



DETECTION OF SARCASM IN TEXT DATA USING DEEP CONVOLUTIONAL NEURAL NETWORKS *

PULKIT MEHNDIRATTA, SHELLY SACHDEVA, AND DEVPRIYA SONI †

Abstract. With the ever increasing acceptance of Online Social Networks (OSNs), a new dimension has evolved for communication amongst humans. People now share continuous streams of messages to contribute their interests, indications, discuss activities, status, latest trends and much more. This gives an opportunity to monitor and mine the opinions of a large number of online active population in real time. On one side where researchers can find out a pattern to judge the mood of the user, a serious problem of detection of irony and sarcasm in textual data poses threat to the accuracy of the techniques evolved till date. In this paper, we are proposing a deep learning based convolutional neural network method that focuses on the skip gram technique to convert words to vectors and further to detect and identify the sarcasm and irony from the normal polarized text taken from Twitter¹. We try to discover patterns from the textual datasets based on the identifiable features. Our approach is giving an overall accuracy of 89.9%.

Key words: Online Social Networks (OSN), Sarcasm, Irony, Automation Techniques, Sentiment Analysis, Opinion Mining, Deep Learning, Convolutional Neural Networks (CNN).

AMS subject classifications. 91D30, 92B20

1. Introduction. The beginning of web 2.0 and Online Social Networking (OSNs) sites have given all new dimension to the world of communication among each other and has provided ample of opportunity for taking general public opinion to countable and provable patterns. Thus, these networks can be used as an effective means to testify trends and popularity in topics ranging from entertainment, politics, environment, economic or social issues and much more. Not, only people use standard languages like English, Spanish, German etc. but they try to be innovative by using emotion icons also known as emoticons, hashtags#, URLs and many more new things. People try to express themselves as much as possible whether they are happy or sad, annoyed or frustrated, excited or anxious, one thing they do not miss is a status update or a tweet.

These Online Social Networks like Facebook², Twitter, LinkedIn, are the virtual worlds or one can say the second life to the people nowadays. Opinion mining is the sub-disciplines within the field of computational linguistics and natural language processing which refers to the mining, classification and understanding the mood and opinion of the people expressed in social networks, blogs, news forums and other user generated content. Analysis of the sentiment is often used while performing opinion mining to identify the sentiment, subjectivity, and other affecting states in user generated text. Around the globe researchers are analyzing trends and tracking user activity to predict the big question "What's next???". Not only this they are trying to see what people like and what they hate. They are trying to find out patterns in user search queries, browsing activity logs and much more. But one thing which from decades troubling the researchers in field of text and sentiment analysis is usage of Irony and Sarcasm.

Sarcasm is a well known, commonly used and well-studied topic in linguistics. In spite of being so widely used and being part of our speech, it's inherently very challenging not only for machines but for humans also to detect sarcasm in text. As the size of text communication is getting shorter day by day, this problem of identification of sarcasm poses a real threat not only to the efficiency of the machine learning algorithms but it's a great threat to the ultimate sentiment of the sentence where it is used. So, if not crucial but it is very important to address the problem of sarcasm in textual data for the further evolution and refinement of the various systems used for analysis of the sentiments. Some studies suggest, the problem with sarcasm is transformation of the overall polarity of the sentence apparently positive or negative to its opposite or negation. The rest of the paper is as follows: section II talks about the basics of sentiment analysis and opinion mining, majority focusing on the current state of the art. In section III, we discuss the overall status of the detection

*This work was supported by Department of Computer Science, Jaypee Institute of Information Technology, Noida, INDIA.

†Department of Computer Science and Information Technology, Jaypee Institute of Information Technology, Noida, INDIA({pulkrit.mehndiratta, shelly.sachdeva, devpriya.soni}@jiit.ac.in).

¹<https://www.twitter.com>

²<https://www.facebook.com>

of the sarcasm in textual data. The introduction to convolutional neural networks (CNN) is discussed under section IV. Here we have given a very brief introduction to what CNN's are and how they work. Section V discusses our methodology, where we explained the steps taken in this study for the detection of sarcasm in text. Section VI provides us with the results we achieved using CNN's for the detection of sarcasm, hence a fusion of text analysis and evolutionary algorithm working in tandem. Conclusion and future scope of this work has been discussed in section VII.

2. Sentiment Analysis and Opinion Mining. One thing that every person wants know that "What other people think" to help them in a decision making process. Many years before when we were at midst of World Wide Web, we asked our friends for their recommendations on various topics like clothing, whom they are planning to vote and much more. But with the help of Internet and technology at our disposal, it has become a matter of mere asking to find out the opinions and experiences of the vast pool of people who are neither our friends or acquaintance nor they are acclaimed professional critics - they are the ones we have never heard of. On the contrary, the number of people giving their views and opinion on the Internet is increasing day by day. According to a survey conducted in 2008 [1], almost 80% of the Internet population does online research before buying any product at least once. Not to mention, but people do research about restaurants, hotels, places to visit, various services (doctors, travels etc.) all the time, and this review system actually has a significant impact on the purchase pattern of the user. We show urgency to share that goods and services consumption is not the only motivation behind general user turning to seek online opinions and reviews. Gathering political information and update is another trending topic that people usually do online. Recent studies [2, 3] show that how a microblogging site can predict the results of the elections of various democracies like Germany, India and much more. This hunger and dependency of the user upon the online review systems and recommendations reveals one thing that people take interest in these new systems and take opinions of various users as a first-hand object. In [1], where authors claim that maximum people use and have positive experiences during online product research, around 60% of the population still reports that either the information about the products they are searching is missing, confusing or impossible to find. Similarly, in the case of elections result prediction [4], authors mention the shortcomings while examining the predictive power of social media to display conflicting results. Hence, there is a clear need to build better information systems than the systems which already exist.

2.1. Applications of Sentiment and Opinion Mining. Sentiments and opinions are central and integral part of almost all human activities. Whenever a decision is to be made by us we always look out for others' opinions and judgment. Businessmen, multi-billion dollars companies, or a naive user everyone needs opinions and he/she would ask family and friends. But with the overwhelming growth of online social networks, individual people and organizations are increasingly using the content for decision-making purposes.

2.2. Challenges in performing Sentiment and Opinion Analysis. Most of the techniques we studied and reviewed suggested that there is a bag of words known as sentiment lexicon which helps and used to express the positive and negative sentiment. Over many years in the past, researchers have designed algorithms and proposed ways to compile a lexicon for the very same purpose of sentiment analysis. For example, **good, great, better, nice, wonderful, amazing** are positive words and **bad, no, not, poor, worse, terrible** is set some negative words.

These words worth a great deal to many but they are way far from sufficient to perform sentiment analysis. Such lexicons are an integral part of the solution but they are not sufficient enough to perform the sentiment analysis. Some of the key issues we faced understanding sentiment analysis are as:

1. Sarcasm is one of the most common problems faced by many researchers, it's not only to find the sentiment rather say if the sentence is sarcastic or not e.g. *"I enjoyed my cab ride by paying \$107 for 2 miles, thanks Uber :("*. This practice is more common among political reviews and opinions as compared to products and services.
2. The usage of sentiment lexicon words always make it easy for the automated detection of the sentiment of the sentence. But sometimes, the phrase or sentence may not have any word from the lexicon arising issues of correct classification. Most of the current systems would fail to classify them in correct bucket.
3. The exact orientation of the word used in the sentence may have opposite meaning and this depends on the domain to domain. Taking these things into account poses serious threat to the overall accuracy

and efficiency of the systems. Addressing all these problems need serious effort and dedication from the community working in this area.

3. Sarcasm and Irony. It all started when human invented language to communicate with each other, the way they discuss things, how they evolve solutions and also the art to amuse or mock others. Sarcasm is as old as language itself. Although its occurrence is not limited to English it is inherited part of British culture. For facts, BBC³ has its own page to teach non-native English speakers the art of using sarcasm in conversation.

According to Oxford English Dictionary sarcasm means *a sharp, cutting expression or a remark, a taunt or bitter gibe*. Nowadays, people use sarcasm *when they want to say something opposite to the truth, in order to be funny, or just to make a point none the less*. On the other hand, irony is considered to be a special kind of sarcasm where users' are just want to convey the opposite to the literal interpretation.

Starting from the known work by Tepperman et al. [5] dealt with the detection of sarcasm in speech, the area has received wide interest from the natural language processing (NLP) community. Not only speech, sarcasm detection has extended its sphere in various other data forms whether it's the tweets, movies or products reviews, and much more. Researchers have tried to evaluate and detect this form of a linguistic pattern via rule-based, supervised, semi-supervised and many other techniques. This continuous effort by the community people has resulted in new and interesting techniques for detection of sarcasm in textual data.

Some of the taxonomies already proposed in the literature are as:

1. In an article Camp [6] showed that usually there are four types of sarcasm: **Embedded:** This includes words and phrases that are having absurdity attached to them. **Like-prefixed:** These are the sentences and set of words which provide an implicit denial of the argument which has been made. **Propositional:** Here the sarcasm is present as a non-sentiment proposition but it has an inbuilt sentiment attached to it.
2. The famous **6-tuple representation**, proposed by Ivanke and Pexman [7] in 2003, was an important milestone in sarcasm identification in linguistics. They defined sarcasm consists of 6-tuples $\langle S, H, C, u, p, p' \rangle$ where:

S= Speaker, H= Listener
C= Content, u= Utterance
p= Literal proposition
p'= Intended proposition

This can be read as *Speaker S generates an utterance u in Context C meaning proposition p but intending that hearer H understands p'*. They quoted a various example in the article itself to show that how the sentences can be divided or mapped on the 6 tuple representation.

3. Similarly, many studies have been proposed by Wilson [8] and Giora [10] where they mentioned about situational disparity theory and negation theory of sarcasm respectively.

Many researchers treat irony and sarcasm as strongly related and sometimes equate these two terms in their studies. In a study, Brown et.al.[10] gives warning that sarcasm is not a discrete logic or part of any linguistic phenomenon but the irony is part of the verbal expression. Hence, researchers take the liberty of using the term sarcasm through verbal irony is a better option. Considering all the definitions evolved by researchers in [9, 11, 12, 13, 14], we came to a conclusion that utterance of sarcastic words involves a shift in overall valence which can go either way: it could be a shift in the literally positive intended sentence to negative or the vice-versa. The occurrence of sarcasm in written and spoken interaction may work differently. In the spoken interactions, sarcasm is often marked with a modulation in the voice or special gesture which not used for normal conversation, this can also be named as an incongruent facial expression [15, 16]. As we do not have any special voice gesture or incongruent facial expression, thus, it is highly likely that most of the time researchers may not be able to pick up the utterance of sarcasm and interpret them, same has been said by authors in [17]. Some of the examples of sarcastic texts are:

1. *The best feeling in the world is definitely being ignored. I love it #keepdoingit #bestthingever #sarcasm.*
2. *Absolutely adore it when my bus is late #sarcasm.*

³http://www.bbc.co.uk/worldservice/learningenglish/radio/specials/1210/how_to_converse/page13.shtml

3. *Wow GPRS data speeds are blazing fast :(.* #beingsarcastic
4. *Hes with his other woman: XBox 360. It's 4:30 fool. Sure I can sleep through the gun - fire #sarcastic #angry*

3.1. Techniques to detect sarcasm. Researchers from different corners of the world are trying to identify and work to solve the problem of detection of sarcasm in textual data. In an article, Davidov et.al. [18] wrote that recognition of sarcasm can benefit multiple communities working in various areas of natural language processing like a summarization of reviews, ranking systems, and dialog systems. In their technique, they performed a semi-supervised method on two different type of data sets i.e. one comprised of Amazon's product reviews and other has Tweets collected from Twitter. They had an accuracy of more than 75% on the both the datasets for identification of sarcasm and irony in them. They focused on the data specific features like punctuations, hashtags(#), sentence syntax and more.

Similarly, authors in [19] collected a training set of almost 78 thousand Dutch tweets. Their results implicate that irony and sarcasm in text is often signaled by a hyperbole, using intensifiers and exclamations, while the others are having an external marker. In another technique [20], authors presented an approach where a bootstrapping learning method to acquire a list of positive and negative tweets from sarcastic tweets have been discussed. Authors mentioned that the technique is still in its infancy and a lot of work has to be done. They have also discussed some ways in which future research can be done like sarcasm is often arisen from a description of a negative event.

One study [21] even got the precision of 98% while detecting sarcasm. They mentioned that it is of utmost importance especially for online social networks and product based services to identify and address the problem of sarcasm. They performed an analysis of the effect of irony and sarcasm on the overall polarity of the sentence/tweet. They designed a hashtag tokenizer for their solution and focused on the point that sarcasm found within hashtags can be detected more easily.

Some researchers have also tried to solve the problem by traditional ways of machine learning. In one such study [22], authors tried supervised learning technique of classification. They trained support vector machine (SVM) classifier with Bag-of-words as feature and TF-IDF as frequency measure. Though it's one of its kind studies the result proposed by authors suggest that only the Bag-of-words measure is not sufficient to claim the presence of irony and sarcasm in textual data. Many have worked in the direction of identification of sarcasm but still, much more is waiting to unfold.

Fig. 3.1 shows the tag cloud of the various features related to twitter or we can say twitter specific in nature that can be used for automatic sarcasm detection on twitter.

3.2. Why Twitter?. Twitter is an online social network and micro-blogging site, which enable users to send and read the short 140-character text messages known as tweets. *"The registered users can post and read tweets, while the unregistered users can only read the post. If a person likes the post of one user then he/she can share that particular tweet from the own profile and the process is known as a retweet. Twitter users have developed metadata annotation schemes which, shows that it compresses substantial amount of information into a comparatively tiny space"*.

Currently, the user base of Twitter is over 650 million [23] worldwide, with almost 0.13 million new users joining daily. Over a billion, new tweets are being posted every month on this from a wide range of interest groups. *"Twitter is in the list of the most visited sites according to Alexa ranking [24], and has been described as "SMS of the Internet". Its large scale and streaming nature makes Twitter an ideal platform for monitoring events in real time"*.

Recently, group of researchers from Carnegie Mellon University (CMU) came up with a study where they tried to detect sarcasm on contextualized basis [25]. They mentioned that most of the approaches in the past were treating sarcasm as a purely linguistic phenomenon, however by adding some extra-linguistic information such as the audience, the author and immediate communicative environment the accuracy of the protocol and further be increased. They also suggested that mere considering #sarcasm hashtag is not a natural indicator of sarcasm expressed in conversation, rather serves as an important communicative function to show the intent of the author to the audience, who may not otherwise be able to draw the correct idea or motive about the message. This article [25] created waves around the globe and got a lot of media attention too [26,27]. Fig 3.2 shows the basic structure of a tweet posted on Twitter.



FIG. 3.1. Tag Cloud of Twitter features to detect Sarcasm



FIG. 3.2. Basic structure of a tweet

4. Convolutional Neural Networks (CNN). Over past few years, machine learning techniques that are being applied to the artificial neural network have gained a lot of attention, especially they have played an important role in the designing of prediction models. Convolutional neural networks (CNNs or ConvNet) [28] are the type of feed forward artificial neural networks in which the pattern of connectivity between the neurons is inspired by the organization of animal visual cortex. Each neuron responds to stimuli in a restricted region of space called receptive field. Further, each field overlaps the next so that to tile the visual field. The response from each neuron within its receptive field can be approximated using the convolutional operation.

4.1. Convolutional Layer. The architecture of convolutional is formed by a stack of various independent layers that convert the input volume into output volume through a differentiated function. The CNN model helps capture spatial features of the given input which is in the form of a matrix. The model is less complex than a simple multilayer perceptron architecture. The simplicity of the model lies in the shared weights architecture design i.e. $N_W \ll N_x$ where N_W represents the number of weights and N_x represents the number of inputs. The weights are arranged as a window with shape $m \times n$ where ' m ' and ' n ' are hyper parameters defined by the user. These $m \times n$ weight matrices are called filters that run on the input matrix to get another matrix (C_1) with rows $x_r - m + 1$ and columns $x_c - n + 1$. The new matrix now goes under a **pooling** operation (explained in next section) which selects the maximum activated neuron from the matrix of the shape defined by the user. These new windows move in strides over the matrix C_1 and forms another matrix say P_1 . The C_1 and P_1 combined forms one convolutional layer. The matrix P_1 is then passed to a fully connected layer that

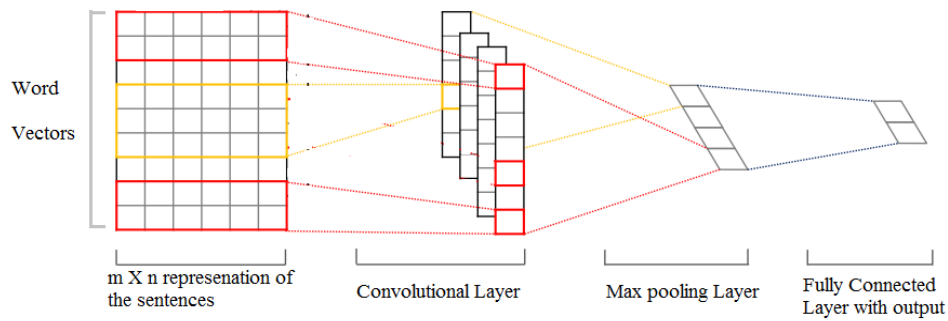


FIG. 4.1. Architecture of a Single Layer Convolutional Neural Network

TABLE 5.1
Total Number of Tweets collected from Twitter

Type of Tweet	Number of Tweets
Sarcastic	151896
Non-Sarcastic	330692
Total	482588

flattens the matrix and then classification can be performed by the simple multilayer perceptron model.

Fig 4.1 shows the various layer that comes into action when we perform the convolution function on our dataset. The final result is a fully connected layer having dropouts and softmax outputs.

4.2. Pooling Layer. In addition to convolutional layers, convolutional neural networks also contain *pooling layer*. They are usually used immediately after convolutional layers, as they play an important roll in simplifying the information in the output from the convolutional layer.

In detail, a pooling layer takes each feature map output from the convolutional layer and prepares a condensed feature map. For example, each unit in the pooling layer condenses a region of the previous layer. The pooling operation can be performed using techniques like *max-pooling*, *L2-pooling*.

5. Methodology. As discussed in section 3, Twitter can act as a perfect platform for our methodology. We started with the data collection, on to train our machine and finally testing our data against the algorithm we are using.

5.1. Data Collection. The advantage of using Twitter for our data set is, we can take as many samples as we want. Every day, people write new tweets which are sarcastic in nature. So, we started collecting the data using a Twitter API (Application Programming Interface) to stream the data. The API we used is tweeks [29]. In here we need to give keywords for collection of tweets. We carefully studied the usage of various keywords like *#Sarcasm*, *#Sarcastic*, *#beingsarcastic*. All the tweets related to these keywords were fetched and stored in the dataset for analysis purpose. The requirement to create our own data set was the unavailability of data set related to sarcasm in tweets.

As we are using classification technique which is part of supervised learning, hence we need to have training as well as a test set to check the accuracy and efficiency of the algorithm. We collected almost 0.5 million tweets which included both sarcastic and non-sarcastic tweets. The details of collected tweets are mentioned in Table 5.1.

5.1.1. Drawback of Twitter. The major disadvantage of taking Twitter into consideration is that the stream of data we receive is *too noisy*. By the term noisy, we mean to say that lot of retweets, social interaction, comments all are part of the data stream. Some people might use the hashtag *#sarcasm* to point out that they are trying to be sarcastic, but it is not sarcastic to others or in general. In another case, people may be

clearly sarcastic but they have not used `#sarcasm` hashtag. One such case can be when one person's tweet is a response to another and which is sarcastic to the context but not otherwise. All such problem adds to the noise we encounter when we stream data from the twitter.

5.2. Preprocessing of Data. Once we have the dataset, its time to clean it up to remove the noise we discussed in the previous subsection. For starters,

1. We removed every tweet in which the sarcasm is either present in the link or they are a response to another tweet.
2. All the tweets starting with `http://` are discarded along with those starting with `@`.
3. All the tweets which were not in English were again not considered for the analysis purpose. To maximize the collection of English tweets we even considered location based tweets i.e. a tweet coming from New York or London is likely to be in English as compared to one coming from Paris.
4. We even removed tweets which are too small for analysis purpose i.e. at least 3 or more words should present.
5. All the retweets are removed. Retweets are the tweets which have been forwarded by a twitter user but are not generated by him/her.
6. We lastly removed all the hashtags, friends mentions, and any non-ASCII value from the tweet.

5.3. Word to Vector Conversion. Deep neural network architectures are giving ground breaking results in every field of artificial intelligence and, Natural Language Processing is one of it. The automation of this field is much needed since the application of basic NLP techniques is tedious and a time-consuming task. The idea is to model the words into an appropriate vector space such that, we can use those vectors to perform any task. Many researchers gave their model to capture appropriate word vectors for better learning. Most famous ones are CBOW [30], skip-gram word2vec [31] and GloVe [32].

We are using the skip-gram word2vec model to convert sentences into vectors that will be further passed and will help us find out/ predicting the surrounding words in the sentence or a document. Formally, given a sequence of words which are being used for the training purpose say $W_1, W_2, W_3, \dots, W_T$, the objective of this technique is to maximize the average log probability given by equation (5.1):

$$(5.1) \quad \frac{1}{T} \sum_{t=1}^T \sum_{-c \leq j \leq c, j \neq 0} \log p(w_{t+j} | w_t)$$

In this equation, c is the size of the training context, which is a function of center word w_t . Increasing the size of the c can result into more training examples thus leading us to a higher accuracy but at an expense of more training time. The basic skip-gram formulation defines $p(w_{t+j} | w_t)$ using a softmax function given by equation (5.2):

$$(5.2) \quad p(w_O | w_T) = \frac{\exp(v_w'^T v_{w_I})}{\sum_{w=1}^W \exp(v_w'^T v_{w_I})}$$

where vw and $v'w$ are the input and output vectors of w and W is the total words in the vocabulary. This formulation is often impractical due to the very large computational cost of computation even for a small δ value.

5.4. Modeling sentences using CNN. Once we get appropriate word vectors from the skip-gram word2vec model, then we take each sentence at a time and change the word in it to a vector. The given sentence will now be a matrix where each row is a word vector. As mentioned, the CNN takes matrices as input. These sentences are padded with an unknown token `<UNK>` till the size is equal to the maximum length sentence (say S_n) in the training data. Vector for `<UNK>` is 0 in all dimensions. The matrix of a sentence has S_n rows and W_d column where W_d are the dimensions of the word vectors. We have taken 150 filters of shape $(i \times W_d)$ where $i = [3,4,5]$. Each of them stride over the input matrix and results in a matrix C_1 of shape $(S_n \times i)$ for each filter. Since the matrices are of different shapes, we perform max-pooling on each matrix and get a feature vector whose size is equal to the number of filters i.e. $(150 \times i)$ in our case. The feature vector is now connected to two neurons that will help in classification.

TABLE 6.1
Accuracy percentage variation with number of iterations

Number of Iterations	Accuracy (%)
1000	83.5
2000	85.3
3000	86.9
4000	87.5
5000	88.2
6000	88.9
7000	89.3
8000	89.7
9000	89.9
10000	89.9

5.5. Classification. In the final step, we parse the feature vector to the two neuron units for the classification. The labels of data are [0,1] and [1,0] for sarcastic and non-sarcastic tweets respectively. The two neuron units classify the data on the basis of the highest activation received from the product of feature vector and respective weights. For the calculation of the optimized function, we used *Adam* [33], an algorithm for first-order gradient-based optimization of stochastic objective functions. The idea behind the usage of this objective function is it's very straight forward to implement, computationally very efficient, needs less memory and is very well suitable for the data sets having a lot of noise or very sparse in nature. Also, the hyperparameters have intuitive interpretations and typically require little tuning. Once passed, the system keeps on evaluating the data set with more number of iterations every time. For our experiment, we have taken readings from 1000 to 10000 iterations with a space span of 1000 between each reading.

6. Results. We have trained our model on a large dataset of tweets, where a tweet is automatically labeled as sarcastic or not, depending on the occurrence of words and learning from the training set. Our experimental setup has more than 0.45 million tweets in it helping to improve the accuracy of the system. We are considering a batch size of 128 tweets at a time to train the model and then proceed similarly for the next batch.

Our system has achieved an accuracy of 89.9% after running the convolution for 10000 iterations as shown in Table 6.1. With every 1000 iterations, we see a significant improvement in the accuracy till 9000 iterations, After this, the result changes with a very minimal value. The major reason for this decline in the improvement of accuracy is the amount of randomness in the data reduces thus the accuracy becomes stagnant. The model uses only a single layer of convolution to achieve accuracy that is at par or even better than many models proposed in the literature. The graph in Fig 6.1. shows the comparison between the number of iterations and percentage accuracy.

7. Conclusion and Discussion. We have discussed an approach in this article that uses convolution neural networks for the classification and detection of sarcasm in textual data taken from a microblogging site known as Twitter. The study reveals that the algorithm performs well with single layer convolution and even out perform many existing techniques. Also, the detection of sarcasm is an open problem as many new dimensions are still to be explored. It is inherited and has become part of our daily routine conversation i.e. verbal or textual.

Most of the researchers mentioned about the amalgamation of scientific theories with practical engineering goals to analyze sarcasm in natural language text. This will not only lead to better understanding of natural language opinions and one can easily reduce the gap between unstructured information and structure data. An accurate measure to detect sarcasm will not only help researcher but it will also benefit the businesses and product services. In future, we are planning to further evaluate our technique and try to in-cooperate the multi-layer or dynamic layer convolution into consideration.

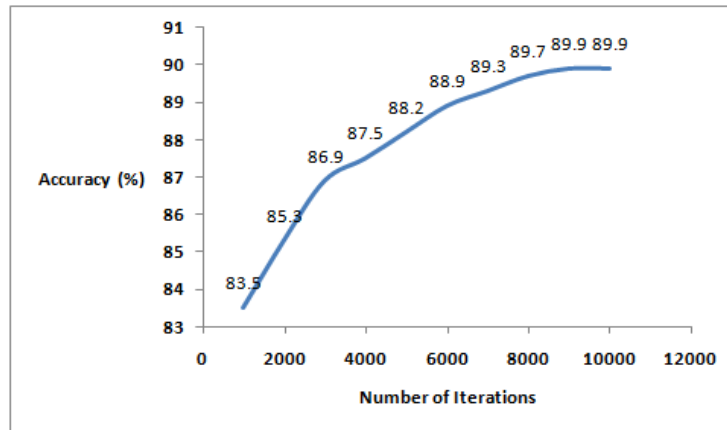


FIG. 6.1. Accuracy vs. Number of Iterations

REFERENCES

- [1] HERRIGAN, JOHN A. *Online shopping. Pew Internet & American Life Project Report 36* (2008).
- [2] TUMASJAN, A., SPRENGER, T. O., SANDNER, P. G., AND WELPE, I. M. *Predicting elections with twitter: What 140 characters reveal about political sentiment*. ICWSM, 10, 178-185.
- [3] MEHNDIRATTA, P., SACHDEVA, S., SACHDEVA, P., AND SEHGAL, Y.. *Elections Again, Twitter May Help!!! A Large Scale Study for Predicting Election Results Using Twitter*. In *Big Data Analytics* (pp. 133-144). Springer International Publishing.
- [4] CHUNG, J. E., AND MUSTAFARAJ, E. *Can collective sentiment expressed on twitter predict political elections?*. In *AAAI* (Vol. 11, pp. 1770-1771)
- [5] TEPPERMAN, J., TRAUM, T.R., AND NARAYANAN, S. "yeah right": sarcasm recognition for spoken dialogue systems.. In *INTERSPEECH*. Citeseer
- [6] CAMP, E. *Sarcasm, Pretense, and The Semantics/Pragmatics Distinction*. 46, 4, 587634, (2012).
- [7] IVANKO, S.L. AND PEXMAN, P.M., *Context incongruity and irony processing*. *Discourse Processes* 35, 3, 241279, (2003).
- [8] WILSON, D., *The pragmatics of verbal irony: Echo or pretence?* *Lingua* 116, 10, 17221743, (2006).
- [9] GIORA, R., *On irony and negation*. *Discourse processes* 19, 2, 239264, (1995).
- [10] BROWN, R. L. *The pragmatics of verbal irony. Language use and the uses of language*, 111-12, (1980).
- [11] HAMAMOTO, H. *Irony from a cognitive perspective*. *Pragmatics and beyond new series*, 257-270, (1998).
- [12] ATTARDO, S. *Irony as relevant inappropriateness*. *Journal of pragmatics*, 32(6), 793-826, (2000).
- [13] PARTINGTON, A. *Irony and reversal of evaluation*. *Journal of pragmatics*, 39(9), 1547-1569, (2007).
- [14] SPERBER, D., AND WILSON, D. *Relevance: Communication and cognition* (Vol. 142). Cambridge, MA: Harvard University Press, (1986).
- [15] BRYANT, G. A., AND TREE, J. E. F. *Is there an ironic tone of voice?*. *Language and speech*, 48(3), 257-277, (2005).
- [16] JAHANDARIE, K. *Spoken and written discourse: A multidisciplinary perspective*. Greenwood Publishing Group, (1999).
- [17] BURGERS, C. F. *Verbal irony: Use and effects in written discourse*, Ipskamp, Nijmegen, The Netherlands, (2010).
- [18] DAVIDOV, D., TSUR, O., AND RAPPOPORT, A. *Semi-supervised recognition of sarcastic sentences in twitter and amazon*. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning* (pp. 107-116). Association for Computational Linguistics (2010).
- [19] LIEBRECHT, C., KUNNEMAN, F., AND VAN DEN BOSCH, A. *The perfect solution for detecting sarcasm in tweets #not*. *WASSA 2013*, 29, (2013).
- [20] RILOFF, E., QADIR, A., SURVE, P., DE SILVA, L., GILBERT, N., AND HUANG, R. *Sarcasm as Contrast between a Positive Sentiment and Negative Situation*. In *EMNLP* (pp. 704-714) (2013).
- [21] MAYNARD, D., AND GREENWOOD, M. A. *Who cares about Sarcastic Tweets? Investigating the Impact of Sarcasm on Sentiment Analysis*. In *LREC* (pp. 4238-4243), (2014).
- [22] DAVE, A. D., AND DESAI, N. P. *A Comprehensive Study of Classification Techniques for Sarcasm Detection on Textual Data*.
- [23] TWITTER STATISTICS, www.statisticsbrain.com/twitter-statistics/
- [24] ALEXA: THE WEB INFORMATION COMPANY, www.alex.com
- [25] BAMMAN, D., AND SMITH, N. A. *Contextualized sarcasm detection on Twitter*. In *Proceedings of the 9th International Conference on Web and Social Media* (pp. 574-77). Menlo Park, CA: AAAI (2015).
- [26] *Researchers are teaching computers sarcasm with Twitter* (2016): <http://thenextweb.com/twitter/2016/01/22/researchers-are-teaching-computers-sarcasm-with-twitter/#gref>
- [27] KNIGHT, W. (2016, January) *My Favorite Thing about the Internet? Definitely the Sarcasm*: <https://www.technologyreview.com/s/545936/my-favorite-thing-about-the-internet-definitely-the-sarcasm/>
- [28] LECUN, Y., BOTTOU, L., BENGIO, Y. AND HAFFNER, P. *Gradient-based learning applied to document recognition*. *Proceedings*

- of the IEEE, 86(11), pp.2278-2324, (1998).
- [29] Researchers are teaching computers sarcasm with Twitter (2016): <http://thenextweb.com/twitter/2016/01/22/researchers-are-teaching-computers-sarcasm-with-twitter/#gref>
 - [30] MIKOLOV, T., CHEN, K., CORRADO, G. AND DEAN, J. *Efficient estimation of word representations in vector space*. arXiv preprint arXiv:1301.3781, (2013).
 - [31] MIKOLOV, T., SUTSKEVER, I., CHEN, K., CORRADO, G.S. AND DEAN, J. *Distributed representations of words and phrases and their compositionality*. In Advances in neural information processing systems (pp. 3111-3119), (2013).
 - [32] PENNINGTON, J., SOCHER, R. AND MANNING, C.D. *Glove: Global Vectors for Word Representation*. In EMNLP (Vol. 14, pp. 1532-1543), (2014).
 - [33] KINGMA, D. AND BA, J. *Adam: A method for stochastic optimization*. arXiv preprint arXiv:1412.6980, (2014).

Edited by: Kavita Sharma

Received: May 5, 2017

Accepted: Sep 2, 2017