© 2023 IJNRD | Volume 8, Issue 4 April 2023 | ISSN: 2456-4184 | IJNRD.ORG

IJNRD.ORG ISSN : 2456-4184

INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG

An International Open Access, Peer-reviewed, Refereed Journal

Effective Aspect-based Sentiment Detection approach using adhoc-cnn model

Madhumita[1], Dr. Ajit Kumar[2]

[1]Research Scholar, Sri JJT University, Vidhyanagari, Jhunjhunu, Rajasthan -333001

[2]Assistant Professor, Sri JJT University, Vidhyanagari, Jhunjhunu, Rajasthan -333001

Keywords:

Sentiment detection, Adhoc-CNN model.

ABSTRACT

A sentiment is a form of expression, which is either positive, negative or neutral; it is frequently used to communicate a negative perspective or an intensified positive statement on social media platforms, particularly on social media. In addition, recognizing aspect-based sentiments is a significant problem in sentiment analysis, as sentiment detection is a crucial factor in text categorization and has several implications. The identification of aspect-based sentiments is formulated as a binary classification problem, with deep learning and classical feature-based mechanisms used to anticipate aspect-based sentiment utterances. In this study, we construct and develop an adhoc-CNN, for recognizing aspect-based sentiment tweets. adhoc- CNN consists of two extended channels. The first channel is assumed to attain sentiment semantics, while the second channel is employed to comprehend the sentiment polarity contrast; moreover, adhoc CNN is used to store features. In addition, the adhoc-CNN model is assessed using the dataset, namely SemEval Dataset. Moreover, the performance of adhoc-CNN is compared to that of several existing sentiment detection mechanisms, and comparative analysis indicates that the proposed model outperforms these models by a significant margin in terms of metrics such as accuracy, precision, recall and F1-score.

1 INTRODUCTION

The past decade has witnessed the Internet's spectacular progress, which has led to the development of a variety of platforms for human contact. Furthermore, because of its utility, accessibility, and reachability, a platform was created in which the entire globe may share knowledge or be entertained in a single digital arena. Additionally, web 2.0 has transformed the digital lifestyle, resulting in the accessibility of UGC (User Grant Service).

Twitter is an intelligent social network and is one of the most widely used communication networks; in addition, it enables users to publish 280-character messages known as tweets. In addition, Twitter has 330 million monthly active users and 1.3 billion accessible accounts. Every day, almost 500 million tweets are sent, and other statistics indicate that 23 percent of internet users are on Twitter. In addition, Twitter data is utilized for a variety of reasons, including sentiment analysis, e-government, political organizations, trend analysis, illness tracking, event tracking, and business management. However, social media platforms are now regarded as the primary source of feedback for every event or product, enabling businesses to give the required product and understand their needs [1]. For an instance, whenever a product launch or an event is commenced, people start evaluating and tweeting. However, active consumers visit these social media platforms to read reviews before associating with or purchasing the product and these organizations rely on positive ratings.

Depending on the significance of tweets, they are classified as either non-literal or literal; non-literal tweets are also referred to as figures of speech sentiments [2]. Literal tweets are tweets that contain conventional language from a dictionary, and the polarity of feeling is straightforward to recognize. A figure of speech comprises phrases and words that express eloquence and dramatic effect. Further figurative language tweets tend to misrepresent the actual sentiment; given the nature of tweets, they can be categorized as Humor, Satire, Sarcasm, Irony, or exaggeration.

In the context of figures of speech, sentiment polarity is ambiguous, necessitating extensive investigation into the interplay between natural language processing, information extraction, and machine learning [3]. Despite metaphorical language that may be present on many social media platforms, including Instagram and Facebook, researchers have mostly focused on Twitter owing to its intellectual reach. Few academics have focused on categories such as metaphor, comedy, and satire; nonetheless, detecting sentiments based on tweets represents one of the most difficult challenges in natural language processing [4].

Nonetheless, to identify sentiment, characteristics like linguistic-based, supervised, and semi-supervised mechanisms are explored, as well as some of the most essential aspects [5].

The pattern-based feature is a type of character is used to determine sentiment in a text document by considering negative situation phrases along with positive sentiments; for instance, a tweet such as "oh I love being ignored" is sarcastic; in this feature, the interjection word start bootstrap learning mechanism is used. Patterns in the presence of encoding are employed to create the feature in this case [6].

Hyperbolic elements in this particular attribute have been utilized to recognize exaggeration or hyperbole in the provided text; it also contributes slightly to the sentiment detection phenomenon. Continuous use of adverbs or adjectives plays an important role in hyperbolic characteristics. Such characteristics are produced by using NLP tools Stanford Parser9, NLTK8, and spacy7 to detect the overstatement of PoS. (Parts of Speech). For example, "excellent weather," where excellent is an adjective [7].

Syntactic feature in this is a fundamental feature for sentiment identification; it contains text length, n-grams, negations, PoS tags, and interjections; stop words such as "am" and "at"; interjections such as "wow"; and the count of negations such as "can't." These sub-features are utilized for enhanced sentiment identification by exploiting the text [8].

Semantic-based features emphasize sentiment polarity utterances; they add a feature depending on attitude score, negative phrases, positive phrases, emotional words, positive words, and negative words. In addition, few scholars emphasized semi-features such as Word Count, Linguistic inquiry, and sentiment lexicons.

One such feature employs the present tense and counts numerous events, including emotions, replies, smileys, and the person who has tagged that content [9] [10].

Twitter-specific features include features based on metadata, primarily historical features, salient terms, user profile data, and historical salient terms. Other features include a gaze-related feature for interpreting sentiment and a legibility feature for gauging the complexity of communicated sentiments; this feature includes the number of polysyllables per word and the total quantity of words.

Motivation and contribution of research work:

It is difficult to interpret the literal meaning of sentiment on social media platforms and microblogging websites, such as Twitter, where it is commonly utilized. In other words, sarcasm is a phrase that confuses even humans as to whether it is praise or criticism. Furthermore, the metaphorical language and style of sarcastic tweets make sentiment analysis difficult; yet, the obstacles and benefits of sentiment identification motivated the researcher to define a problem and solve it using an automated procedure (automatic sarcasm detection). Automatic sentiment detection is only a computer technique that predicts if the sentiment is positive negative or neutral. In addition, this phenomenon inspired several researchers to implement sentiment detection in diverse domains.

- The purpose of our study is to determine sentiments that are derived from the contrast between positive sentiments and negative sentiments.
- To build and implement adhoc-CNN (adhoc CNN), which consists of two extended channels for efficient sarcasm identification; the first channel is intended to accomplish the sentiment semantics, while the second channel is intended to discern the sentiment polarity contrast.
- CNN is designed for holding the features; the DC-CNN model is assessed using the two datasets, i.e., the training and test sets. Task 3 of the SemEval workshop and restaurant dataset.
- Additionally, the performance of adhoc-CNN is compared to that of numerous sentiment identification strategies, and comparative analysis indicates that the proposed model beats existing models by a significant margin in terms of metrics like precision, recall, F1-score, and Accuracy.

This study here is well organized like any other normal research study. The first section begins with a history of Twitter and the significance of sentiment analysis and then emphasizes the significance of sentiment analysis. The rationale and contribution of the research activity are mentioned later in the same section. The second section focuses on current mechanisms and highlights their shortcomings. The third section describes the mathematical model of adhoc-CNN as well as its sub-mechanisms and layers; the fourth section analyses adhoc-CNN by comparing it to several state-of-the-art techniques.

2 RELATED WORK

The detection of sentiments in textual data has been the subject of research. Literature-based approaches to the automatic recognition of sentiment include lexicon-based, rule-based NLP, pattern-based, lexicon-based approaches, Corpus-based, statistically based approaches), and machine learning-based methods. Currently, the deep learning technique, which is a new trend, has gained significant ground in sarcasm recognition, even though only a small number of academics have applied it.

Numerous researchers have conducted experiments on spotting sarcasm in textual data over the past few decades because of Twitter's widespread use. In addition, the majority of extant studies are statistically, corpus-, lexicon-, pattern-, and rule-based. Moreover, deep learning has recently shown promise in natural language processing; as a result, several researchers have concentrated on the notion of a neural network to train the data for sarcasm detection; a few of the existing studies are presented below. [11] Other researchers employed pragmatic and lexical features to identify sarcasm in a statement. In the meanwhile, research [12] indicates that sarcasm must be identified by the previous statement and the sarcastic reply; research [13] focuses on lexical elements and observes that punctuation and interjection play a crucial part in sarcasm identification in a specific corpus. However, deviating from the above method, [14] and [15] observed that gestural or oral expression represented by special keyword characters and emoticons has a significant effect on sarcasm; [16] used this observation along with pattern-based and syntactic features to design a sarcasm classifier and train the model. [17] Utilizing n-grams and other signals, such as exclamations and intensifiers, since the detection characteristic is not sufficient. Consequently, [18] enhanced it and developed numerous features such as ellipsis, quotation marks, and hyperbole features for the specified goal; [19] demonstrated that in a bad scenario, positive remarks are interpreted as sarcastic comments. [20] utilized several aspects, including explicit and implicit incongruity, pragmatic and lexical; [21] employed pattern-based sarcastic commentary; and [22] investigated the psychology behind sarcasm. In this paper, behavioral modeling was constructed to identify sarcasm from Twitter data, and the influence of past tweets that give sarcasm detection and contextual information is illustrated through several examples [23]. Additional neural networks are required to learn the most recent features for investigating implicit semantic patterns to capture the extracted feature, which aids in the detection of sentiments. Developed an LSTM (Long Short-Term Memory)-based neural network for sarcasm detection with several gates to store information for a longer period, since this was deemed one of the most important challenges. Moreover, [24] developed a network by integrating a pooling neural network and GRU, and [25-26] used pre-trained CNN to extract and understand the difference between positive and negative; [27] developed a fusion mechanism and extracted the contextual information from discourse section, further user embedding was used to encode the personality feature and stylometric feature of users; this mechanism proved to be effective on Reedit1 corpus; [28] developed a fusion mechanism and extracted the contextual information. Additionally, by employing attention encoders [29-31] through memory interactions and neural-memory processes, researchers have improved aspect representations mechanisms [32-33] for focusing attention, in that order [34], [35], [36]. These two factors manage the relationships between different features. Comforting phrases that ease the extraction process, can assist with multitask learning as well as ambiance [37]. For example, Li and Lam [38] suggested two Long Short memory-interaction mechanisms, Aspect and sentiment extraction using Long Short-Term Memory (LSTM) words in a setting that supports several learning tasks. The memory operations relied on the positional memory model information on aspects, emotive language, and global the number of emotive words. They also included the restrictions of genuine phrases to aid the extraction process via a different LSTM. Multilayer neural attention using tensor operators was proposed by Wang et al. [39].

Moreover, during the study, it was discovered that a substantial amount of research has been conducted on sentiment identification, but only a few algorithms demonstrate the techniques. Due to the sensitivity of tweets, these systems have either overlooked performance data or produced unsatisfactory outcomes. Few additional strategies obtained acceptable performance on a simple dataset but failed on a complex dataset; these methods also suffer from the issue of complexity. Consequently, this research focuses on the design and development of a technique that takes into account all prominent performance metrics and two complicated datasets; potential approaches are detailed in the next section.

3 PROPOSED METHODOLOGY

In this study, an efficient mechanism termed $adhoc - CNN$ is developed for efficiently recognizing aspect-based sentiments; also, the mathematical model of the suggested work is shown in this part. $adhoc - CNN$ that consists of two extended channels in improvised CNN, with each channel serving a specific function. Figure 1 depicts the planned architectural design.

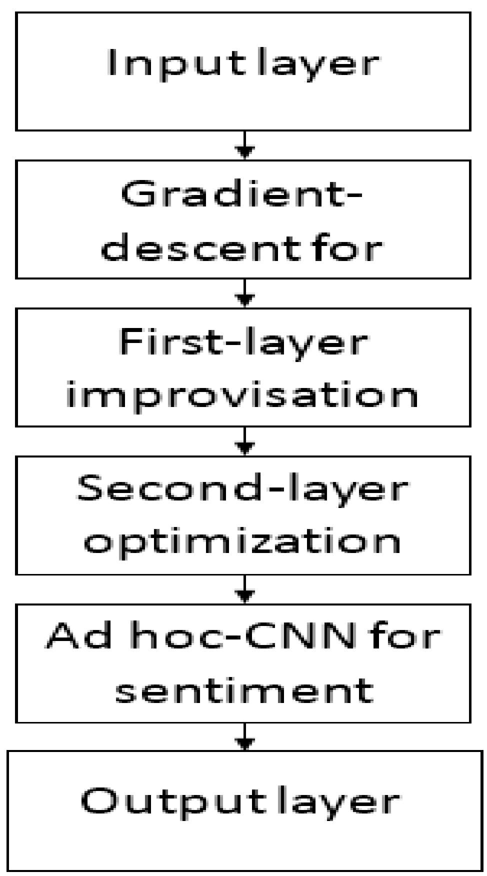

Figure 1 System architecture of the proposed model

In this section, the architecture of the proposed model is designed; it consists of two layers the first one deals with feature engineering, and the second layer deals with aspect-based sentiment detection.

3.1 Encoding layer

In this section, the input is fed via the encoding layer where each word sequence is mapped onto the vector representation for processing. Consider a sequence $j = \{v_1, v_2, v_3, \dots v_k\}$ here k determines the word embedding in a sentence, the matrix is further formulated by $H_\delta = \{X^{kUg}\}$ here $g$determines the embedding position and δ indicates word sequence. The word embedding is encoded via the $H_{pos} = X^{kUg}$, at this point word embedding is integrated along the location embedding as shown in the equation given below:

$$H = \left\{X^{k(g+G)}\right\} \tag{1}$$

3.2 Feature Engineering

Natural language processing mostly makes use of neural networks to achieve suitable performance, by generalizing this the information is stored for an elongated period resulting in insufficiency and decay in error resulting in backflow. However, these results in gradient descent, this approach consists of three components, the input as depicted in the equation shown below: as stated above H is a matrix:

$$S_t = \gamma\left(Y_s l_{t-1} + U_s f_{t-1} + V_s u_t\right) \tag{2}$$

m_t as the forget gate and mathematically represented as

$$m_t = \gamma\left(U_s f_{t-1} + V_s u_t + Y_p l_{t-1}\right) \tag{3}$$

The output is given by:

$$j_t = \gamma\left(U_j f_{t-1} + Y_j l_{t-1} + V_j u_t\right) \tag{4}$$

$$\ddot{n}_t = \tanh\left(U_j l_{t-1} + V_n u_t\right) \tag{5}$$

n_t as the memory cell and given as:

$$n_t = m_t . l_{t-1} + S_t . \ddot{n}_t \tag{6}$$

The hidden state vector in the model is represented as:

$$b_t = j_t . \tanh(n_t) \tag{7}$$

As shown in the above equation i.e from equation 2 to equation 7; u_t depicts the given vector along sentence H. This denotes the multiplication operation and α denotes the sigmoid function. Henceforth various constraints V, T *and* U are trained by the following input. Furthermore, the output is shown as the equation $q_{GNN} = \{b_1, b_2, b_3, \dots b_v\}$ is derived in form of a matrix from this phase, the feature engineering is carried out upon mathematical modeling by the below equation:

$$q_{GNN} = \sum_{i=1}^{v} b_i\,(v)^{-1} \tag{8}$$

3.3 First-layer

In the proposed model, each layer is assigned a task; considering the fact of aspect-based sentiment analysis on the application of a common principle, hence the first layer is designed that upholds the essential features to detect the sentiment.

Here, the first input given to the sentiment is encoded via the output as $q_{GNN} = \{b_1, b_2, b_3, \dots b_v\}$ this shows that the input is fed to the first layer. The first layer consists of several layers such as the attention and linear layer; the input to the first layer consists of two divisions i.e., word vector combined with the output from the feature engineering phase. In this phase, the weight associated which each word is computed and multiplied along further word embedding as shown in the new representation.

Step1	$q_{FL} = \{b_1, b_2, b_3, \dots b_v\}$ as feed via the external memory of the channel i.e., $c \epsilon\, H^{a\times b}$, where a represents the unit size of the network
Step2	given each depiction b_{mn} into given linear layer
Step3	the result is shown from step 2 into the function $tanh$ and evaluates the unique representation through the below equation.
	$b'_{mm} = \tanh(w_c + Z_c\, b_{mm})$
	b'_{mn} shows the word representation.
Step4	Added computation of the weights of each word from equation8, by word embedding c_o.

Step5	The matrix obtained from the above step is normalized with softmax function to attain the final matrix
	$k_{mn} = \exp(b'^{\varpi}_{mn} r_l) \left(\sum_{m=1}^{v} \exp(b'^{\varpi}_{mn} r_l)\right)^{-1}$
Step 6	on computing the feature vector to use the output vector in the first layer; η determines the sentiments.
	$\iota_\kappa = \sum_{m=1}^{v} k_{mn} b_{mn}$

3.4 Second-Layer

In this layer to correlate the difference among the aspect-based sentiment, hence the attention is determined via the word in a sentence as given by the input, the encoding layer upon application of an encoder the input, and the output from the feature engineering phase. This procedure is similar to the first layer stated in equations 10 to 15.

$$b'_{st} = \tanh(T_t\, b_{mt} + w_t) \tag{9}$$

$$k_{mt} = (\exp(b'^{\varpi}_{mn} r_s)) \left(\sum_{h=1}^{v} \exp(b'^{\varpi}_{mt} r_s)\right)^{-1} \tag{10}$$

$$A_z = \sum_{m=1}^{v} k_{mn} b_{mn} \tag{11}$$

The above equation shows the output from the second layer.

3.5 adhoc-CNN

The input fed to the proposed model termed as H the output of this model is shown as the convolutional model given by:

$$H_d = \{x_d^1, x_d^2, \ldots\ldots, x_d^v\}_{v\times q} \tag{12}$$

In the equation stated above q and d determines the convolution filter and a layer embedding; the feature representation depicted by the convolution block is shown in the below equation. Here A shows the convolutional operation and x_d shows the set of weights.

$$H_d = A(H, x_d) \tag{13}$$

However, in this convolutional operation, each block shows a rectified linear unit, batch normalization, and convolution of every convolution operation formulated by the specific filter given by$x_{conv} \epsilon H^{b\times(T+t)}$, where b depicts the window size. The convolution operation is shown by the below equation.

$$n_m = b(z + T_{conv}\,.\,t_{m:m+b-1}) \tag{14}$$

As shown in the equation above n_m determines the word range in between the t_m to $t_{m:m+b-1}$. The range value is given in between b to v. A feature map is built for the equation shown below:

$$e = [e_1, e_2, \ldots\ldots, e_{v-b+1}] \tag{15}$$

Henceforth, in the equation given, by using the $adhoc\ max - pooling$ layer of using max-pooling layer; in this $adhoc\ max - pooling$; each word embedding consists of the window, these windows reach peak value; since traditional CNN results in data loss. The $adhoc\ max - pooling$ holds the information along with the features chosen from the $adhoc - CNN$.

3.6 Aspect-based sentiment classification

$adhoc - CNN$ incorporates three features i.e., q_{CNN}, $A_z$and q_{GNN}. These are combined and shown below by the equation:

$$q_f = combine\left[q_{CNN}; q_{GNN}; A_z\right] \tag{16}$$

The q_f is embedded with three layers, as depicted via a normalized function for sentiment classification shown as:

$$t = softmax\left(t_x Multi\_layer\left(q_f\right) + o_x\right) \quad (17)$$

The sentiment classification is as shown below:

$$C^{Asp-sent} = \frac{1}{K_g} \sum_{m=1}^{K_g} \left(\dot{f}_m \log f_m + (1 - \dot{f}_m) \log(1 - \dot{f}_m)\right) \quad (18)$$

Here f_m determines the predicted label for mth text K_g denotes the total sequence of text and f_m depicts the true label for mth text.

4 PERFORMANCE EVALUATION

In this part, we assess the proposed model $adhoc - CNN$; to construct the model, python is used with different machine learning libraries on Visual Studio 2017 as the integrated development environment (IDE); and the system model is a Windows platform with 2GB Nvidia Graphics, 16GB RAM, and 1TB SATA HDD. $adhoc - CNN$ is a GPU-dependent model that uses GPU for data training. In addition, this part consists of several subsections; in the first dataset, details are addressed, then evaluation metrics are computed and compared with other technologies, and finally, a comparative performance analysis is presented.

4.1 Dataset Details

First, we consider the Semeval2018 Task3 dataset, for sentiment detection of tweets [28]. This corpus contains a balanced training set of 3,833 tweets and a test set of 784 tweets. The experimental results are based on the same set of SemEval2018 task 3A, which is a shared task involving the binary categorization of positive and negative sentiments.

4.2 Evaluation Metrics

Precision, recall, and F-score is used to evaluate the efficacy of the aspect-based sentiment identification. Precision is the proportion of accurately predicted aspect-based sentiment occurrences relative to the total number of anticipated sentences. A recall is the proportion of accurately anticipated aspect-based sentiment occurrences sentences relative to the total number of actual aspect-based sentimental statements. The F-score is the harmonic mean of accuracy and recall [40]. These metrics are computed as follows:

Precision, also known as a positive predictive value, is one of the performance metrics used for the categorization of the prediction; also, it reflects the ratio of TP (True Positive) to TP and FP (False Positive), [41-42] with a greater precision value indicating a more effective model.

$$precision = \frac{true\_postive}{true\_positive + false\_positive} \quad (23)$$

In general, recall is the proportion of genuine positives properly detected; additional recall is the ratio of true positives to the total of true positives and false negatives.

$$Recall = \frac{true\_postive}{true\_positive + false\_negative} \quad (24)$$

The F1 score is an assessment of the test's accuracy; further F1 scores are derived using precision and recall. In general, we may assert that the F1 metric is the mean of accuracy and recall.

$$F1_{Score} = \frac{(2\,precision)recall}{Precision + Recall} \quad (25)$$

Where $true\_positive$ is the number of estimated word sequences that are accurate. $false\_positive$ is the number of phrases that were expected to be sarcastic but are not. The number of sentences that are predicted to be non-sarcastic but are sarcastic. However, current models exhibit some promise with an accuracy of 82%; in comparison to these models, $adhoc - CNN$ achieves an astounding 97.7 percent accuracy. Similarly, in terms of accuracy, other state-of-the-art techniques of the present model attain 96 % precision. Compared to these methodologies, recall for $adhoc\ CNN$s at the rate of 96 % is somewhat higher. Figure 2 shows the accuracy comparison. Figure 3 shows the precision comparison. Figure 4 shows the recall comparison. Figure 5 shows the F-1 score comparison.

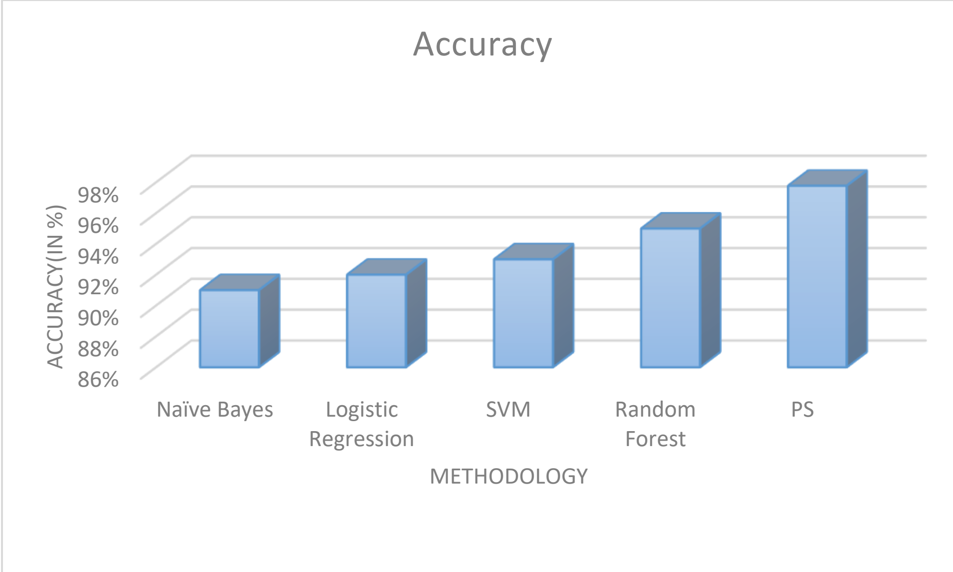

Figure 2 Accuracy comparison on Sameval 2018 dataset

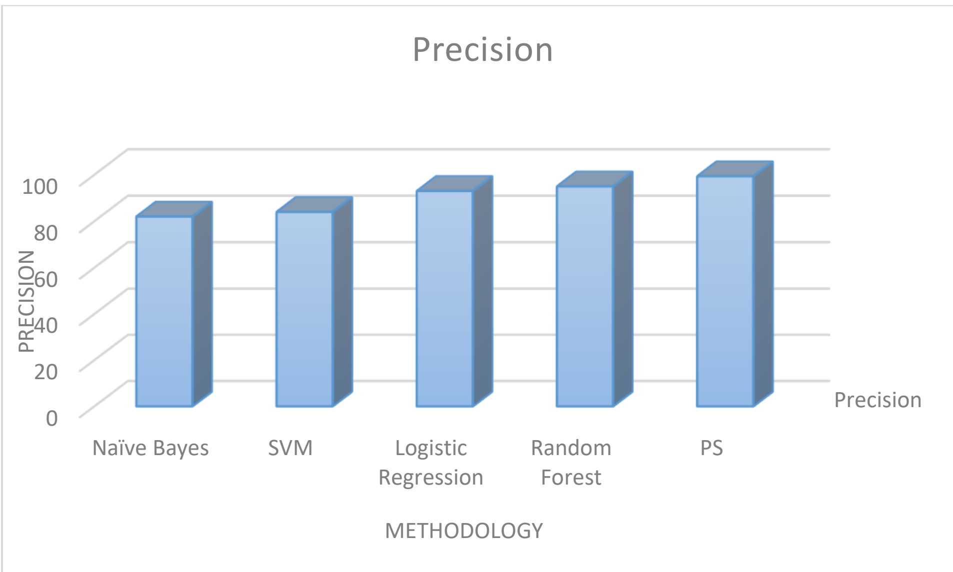

Figure 3 Precision comparison on Sameval 2018 dataset

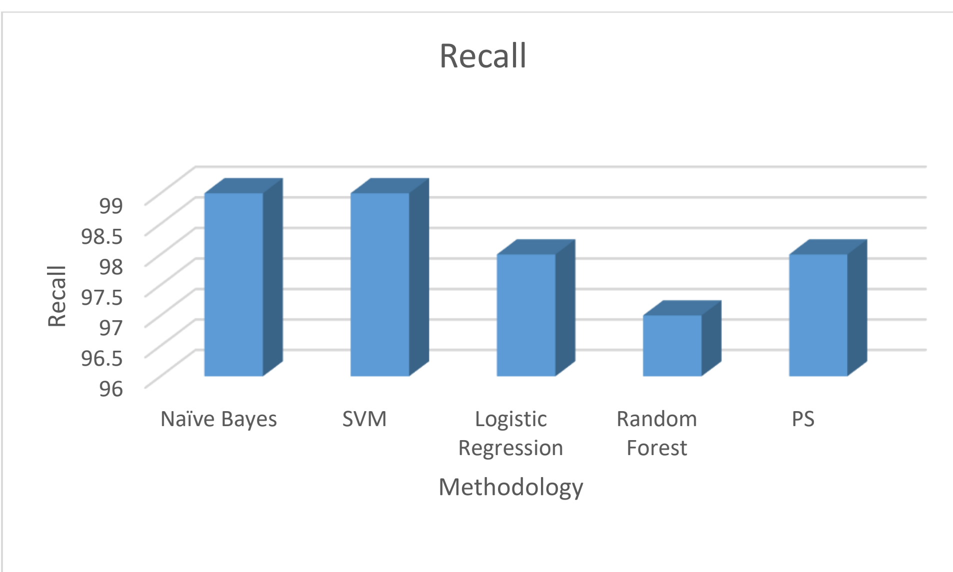

Figure 4 Recall comparison on Sameval 2018 dataset

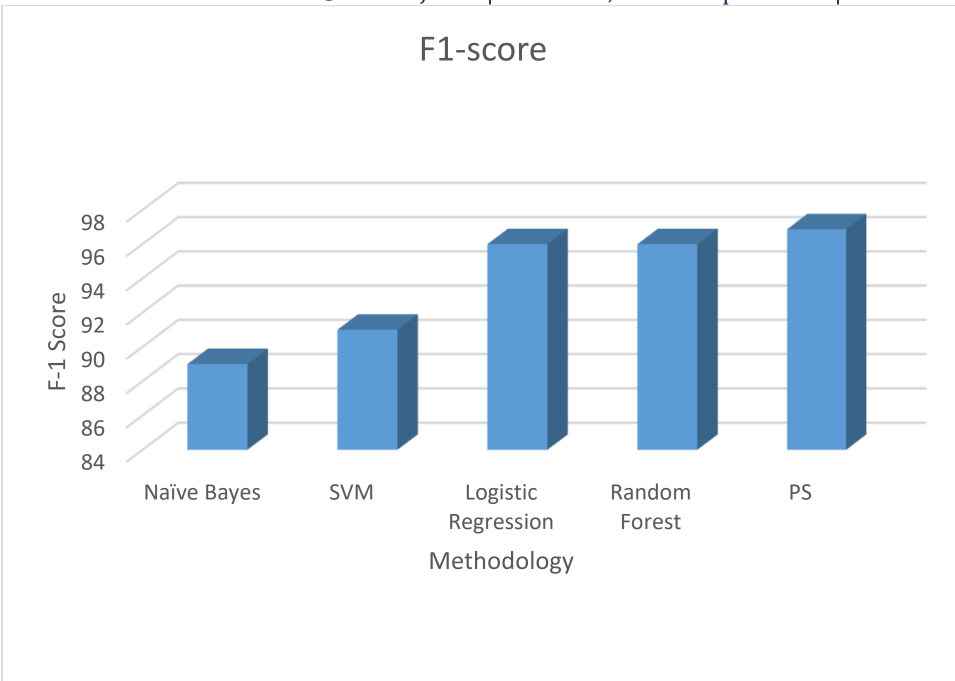

Figure 5 F1-score comparison on Sameval 2018 dataset

5 CONCLUSION

In social media, particularly on Twitter, sarcastic remarks are used to critique playfully; detecting aspect-based sentiment on the Twitter platform is one of the most difficult jobs using textual data and a vast area of research for natural language processing (NLP). Due to Twitter's tremendous growth over the past decade, several attempts have been made to identify aspect-based sentiments. We construct and implement a $adhoc - CNN$ technique to recognize sarcastic comments in this research. $adhoc - CNN$ consists of two adaptive channels and enhancements to CNN's ability to learn and forecast inefficiently. adhoc-CNN is assessed using two standard datasets, namely SamEval 2018 with a 91% accuracy. adhoc-CNN beats the previous model by a significant amount. Although adhoc-CNN works exceptionally well on these datasets, given the weaknesses of the Twitter platform and user comments, there are still some areas that require more investigation, in the real-time dataset.

Reference

[1] Ghosh and T. Veale, "Fracking sarcasm using neural network," in Proceedings of the 7th workshop on computational approaches to subjectivity, sentiment and social media analysis, 2016, pp. 161–169.

[2] E. Riloff, A. Qadir, P. Surve, L. De Silva, N. Gilbert, and R. Huang, "Sarcasm as contrast between a positive sentiment and negative situation," in Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2013, pp. 704–714.D. I. H. Farías, V. Patti, and P. Rosso, "Irony detection in twitter: The role of affective content," ACM Transactions on Internet Technology (TOIT), vol. 16, no. 3, p. 19, 2016.

[3] A. Ghosh and T. Veale, "Magnets for sarcasm: Making sarcasm detection timely, contextual and very personal," in Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, 2017, pp.482–491.

[4] D. Tang, B. Qin, and T. Liu, "Aspect level sentiment classification with deep memory network," arXiv preprint arXiv:1605.08900, 2016.

[5] Y. Tay, L. A. Tuan, S. C. Hui, and J. Su, "Reasoning with sarcasm by reading in-between," arXiv preprint arXiv:1805.02856, 2018.

[6] A. Joshi, P. Bhattacharyya, and J. M. Carman, ``Automatic sarcasm detection: A survey,'' *ACM Comput. Surv.* (CSUR), vol. 50, no. 5, p. 73, 2017.

[7] M. Rush, S. Chopra, and J. Weston, ``A neural attention model for abstractive sentence summarization,'' 2015, *arXiv:1509.00685*. [Online]. Available: https://arxiv.org/abs/1509.00685.

[8] D. Golub and X. He, ``Character-level question answering with attention,'' 2016, *arXiv:1604.00727*. [Online]. Available: https://arxiv.org/abs/1604.00727.

[9] D. Bahdanau, K. Cho, and Y. Bengio, ``Neural machine translation by jointly learning to align and translate,'' 2014, *arXiv:1409.0473*. [Online]. Available: https://arxiv.org/abs/1409.0473

[10] [J. Tepperman, D. Traum, and S. S. Narayanan, ``Yeah right: Sarcasm recognition for spoken dialogue systems,'' in *Proc. InterSpeech*, Pittsburgh, PA, USA, Sep. 2006, pp. 1838_1841.

[11] J. Eisterhold, S. Attardo, and D. Boxer, ``Reactions to irony in discourse: Evidence for the least disruption principle,'' *J. Pragmatics*, vol. 38, no. 8, pp. 1239_1256, 2006.

[12] R. J. Kreuz and M. G. Caucci, ``Lexical in_uences on the perception of sarcasm,'' in *Proc.Workshop Comput. Approaches Figurative Lang.*, 2007, pp. 1_4.

[13] S. Amir, C. B. Wallace, H. Lyu, P. Carvalho, and J. M. Silva, ``Modelling context with user embeddings for sarcasm detection in social media,'' in *Proc. CONLL* 2016.

[14] R. Gonzalez-Ibanez, S. Muresan, and N.Wacholder, ``Identifying sarcasm in Twitter: A closer look,'' in Proc. 49th Annu. Meeting Assoc. Comput. Linguistics, Hum. Lang. Technol., vol. 2, 2011, pp. 581-586.

[15] D. Davidov, O. Tsur, and A. Rappoport, ``Semi-supervised recognition of sarcastic sentences in Twitter and Amazon,'' in *Proc. 14th Conf. Comput. Natural Language Learn.*, 2010, pp. 107_116.

[16] Liebrecht, F. Kunneman, and A. van den Bosch, ``The perfect solution for detecting sarcasm in tweets #not,'' in Proc. 4th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal., 2013, pp. 2937.

[17] K. Buschmeier, P. Cimiano, and R. Klinger, ``An impact analysis of features in a classication approach to irony detection in product reviews,'' in Proc. 5th Workshop Comput. Approaches Subjectivity, Sentiment social media Anal., 2014, pp. 4249.

[18] G. Zhai, Y. Yang, H. Wang and S. Du, "Multi-attention fusion modeling for sentiment analysis of educational big data," in Big Data Mining and Analytics, vol. 3, no. 4, pp. 311-319, Dec. 2020, doi: 10.26599/BDMA.2020.9020024.

[19] A. Joshi, V. Sharma, and P. Bhattacharyya, ``Harnessing context incongruity for sarcasm detection," in Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process., vol. 2, 2015, pp. 757762.

[20] M. Bouazizi and T. O. Ohtsuki, ‘‘A pattern-based approach for sarcasm detection on Twitter,’’ *IEEE Access*, vol. 4, pp. 5477–5488, 2016.

[21] A. Rajadesingan, R. Zafarani, and H. Liu, ``Sarcasm detection on Twitter:A behavioral modeling approach," in *Proc. 18th ACM Int. Conf. Web Search Data Mining*, Feb. 2015, pp. 79_106.

[22] Ghosh and D. T. Veale, ``Fracking sarcasm using neural network," in *Proc. 7th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal.*, 2016, pp. 161_169.

[23] M. Zhang, Y. Zhang, and G. Fu, ``Tweet sarcasm detection using deep neural network," in *Proc. COLING 26th Int. Conf. Comput. Linguistics, Tech. Papers*, 2016, pp. 2449_2460.

[24] S. Poria, E. Cambria, D. Hazarika, and P. Vij, ``A deeper look into sarcastic tweets using deep convolutional neural networks," in *Proc. COLING 26th Int. Conf. Comput. Linguistics, Tech. Papers*, 2016, pp. 1601_1612.

[25] E. Sulis, D. I. H. Faras, P. Rosso, V. Patti, and G. Ruffo, ``Figurative messages and affect in twitter: Differences between irony, sarcasm and not," *Knowl.-Based Syst.*, vol. 108, pp. 132_143, Sep. 2016.

[26] Hazarika, S. Poria, S. Gorantla, E. Cambria, R. Zimmermann, and R. Mihalcea, ``Cascade: Contextual sarcasm detection in online discussion forums," in Proc. 27th Int. Conf. Comput. Linguistics, 2018, pp. 18371848.

[27] Van Hee, E. Lefever, and V. Hoste, "Semeval-2018 task 3: Irony detection in english tweets," in SemEval-2018 workshop at the 16th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL HLT 2018). Association for Computational Linguistics, 2018, pp. 39–50.

[28] Riloff E, Qadir A, Surve P, De Silva L, Gilbert N, Huang R (2013) Sarcasm as contrast between a positive sentiment and negative situation. In: EMNLP 2013—2013 conference on empirical methods in natural language processing, proceedings of the conference. Association for Computational Linguistics (ACL), pp 704–714.

[29] Peters ME, Neumann M, Iyyer M, Gardner M, Clark C, Lee K, Zettlemoyer L (2018) Deep contextualized word representations. arXiv preprint arXiv:1802.05365.

[30] Cer D, Yang Y, Kong SY, Hua N, Limtiaco N, John RS, Constant N, Guajardo-Cespedes M, Yuan S, Tar C et al (2018) Universal sentence encoder. arXiv preprint arXiv:1803.11175.

[31] Cambria, S. Poria, A. Gelbukh, and M. Thelwall, "Sentiment Analysis is a Big Suitcase," IEEE Intelligent Systems, vol. 32, no. 6, pp. 74-80, 2017.

[32] Liu, Sentiment Analysis and Opinion Mining (Synthesis Lectures on Human Language Technologies). Morgan & Claypool Publishers, 2012.

[33] K. Dave, S. Lawrence, and D.M. Pennock, "Mining the Peanut Gallery: Opinion Extraction and Semantic Classification of Product Reviews," Proc. 12th Int'l Conf. World Wide Web, pp. 519- 528, 2003.

[34] A. Tripathy, A. Anand, and S.K. Rath, "Document-level Sentiment Classification using Hybrid Machine Learning Approach," Knowledge Information Systems, vol. 53, no. 3, pp. 805- 831, 2017.

[35] B. Liu, "Sentiment Analysis and Subjectivity," Handbook of Natural Language Processing, pp. 627-666, 2010.

[36] K. Schouten and F. Frasincar, "Survey on Aspect-Level Sentiment Analysis," IEEE Trans. Knowledge and Data Eng., vol. 28, no. 3, pp. 813-830, 2016.

[37] J. Wang et al., "Mining Multi-Aspect Reflection of News Events in Twitter: Discovery, Linking and Presentation," Proc. 2015 IEEE Int'l Conf. Data Mining, pp. 429-438, 2015.

[38] W. Wang, S.J. Pan, D. Dahlmeier, and X. Xiao, "Recursive Neural Conditional Random Fields for Aspect-Based Sentiment Analysis," Proc. 2016 Conf. Empirical Methods in Natural Language Processing, pp. 616–626, 2016.

[39] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain, "Sentic LSTM: A Hybrid Network for Targeted Aspect-Based Sentiment Analysis," Cognitive Computation, vol. 10, no. 4, pp. 639-650, 2018.

[40] H. Luo, T. Li, B. Liu, B. Wang, and H. Unger, "Improving Aspect Term Extraction with Bidirectional Dependency Tree Representation," arXiv Preprint arXiv:1805.07889, p. 12, 2019.

[41] H. Li and W. Lu, "Learning Latent Sentiment Scopes for Entity-Level Sentiment Analysis," Proc. 31st AAAI Conf. Artificial Intelligence, pp. 3482-3489, 2017.

[42] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos, "Semeval-2015 Task 12: Aspect Based Sentiment Analysis," Proc. 9th Int'l Workshop on Semantic Evaluation, pp. 486-495, 2015.