare our proposed model and discusses the results. To evaluate the performance of the proposed model, we conduct a comprehensive comparison with several advanced state-of-the-art methodologies along with published results.

The results are summarized in the Tables II–IV. The table presents the results of our proposed method (Proposed) and contrasts them to other state-of-the-art baseline methodologies along with published results using the same datasets. As can be observed, the proposed model outperforms all approaches as well as all methods with published results. Our model

[5] https://github.com/cbaziotis/ekphrasis

[6] https://nlds.soe.ucsc.edu/sarcasm2

TABLE IV
COMPARISON OF THE PROPOSED METHOD: RILOFF SARCASTIC DATASET

Model\Dataset	Riloff Sarcastic Dataset	
	Accuracy	F1-Score
ELMo [24]	0.85	0.85
USE [24]	0.87	0.78
NBSVM [24]	0.75	0.58
FastText [24]	0.83	0.64
Xlnet [24]	0.86	0.86
Bert-Cased [24]	0.86	0.86
BERT-Uncased [24]	0.87	0.87
Roberta [24]	0.89	0.85
Illi et al. [10]	0.86	0.75
DESC [23]	0.87	0.86
Farias et al. [8]	-	0.75
Tay et al. [32]	0.82	0.73
Ghosh & Veale [5]	-	0.88
Potamias et al. [24]	0.91	0.91
Proposed	**0.93**	**0.93**
△ Compared to Previous best	**2.19%**	**2.19%**

achieves better performance because it improves the quality of Tweets by handling noise, polysemy, semantics, syntax and sentiment within Tweet context. Our proposed model attained good results because the poor quality of tweets is improved which helps to learn a good representation of text and our proposed method addressed the language attributes which helps to capture deeper characteristics and relationships. Our method learns text representations of good quality by including polysemous, semantics, semantics, OOV words, words sentiments, syntactical information as well as it learns high-level representations from transformer-based BERT (ALBERT) and this helps achieve better classification results. Our model outperforms the results of the previous best-model (underlined) by 2.43%, 2.53% and 2.19% for Twitter SemEval-2018, SARC 2.0 politics and Riloff Sarcast datasets respectively when the evaluation metric is *Accuracy* and 2.50%, 2.56% and 2.19%, when the metric is *F1-Score*, for each dataset respectively. As our model gives a constant improvement over other models for all datasets, we can deduce that it is a robust solution for the detection of irony and sarcasm.

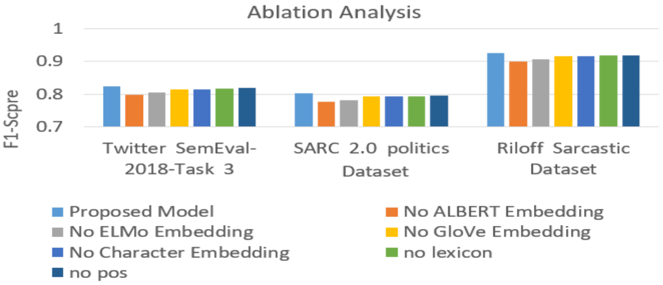

Fig. 2. Ablation analysis.

D. Ablation Analysis

It is clear from Fig. 2 that all embeddings from representation layer in our proposed model adds to the overall system performance. Performance drops slightly in each case when we remove GloVe, Character, Lexicon or POS embeddings for all datasets from the proposed model. Further, the empirical analysis also illustrates that performance deteriorates when we remove the ELMo embedding from the proposed model and notable drop is observed in the case where we remove ALBERT from the model. Hence, we can deduce that the strengths of the proposed model lie in the consolidation of different components that contribute to the diversity of the model and its improved performance.

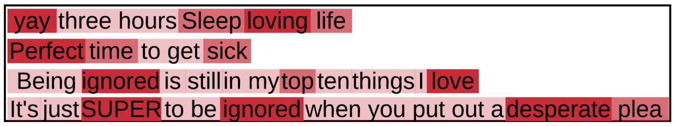
yay three hours Sleep loving life
Perfect time to get sick
Being ignored is still in my top ten things I love
It's just SUPER to be ignored when you put out a desperate plea

Fig. 3. Attention heat map visualization.

E. Visualization

An attention heat-map visualization is shown in Fig. 3. The colour intensity of each word corresponds to its weight (importance), ascribed by the attention mechanism. The word-cloud of most common words in a) Irony/Sarcasm posts and b) non-Irony/Sarcasm posts are shown in Fig. 4.

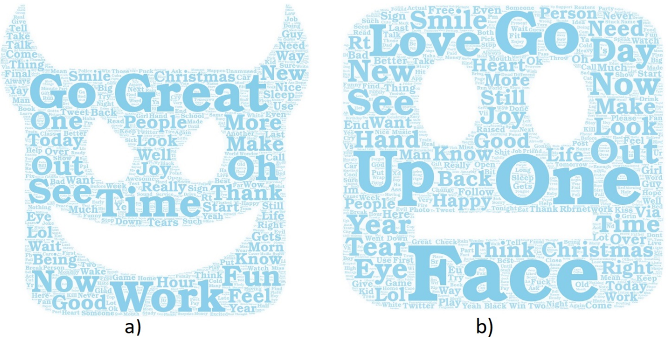

Fig. 4. Word Cloud a) Irony/Sarcasm posts b) non-Irony/Sarcasm posts

V. CONCLUSION

In this study, we present the first transformer-based deep, intelligent contextual embedding (T-DICE) end-to-end methodology, leveraging the pre-trained ALBERT-model, combined with other embeddings and attention-based Bilstm, to deal with figurative language (Irony/Sarcasm) in social media. Our network is compared with all, to the best of our knowledge, published methodologies using three different benchmark datasets. The proposed methodology handles language ambiguities such as polysemy, semantics, syntax and words sentiment within the context of the post by learning representations from six different embeddings. Additionally, attention-based BiLSTM adds to the performance by capturing important words. Our experiment demonstrates that the proposed model outperforms several baselines and achieves state-of-the-art performance for the detection of figurative language.

References

[1] Silvio Amir, Byron C. Wallace, Hao Lyu, and Paula Carvalho Mário J. Silva. Modelling context with user embeddings for sarcasm detection in social media, 2016.

[2] Francesco Barbieri and Horacio Saggion. Modelling irony in twitter. In *Proceedings of the Student Research Workshop at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 56–64, 2014.

[3] Christos Baziotis, Nikos Athanasiou, Pinelopi Papalampidi, Athanasia Kolovou, Georgios Paraskevopoulos, Nikolaos Ellinas, and Alexandros Potamianos. Ntua-slp at semeval-2018 task 3: Tracking ironic tweets using ensembles of word and character level attentive rnns, 2018.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[5] Aniruddha Ghosh and Tony Veale. Fracking sarcasm using neural network. In *Proceedings of the 7th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 161–169, San Diego, California, June 2016. Association for Computational Linguistics.

[6] Roberto González-Ibáñez, Smaranda Muresan, and Nina Wacholder. Identifying sarcasm in twitter: A closer look. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 581–586, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[7] Devamanyu Hazarika, Soujanya Poria, Sruthi Gorantla, Erik Cambria, Roger Zimmermann, and Rada Mihalcea. Cascade: Contextual sarcasm detection in online discussion forums, 2018.

[8] D.I. Hernández Farías, M. Montes-y Gómez, H.J. Escalante, P. Rosso, and V. Patti. A knowledge-based weighted knn for detecting irony in twitter. *Lecture Notes in Computer Science*, 11289 LNAI:194–206, 2018. cited By 0.

[9] Yu-Hsiang Huang, Hen-Hsen Huang, and Hsin-Hsi Chen. Irony detection with attentive recurrent neural networks. In *ECIR*, 2017.

[10] Suzana Ilić, Edison Marrese-Taylor, Jorge A. Balazs, and Yutaka Matsuo. Deep contextualized word representations for detecting sarcasm and irony, 2018.

[11] Mikhail Khodak, Nikunj Saunshi, and Kiran Vodrahalli. A large self-annotated corpus for sarcasm, 2017.

[12] Yoon Kim. Convolutional neural networks for sentence classification. In *EMNLP*, 2014.

[13] Lakshya Kumar, Arpan Somani, and Pushpak Bhattacharyya. "having 2 hours to write a paper is fun!": Detecting sarcasm in numerical portions of text. *ArXiv*, abs/1709.01950, 2017.

[14] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *CoRR*, abs/1603.01360, 2016.

[15] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations, 2019.

[16] Navonil Majumder, Soujanya Poria, Haiyun Peng, Niyati Chhaya, Erik Cambria, and Alexander Gelbukh. Sentiment and sarcasm classification with multitask learning, 2019.

[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–9, 2013.

[18] Usman Naseem, Shah Khalid Khan, Imran Razzak, and Ibrahim A. Hameed. Hybrid words representation for airlines sentiment analysis. In Jixue Liu and James Bailey, editors, *AI 2019: Advances in Artificial Intelligence*, pages 381–392, Cham, 2019. Springer International Publishing.

[19] Usman Naseem and Katarzyna Musial. Dice: Deep intelligent contextual embedding for twitter sentiment analysis. *2019 15th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1–5, 2019.

[20] Usman Naseem, Imran Razzak, and Ibrahim A Hameed. Deep context-aware embedding for abusive and hate speech detection on twitter. *Australian Journal of Intelligent Information Processing Systems*, page 69.

[21] Usman Naseem, Imran Razzak, Katarzyna Musial, and Muhammad Imran. 2020.

[22] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *CoRR*, abs/1802.05365, 2018.

[23] Rolandos Alexandros Potamias, Georgios Siolas, and Andreas Stafylopatis. A robust deep ensemble classifier for figurative language detection. In *EANN*, 2019.

[24] Rolandos Alexandros Potamias, Georgios Siolas, and Andreas Georgios Stafylopatis. A transformer-based approach to irony and sarcasm detection, 2019.

[25] Diego Recupero and Erik Cambria. Eswc'14 challenge on concept-level sentiment analysis. volume 475, pages 3–20, 05 2014.

[26] Antonio Reyes, Paolo Rosso, and Davide Buscaldi. From humor recognition to irony detection: The figurative language of social media. *Data Knowl. Eng.*, 74:1–12, April 2012.

[27] Ellen Riloff, Ashequl Qadir, Prafulla Surve, Lalindra De Silva, Nathan Gilbert, and Ruihong Huang. Sarcasm as contrast between a positive sentiment and negative situation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 704–714, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

[28] Zafar Saeed, Rabeeh Ayaz Abbasi, Onaiza Maqbool, Abida Sadaf, Imran Razzak, Ali Daud, Naif Radi Aljohani, and Guandong Xu. What's happening around the world? a survey and framework on event detection techniques on twitter. *Journal of Grid Computing*, pages 1–34, 2019.

[29] Zafar Saeed, Rabeeh Ayaz Abbasi, Imran Razzak, Onaiza Maqbool, Abida Sadaf, and Guandong Xu. Enhanced heartbeat graph for emerging event detection on twitter using time series networks. *Expert Systems with Applications*, 2019.

[30] Zafar Saeed, Rabeeh Ayaz Abbasi, Muhammad Imran Razzak, and Guandong Xu. Event detection in twitter stream using weighted dynamic heartbeat graph approach. *arXiv preprint arXiv:1902.08522*, 2019.

[31] Zafar Saeed, Rabeeh Ayaz Abbasi, Abida Sadaf, Muhammad Imran Razzak, and Guandong Xu. Text stream to temporal network-a dynamic heartbeat graph to detect emerging events on twitter. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 534–545. Springer, 2018.

[32] Yi Tay, Anh Tuan Luu, Siu Cheung Hui, and Jian Su. Reasoning with sarcasm by reading in-between. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1010–1020, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[33] Cynthia Van Hee, Els Lefever, and Véronique Hoste. Exploring the fine-grained analysis and automatic detection of irony on twitter. *Language Resources and Evaluation*, 52(3):707–731, Sep 2018.

[34] Chuhan Wu, Fangzhao Wu, Sixing Wu, Junxin Liu, Zhigang Yuan, and Yongfeng Huang. 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*, pages 51–56, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[35] Shiwei Zhang, Xiuzhen Zhang, Jeffrey Chan, and Paolo Rosso. Irony detection via sentiment-based transfer learning. *Information Processing Management*, 56(5):1633 – 1644, 2019.