JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

SARCASM DETECTION IN TEXT USING DEEP LEARNING NETWORKS

[1]Mushafaq Ashai, [2]Kamal Kumar
[1]Mtech CSE, [2]HOD & Assistant Professor
[1]Kurukshetra University, Kurukshetra, India
[2]Swami Devi Dyal Institute of Engineering and Technology, Kurukshetra University, India

Abstract: Textual sentiment or opinion analysis systems mine textual data to identify personal feelings or views about a particular item or event. However, if sarcastic features of conversation are not taken into account, then these systems may be biased. As a result, sarcasm detection in textual communication is important for these systems' performance. Several research have used numerous methods to identify sarcasm in text, but all lack a critical component of any textual form of communication: context and semantics. The context and semantics are captured using BERT Model. Then, the classifiers are subsequently trained using these rich context and semantic embeddings. Using two datasets, we compared our system to state-of-the-art systems and found that BERT had higher F1-score, recall, and precision. As a result, we conclude that incorporating contextual and semantic data into sarcastic classifiers increases their overall performance.

Index Terms – Sarcasm detection, Classification, Semantic, BERT, embeddings, Word2Vec

1. INTRODUCTION

Sarcasm is a mocking communicative utterance of phrases and words that are employed to flip the polarity of positive sentences to negative or vice versa. Usage of sarcastic textual information has increased dramatically because of the increase in online social media usage. Sarcasm detection is vital for many Natural Language Processing (NLP) tasks like sentiment analysis, emotion detection, and opinion mining. Without recognizing sarcasm, the actual meaning cannot be interpreted and hence the discourse's, sentiment and emotions are not identified.

Sarcasm detection task is efficient in many ways viz: News market, suicide management, E-Learning, Recommendation systems (Ecommerce shopping) and many more. Detecting sarcasm is a classification task [24]. This paper talks about designing a sarcasm classifier capable of classifying text into sarcastic and non-sarcastic classes by using contextual and semantic model. The work is to find if the sentence is sarcastic or not. Sarcasm detection approaches are categorized into three types [24]: rule-based, machine learning based and statistical-based. In the rule-based system, if someone replies negatively in a positive context, we classify the text as sarcastic. In the deep learning-based approaches, the features are extracted by the deep learning classifiers, and by employing these extracted features, the system classifies text as sarcastic and non-sarcastic. In the statistical settings, features like uni-grams, part-of-speech, adjectives are identified using feature engineering. These features are passed to the machine learning classifiers to classify the text. All these methods work well but lack a critical aspect of any textual form of communication: the semantics and context in which the communication is taking place. Several studies used semantics and context as separate features for sarcasm detection and achieved comparable results. This paper presents a novel approach where we capture both semantic and contextual features for sarcasm classification and report the results. Our test will be on certain datasets approached for detecting sarcasm and then result has to be compared with the baseline. Our results confirm that using both semantics and contextual features combined produces better results.

To capture semantics, we use BERT along with the sentiment lexicons to obtain semantic extensions of the text. Sentiment lexicons contain sentiment describing terms, and we probe for these terms in the text. If it contains the sentiment defining term, we obtain its semantic extension by our novel algorithm that employs BERT. Thus, for all sentences that possess sentiment, we obtain its extension in semantic space.

With the advances in NLP, particularly with transformers, the NLP researchers are using them in various studies, and their usage has outperformed the existing work. Transformers work by learning from the colossal amount of data. The transformer architectures allow us to get embedded vectors that are rich and precise and help to obtain context. BERT (Bidirectional Encoder Representations from Transformers) [10]is the latest transformer model successfully implemented in various NLP tasks. BERT is pre-trained on the large scale of text data and provides contextually rich word embeddings. We use BERT to capture context and produce contextually rich embeddings for sentences in the dataset.

Thus, we use BERT algorithms to obtain both the semantics and the context to generate our novel sarcasm classifier. Then these embeddings are employed to train different machine learning classifiers.

This research aims to create a sarcasm classifier that captures both the context and semantics of the text. The classifier is designed using a hybrid scheme employing both deep learning and machine learning techniques. We obtain BERT embeddings using the BERT and feed these embeddings to various machine learning classifiers. The idea is to add context and semantics to the machine learning-based classification. BERT embeddings provide context. Moreover, BERT adds semantics to the text. These semantically and contextually rich embeddings then are feed to the machine learning-based classifiers. We compare our system with the BERT as a baseline and found improved results. The results are compared with word2Vec Model.

In all the three experiments, we use the same BERT models for the baseline and in our system with the same parameters. The result show that using both context and semantics for sarcasm detection aids in sarcasm classification.

The remainder of the paper is structured as follows: Section 2 describes literature in sarcasm detection; section 3 discusses the proposed methodology; section 4 presents the experiments and results obtained; section 5 presents discussion and section 6 provides conclusions and future scope.

2. LITERATURE SURVEY

According to literature, there are four types of sarcasm: (1) Propositional sarcasm: It is proportional when the statement is simple, but it has an implicit sentiment of sarcastic nature for example," you are an excellent human!". For this sentence if context as well as meaning is not known, then it may look sarcastic (2) Embedded: These sentences have sarcastic utterances embedded within it. They usually depict implicit sarcasm. (3) Like-prefixed: These types of sarcastic sentences are prefixed with the word" like", for example," Like you are the best teacher in the world!". (4) Illocutionary: It does not contain textual clues .It contains examples such as making faces or other gestures.

In this section, we discuss the state-of-art of the sarcasm detection field: we start with linguistic theories for sarcasm detection, followed by datasets used and then, the approaches used and the reported performance of various studies. We highlight the limitations and advantages of each aspect, intending to obtain conclusions and determine pending issues

2.1 Linguistic theories of sarcasm detection:

Since Grice's theory of pragmatics [21], several theories are developed by linguistics. Few of these theories are explained here:

2.1.1 Echoic mention theory: Sperber and Wilson (1981) proposed this verbal sarcasm theory that deals with direct utterances of sarcasm in the communicated text. It states that a literal propositional is not always taken as intended through the communicated sentences. The following sentence clear the meaning of this theory;" I do not watch the football matches every day, and well I am alive and Kicking". The speaker wants to convey that watching football does not have any effect on entertainment.

2.1.2 Echoic remainder theory: This theory was proposed by Roger and Sam Glucksberg in 1989[27]. As the name suggests, this theory reminds the listener that such events have happened in the past, and their usage in the sentences elicit the sarcasm.

2.1.3 Sarcasm as a dropped negation: This theory proposed by Rachel Giora in 1995 [18] states that the sarcastic sentences do not have explicit negation markers which can be converted into non-sarcastic sentences. For example, the sentence." Watching Netflix at midnight with a headache is serious fun", is proportionate to its non-sarcastic sentence" Watching Netflix at midnight with a headache is not serious fun."

2.1.4 Tuple-representation of sarcasm: [21] represents the sarcasm as 6-tuple. The tuple has the form T= < Sp,Hr,Co,Ut,LP,IP> where Sp is the speaker in the conversation; Hr is the listener in the conversation; Co is the speaker's context; Ut is the uttered sentences, LP is the literal meaning of the speaker's sentences, and IP is the intended meaning of the sentences of the speaker.

2.2 Approaches of Automatic Sarcasm Detection:
The automatic sarcasm detection task is mainly categorized into three approaches: rule-based, deep-learning, and machine learning-based approaches.

2.2.1 Rule-based systems [31] uses rules for classification derived from the hashtag analysis. They tokenize the hashtags, break them into single words and then match them against a Linux dictionary using rules like matching locations, organizations, and currencies. Using this rule-based system, they achieved a precision of 98%

2.2.2 Machine learning-based systems: In this section, we investigate machine-learning-based sarcastic classifiers and the feature sets employed. Most of the classifiers use bag-of-words as features; however, various studies also employ other features like [51] use pattern-based features derived from a large corpus of 66,000 sarcastic tweets, [19] uses sentiment lexicon-based features and train SVM using these features. They also employed emotions and user mentions and [12] used various features like affective aspects, punctuation marks, part-of-speech, length of words, emotions, semantic similarity and using different affective lexicons. They used Naïve Bayes, Decision Trees and SVM as their classifiers.

2.2.3 Deep Learning-Based systems: With the increase in processing speed and a decrease in the price of high-performance systems, deep learning-based systems are gaining popularity in Natural language processing systems. Deep learning-based sarcastic classifiers perform relatively well and are reported in the literature. We propose to use a hybrid classifier using both deep learning and machine learning. We propose incorporating the semantics and apply the BERT to obtain the BERT embeddings and then use various machine learning classifiers trained on the BERT embeddings.

3. METHODOLOGY

We describe our classifier for automatic sarcasm detection in this section. The classifier exploits text semantics, context as a feature and other surface features extracted using transformers.

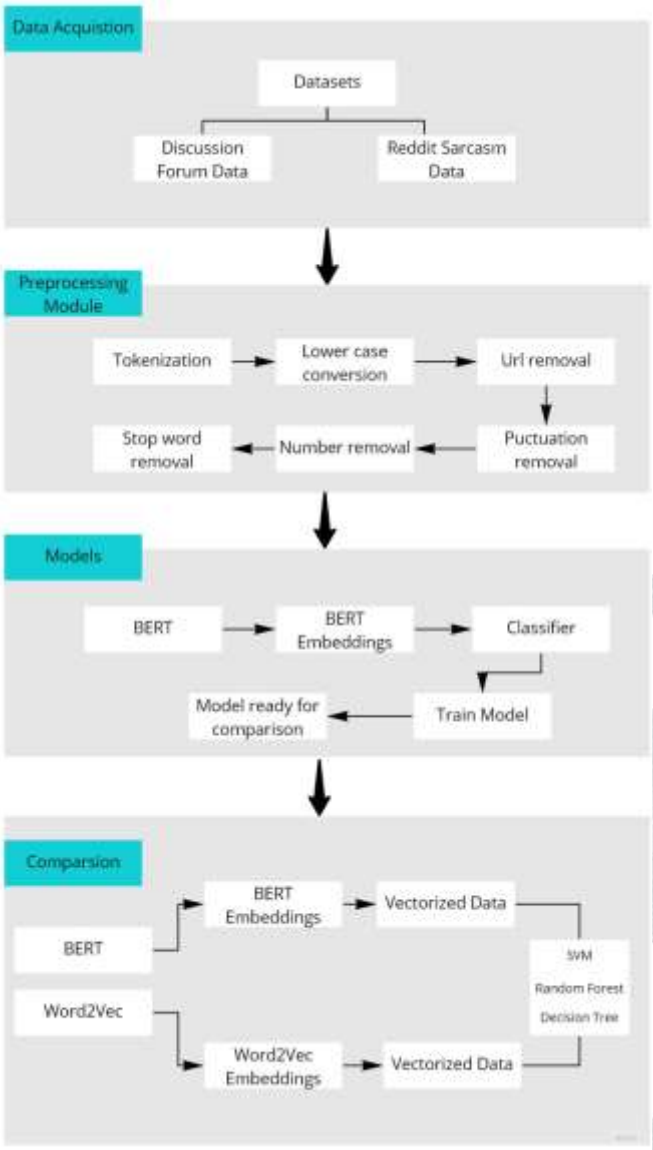

Figure 1 Overall working of the model

Our classifier uses BERT and affective lexicons to capture semantic features. The sentences in the dataset are probed for affective words, and if an affective word is found, we feed the affective word to the BERT. Vector along with other words in the sentence are fed in BERT model to retrieve the sentence's BERT embedding. Our model has the following steps.

3.1 Pre-processing:

Pre-processing being the primary and first step in our sarcasm detection process cleans data for uniformity, removes noise and inconsistencies. Pre-processing text includes few steps: First task that will be done is URLs (links) removal, images and hash-tags. Also, its preferred to remove account holder name & special characters with spell check applied through dictionary usage; The abbreviations which are used are replaced with substitutes. The main purpose of this research is to improve text classification accuracy. Data acquisition being the first module refers to data collection from various sources, Pre-processing is the second module in which text is refined and allowed to follow certain pattern so that its well streamed. Next, its followed by certain other classification techniques for further analysis and thus classifies tweeter posts by exploring deep learning usage for the tweet sarcasm detection. The steps of preprocessing are shown below in Figure 2

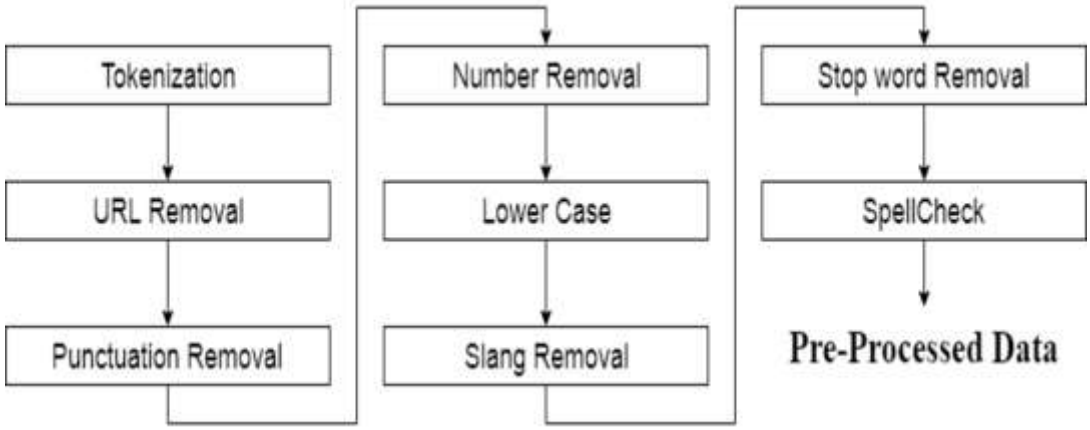

Figure 2 Steps of Pre-Processing

3.2 Semantics using BERT and Affective Lexicons

Affective lexicons are set of polarity words and thus an excellent resource to analyze the sentiment polarity. Affective words describe the emotions expressed in conversations. BERT is used in various studies to capture semantics and converts data into a vector space dimensional model. We combine the affective lexicons and concatenate the words in all lexicons. The text is probed for affective words in the combined lexicon, and if there is a hit, the word is then passed to BERT to obtain its semantic extension. We use our novel algorithm to generate the vectors.

3.3 BERT Embeddings

Google's BERT (Bidirectional Encoder Representations from Transformers) has become the state-of-art in NLP (Natural Language Processing) and is implemented in various NLP tasks. Word embeddings are low-dimensional and dense vector representations of words. It is feasible to model the semantic relevance of words numerically and execute mathematical operations by converting a word to an embedding. Google's transformers are more beneficial than the existing sequential models (GRU, LSTM and RNN,). The benefits are not limited to the efficient modelling of long-term dependencies between the words in a temporal sequence and the adequate training of the models by elucidating the sequential dependency of preceding words. BERT has two models: Base model with 12 number of transformer blocks, 768 hidden layers and 12 attention heads and the other model with 24 transformer blocks,24 hidden layers and 16 attention heads. We use BERT to obtain the surface-level contextual features and convert the text into the numerical format that is then passed to several machine learning models for building the classifier. We use several BERT models in our experiments to validate our hypothesis that using both semantic and contextual features aid in the sarcasm detection task. We use" bert-base-nli-mean tokens"," bert-base-uncased", and" bert-base-cased" as our model for obtaining BERT embeddings. We perform our experiments using 6000 messages from each dataset and use 80-20 test train split for evaluation.

3.4 Machine learning classifier:

Once we obtain the embeddings for each sentence, these are then passed to different machine-learning classifiers for training. The Classifiers used are SVM, Random Forest and Decision Trees.

4. EXPERIMENTAL SETUP AND RESULTS

We discuss the experiments, datasets used, and the results achieved in this section. The sarcasm classifier we propose classifies text into sarcastic and non-sarcastic classes. Thus, we take sarcasm detection as a binary classification. We extracted semantic features using BERT and other surface-level and contextual features using BERT embeddings. These embeddings are then fed to our machine learning classifiers. We use Support Vector Machine (SVM), Decision Trees (DT), Random Forest (RF) for classification. We compare our results with BERT as baseline. We tested our classifier on two standard datasets.

4.1 Datasets

In our experiments, we employ two datasets. The datasets we use are short text-based. The datasets are both manually and automatically annotated and are used in various studies. Also, these datasets are well balanced with a sufficient number of both sarcastic and non-sarcastic examples.

The first date set that is used is Self-Annotated Reddit Corpus (SARC). This dataset contains wide variety of corpus that is utilized for investigation of sarcasm and also for training purpose. SARC data is additional used for accessing and evaluating systems that detect sarcasm. The corpus proves to be ten times more powerful than any other past dataset as it had over 1.3 million sarcastic statements. Moreover, it also contains more occasions of non-sarcastic explanations which permits for learning in both unbalanced and balanced label regimes. All the explanations or statements that are present in the data are not independently annotated rather they are self-annotated by the author and thus provides the complete information on the topic, the user/ client, complete discussion setting i.e., context. The corpus is evaluated for precision and because of the accuracy that is achieved we are able to construct benchmarks for the detection of sarcasm. Thus, we are in a position to evaluate standard and baseline methods. The second set of data that we will use is Discussion Forum data (DFD). DFD is a large-scale data and is truly diverse corpus of sarcasm using a combination of linguistic analysis & crowd sourced annotation. The ultimate corpus is composed of explanatory questions, generic irony, & hyperbole data. Various directed and supervised learning experiments are conducted to highlight the quality of this corpus and achieve the finest F1 = 0.74 by employing simple feature sets. The weakly-supervised learning algorithms used to show that this dataset can accomplish high accuracy for rhetorical questions and hyperbole datasets which is much higher than the most excellent accuracy that is possible for the Generic data

Datasets can also be categorized into three types based on their size: short texts (Twitter datasets), long text (blog posts), and transcript datasets.

4.4.1 Short text: With the advent of social media and usage of these platforms through smartphones, there is an abundance of opinionated data present online that can be mined for sarcasm detection. However, due to the size restrictions on tweets, these tweets are short and contain abbreviations that lead to noise. Despite drawbacks, Twitter datasets are widely used for sarcasm detection. It is due to the widespread use of Twitter as a platform to express an opinion and Twitter search API.
4.4.2 Long text: The long text used for sarcasm detection usually includes data from the review sites, including movie reviews, product reviews, and hotel reviews.
4.4.3 Transcripts: Literature also reports the use of transcripts and dialogues as a form of datasets for sarcasm detection.

4.2 Results

The results obtained by our sarcasm classifier are present in this section. We report the various matrices on the two datasets discussed in section 4.1. Particularly, we report precision (P), F1-score (F1), recall (R), and accuracy (A) for both the classes. We compared our system with the embeddings generated using BERT model These parameters can be evaluated using confusion matrix. A confusion matrix is basically a table like structure. It's used to check the performance of a "classifier". The performance is checked on the set of test data for which the actual values are known in advance. We perform our experiments using 80-20 train-test split for evaluation.

Table 1 Results on Discussion Forum Dataset using BERT-base-nli-mean-tokens model

Classifier	Label	Baseline				Our System			
		P	R	F1	A	P	R	F1	A
SVM	0	0.78	0.79	0.78	0.78	**0.82**	**0.78**	**0.80**	**0.80**
	1	0.79	0.78	0.78		**0.78**	**0.83**	**0.8**	
DT	0	0.80	0.82	0.81	0.81	**0.84**	**0.83**	**0.84**	**0.84**
	1	0.82	0.80	0.81		**0.83**	**0.84**	**0.83**	
NB	0	0.69	0.73	0.71	0.71	**0.73**	**0.74**	**0.74**	**0.73**
	1	0.72	0.68	0.70		**0.73**	**0.71**	**0.72**	
RF	0	0.86	0.87	0.87	0.87	**0.87**	**0.88**	**0.88**	**0.89**
	1	0.87	0.86	0.87		**0.88**	**0.88**	**0.89**	

Table 2 Results on SARC Dataset using BERT-base-nli-mean-tokens model

Classifier		Baseline				Our System			
	Label	P	R	F1	A	P	R	F1	A
SVM	0	0.63	0.92	0.75	0.64	**0.70**	**0.77**	**0.73**	**0.67**
	1	0.74	0.29	0.41		**0.64**	**0.55**	**0.57**	
DT	0	0.66	0.75	0.7	0.64	**0.71**	**0.73**	**0.72**	**0.61**
	1	0.61	0.5	0.55		**0.62**	**0.59**	**0.60**	
NB	0	0.66	0.69	0.68	0.62	**0.64**	**0.55**	**0.59**	**0.65**
	1	0.57	0.53	0.55		**0.51**	**0.6**	**0.55**	
RF	0	0.63	0.82	0.71	0.63	**0.65**	**0.86**	**0.74**	**0.66**
	1	0.62	0.39	0.48		**0.7**	**0.41**	**0.52**	

From each dataset sentences are retrieved and embeddings are generated using baseline and our system. After obtaining embeddings, we train three different classifiers and record the results. The classifiers we use are Support Vector Machine (SVM), Decision Trees (DT), and Random Forest (RF). The classifiers are trained on embeddings generated by Baseline and our system using the same BERT models and the number of messages used to train the classifiers are same. The parameters for training are also same in both the settings. We use three different classifiers to check the impact of these generated embeddings on the performance of the classifiers and to validate our hypothesis that by using our system, the system's efficacy improves irrespective of the classifier used.

5. CONCLUSIONS AND FUTURE SCOPE

In this research, we have compared the effect of using BERT as the model for obtaining Word embeddings of the social media post, since BERT captures both contexts, as well as the semantics of the underlying text and hence, aided further in determining the sarcastic posts. We explored the use of deep learning network for the detection of tweet sarcasm. We employed Decision Tree, SVM and Random Forest for sarcasm detection. The results show that SVM actually improved model performance. More importantly, we discovered that the pre-trained BERT classifier can achieve better performance as compared to state-of-the-art results for sarcasm detection. We have also compared the effect of introducing BERT with another embedding framework Word2Vec.

Our future research will deal with:

- Capturing semantics at the fine-grain level using more distributional semantic models like FastText and Glove
- To use large BERT models to identify the context.
- To run our algorithm on more datasets.

6. REFERENCES

[1]. Abercrombie, G., Hovy, D.: Putting sarcasm detection into context: The effects of class imbalance and manual labelling on supervised machine classification of twitter conversations. In: Proceedings of the ACL 2016 student research workshop, pp. 107–113 (2016)

[2]. Amir, S., Wallace, B.C., Lyu, H., Silva, P.C.M.J.: Modelling context with user embeddings for sarcasm detection in social media. arXiv preprint arXiv:1607.00976 (2016)

[3]. Bamman, D., Smith, N.: Contextualized sarcasm detection on twitter. In: Proceedings of the International AAAI Conference on Web and social media, vol. 9 (2015)

[4]. 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)

[5]. Barbieri, F., Saggion, H., Ronzano, F.: Modelling sarcasm in twitter, a novel approach. In: Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 50–58 (2014)

[6]. Bharti, S.K., Babu, K.S., Jena, S.K.: Parsing-based sarcasm sentiment recognition in twitter data. In: 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 1373–1380. IEEE (2015)

[7]. Bouazizi, M., Ohtsuki, T.: Opinion mining in twitter how to make use of sarcasm to enhance sentiment analysis. In: Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015, pp. 1594–1597 (2015)

[8]. Buschmeier, K., Cimiano, P., Klinger, R.: An impact analysis of features in a classification approach to irony detection in product reviews. In: Proceedings of the 5th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis, pp. 42–49 (2014)

[9]. Davidov, D., Tsur, O., Rappoport, A.: Semi-supervised recognition of sarcasm in twitter and amazon. In: Proceedings of the fourteenth conference on computational natural language learning, pp. 107–116 (2010)
[10]. 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)
[11]. Ding, X., Liu, B., Yu, P.S.: A holistic lexicon-based approach to opinion mining. In: Proceedings of the 2008 international conference on web search and data mining, pp. 231–240 (2008)
[12]. Far´ıas, D.I.H., Patti, V., Rosso, P.: Irony detection in twitter: The role of affective content. ACM Transactions on Internet Technology (TOIT) 16(3), 1–24 (2016)
[13]. Fersini, E., Pozzi, F.A., Messina, E.: Detecting irony and sarcasm in microblogs: The role of expressive signals and ensemble classifiers. In: 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 1–8. IEEE (2015)
[14]. Filatova, E.: Irony and sarcasm: Corpus generation and analysis using crowdsourcing. In: Lrec, pp. 392–398. Citeseer (2012)
[15]. Ghosh, A., Li, G., Veale, T., Rosso, P., Shutova, E., Barnden, J., Reyes, A.: Semeval2015 task 11: Sentiment analysis of figurative language in twitter. In: Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015), pp. 470–478 (2015)
[16]. Ghosh, A., Veale, T.: Fracking sarcasm using neural network. In: Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis, pp. 161–169 (2016)
[17]. Ghosh, D., Fabbri, A.R., Muresan, S.: Sarcasm analysis using conversation context. Computational Linguistics 44(4), 755–792 (2018)
[18]. Giora, R.: On irony and negation. Discourse processes 19(2), 239–264 (1995)
[19]. Gonz´alez-Ib´anez, R., Muresan, S., Wacholder, N.: Identifying sarcasm in twitter: a closer look. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pp. 581–586 (2011)
[20]. Gregory, H., Li, S., Mohammadi, P., Tarn, N., Draelos, R., Rudin, C.: A transformer approach to contextual sarcasm detection in twitter. In: Proceedings of the Second Workshop on Figurative Language Processing, pp. 270–275 (2020)
[21]. Grice, H.P., Cole, P., Morgan, J.L.: Syntax and semantics. logic and conversation (3), 41-58 (1975)
[22]. Jan, R., Khan, A.A.: Emotion mining using semantic similarity. In: Natural Language Processing: Concepts, Methodologies, Tools, and Applications, pp. 1115–1138. IGI Global (2020)
[23]. Jena, A.K., Sinha, A., Agarwal, R.: C-net: Contextual network for sarcasm detection. In: Proceedings of the Second Workshop on Figurative Language Processing, pp. 61–66 (2020)
[24]. Joshi, A., Sharma, V., Bhattacharyya, P.: Harnessing context incongruity for sarcasm detection. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pp. 757–762 (2015)
[25]. Joshi, A., Tripathi, V., Bhattacharyya, P., Carman, M.: Harnessing sequence labeling for sarcasm detection in dialogue from tv series 'friends'. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning, pp. 146–155 (2016)
[26]. Khodak, M., Saunshi, N., Vodrahalli, K.: A large self-annotated corpus for sarcasm. arXiv preprint arXiv:1704.05579 (2017)
[27]. Kreuz, R.J., Glucksberg, S.: How to be sarcastic: The echoic reminder theory of verbal irony. Journal of experimental psychology: General 118(4), 374 (1989)
[28]. Liebrecht, C., Kunneman, F., van Den Bosch, A.: The perfect solution for detecting sarcasm in tweets# not (2013)
[29]. Liu, P., Chen, W., Ou, G., Wang, T., Yang, D., Lei, K.: Sarcasm detection in social media based on imbalanced classification. In: International Conference on Web-Age Information Management, pp. 459–471. Springer (2014)
[30]. Lukin, S., Walker, M.: Really? well. apparently bootstrapping improves the performance of sarcasm and nastiness classifiers for online dialogue. arXiv preprint arXiv:1708.08572 (2017)
[31]. Maynard, D.G., Greenwood, M.A.: Who cares about sarcastic tweets? investigating the impact of sarcasm on sentiment analysis. In: Lrec 2014 proceedings. ELRA (2014)
[32]. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
[33]. Mishra, A., Kanojia, D., Nagar, S., Dey, K., Bhattacharyya, P.: Harnessing cognitive features for sarcasm detection. arXiv preprint arXiv:1701.05574 (2017)
[34]. Misra, R., Arora, P.: Sarcasm detection using hybrid neural network. arXiv preprint arXiv:1908.07414 (2019)
[35]. Mohd, M., Jan, R., Shah, M.: Text document summarization using word embedding. Expert Systems with Applications 143, 112958 (2020
[36]. Mukherjee, S., Bala, P.K.: Sarcasm detection in microblogs using na¨ıve bayes and fuzzy clustering. Technology in Society 48, 19–27 (2017)

[37]. Nielsen, F.˚A.: A new anew: Evaluation of a word list for sentiment analysis in microblogs. arXiv preprint arXiv:1103.2903 (2011)
[38]. Oraby, S., Harrison, V., Reed, L., Hernandez, E., Riloff, E., Walker, M.: Creating and characterizing a diverse corpus of sarcasm in dialogue. arXiv preprint arXiv:1709.05404 (2017)
[39]. Poria, S., Cambria, E., Hazarika, D., Vij, P.: A deeper look into sarcastic tweets using deep convolutional neural networks. arXiv preprint arXiv:1610.08815 (2016)
[40]. Pt´aˇcek, 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)
[41]. Rajadesingan, A., Zafarani, R., Liu, H.: Sarcasm detection on twitter: A behavioral modeling approach. In: Proceedings of the eighth ACM international conference on web search and data mining, pp. 97–106 (2015)
[42]. Rakov, R., Rosenberg, A.:" sure, I did the right thing": a system for sarcasm detection in speech. In: Interspeech, pp. 842–846 (2013)
[43]. Reyes, A., Rosso, P.: Making objective decisions from subjective data: Detecting irony in customer reviews. Decision support systems 53(4), 754–760 (2012)
[44]. Reyes, A., Rosso, P.: On the difficulty of automatically detecting irony: beyond a simple case of negation. Knowledge and Information Systems 40(3), 595–614 (2014)
[45]. Reyes, A., Rosso, P., Buscaldi, D.: From humor recognition to irony detection: The figurative language of social media. Data & Knowledge Engineering 74, 1–12 (2012)
[46]. Reyes, A., Rosso, P., Veale, T.: A multidimensional approach for detecting irony in twitter. Language resources and evaluation
[47]. (1), 239–268 (2013) 47. Riloff, E., Qadir, A., Surve, P., De Silva, L., Gilbert, N., Huang, R.: Sarcasm as contrast between a positive sentiment and negative situation. In: Proceedings of the 2013 conference on empirical methods in natural language processing, pp. 704–714 (2013)
[48]. Riloff, E., Wiebe, J.: Learning extraction patterns for subjective expressions. In: Proceedings of the 2003 conference on Empirical methods in natural language processing, pp. 105–112 (2003)
[49]. Sarsam, S.M., Al-Samarraie, H., Alzahrani, A.I., Wright, B.: Sarcasm detection using machine learning algorithms in twitter: A systematic review. International Journal of Market Research 62(5), 578–598 (2020)
[50]. Tepperman, J., Traum, D., Narayanan, S.:" yeah right": Sarcasm recognition for spoken dialogue systems. In: Ninth international conference on spoken language processing (2006)
[51]. Tsur, O., Davidov, D., Rappoport, A.: Icwsm—a great catchy name: Semi-supervised recognition of sarcastic sentences in online product reviews. In: Proceedings of the International AAAI Conference on Web and social media, vol. 4 (2010)
[52]. Veale, T., Hao, Y.: Detecting ironic intent in creative comparisons. In: ECAI, vol. 215, pp. 765–770 (2010)
[53]. Wallace, B.C., Charniak, E., et al.: Sparse, contextually informed models for irony detection: Exploiting user communities, entities and sentiment. 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), pp. 1035–1044 (2015)
[54]. Wallace, B.C., Kertz, L., Charniak, E., et al.: Humans require context to infer ironic intent (so computers probably do, too). In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pp. 512–516 (2014)
[55]. Wang, Z., Wu, Z., Wang, R., Ren, Y.: Twitter sarcasm detection exploiting a contextbased model. In: international conference on web information systems engineering, pp. 77–91. Springer (2015)
[56]. Xu, L., Xu, V.: Project Report: Sarcasm Detection (2019