CRF-MEM: Conditional Random Field Model Based Modified Expectation Maximization Algorithm for Sarcasm Detection in Social Media

Anbarasu Sivalingam[1], Karthik Sundararajan[2], Anandhakumar Palanisamy[3]*

[1] *Department of Mathematics, Anna University, India*
[2] *Department of Computer Science & Engineering, B. N. M. Institute of Technology, India*
[3] *Department of Information Technology, Anna University, India*
anbu@annauniv.edu, karthi24mar@gmail.com, anandh@annauniv.edu

Abstract

Text processing is an important task in various machine learning applications. One among the applications is Sentiment analysis. However, the presence of sarcasm makes it difficult for analyzing the sentiment of the statement. In the current scenario, the amount of sarcastic statements in any social media platform is high taking the forms of memes, comments, trolls etc. Hence it is important to identify sarcasm to preserve the polarity of any given statement. Sarcasm usually means the opposite of what the sentence seems to convey. While the existing works in literature have focused on detecting sarcasm, the proposed model, in addition to that, determines the levels of sarcasm present in the text, which will aid in finding the level of harshness present in the statement. In this work, an unsupervised learning model, Conditional Random Field model based Modified Expectation Maximization (CRF-MEM) algorithm has been proposed for detecting sarcasm in tweets. The proposed model aims to overcome the limitation present in the traditional EM algorithm, the random assignment factor, with the proposed aspect relationship value. Experimental results showed that the proposed CRF-MEM achieved an accuracy of 91.89% whereas the traditional EM displayed an accuracy of 80% in detecting sarcasm from text.

Keywords: Text mining, Natural Language Processing, Computational linguistics, Artificial intelligence

1 Introduction

Natural Language Processing (NLP) is an essential field which involves understanding the process and principles of human interaction. NLP based application development is quite challenging as it has to handle various challenges in natural language understanding and generation. There are a number of NLP tasks which are subdivided into categories for convenience. Social media is one of the surging trends in the world today. Social media provides a platform that facilitates sharing of information, ideas and other forms of interests. It is being used in many forms by different platforms for various reasons. Users access the social media services through web-based technologies on mobile phones, desktops and computers. Apart from connecting to a network of people, it also provides a vast platform to the community groups or people to discuss different issues of their interest. People often share their views towards any happening with tweets, retweets, post, likes and` dislikes etc. Twitter is one among the popular social media networks which are used by millions daily. Around 330 million active users use twitter and it has over 500 million user accounts. Twitter is the largest source of breaking news every day. Twitter provides valuable data that can help in strategic planning, providing beneficial insights for the community. The data obtained from twitter are being used in developing many applications such as sentiment analysis, sarcasm detection, community detection etc.

Sentiment analysis is a process which identifies and extracts meaningful information from the user's opinions. However, due to the informal language widely preferred in social media by the users, understanding the motivation behind the sentence is very difficult. Sarcasm is a type of speech or writing which actually means the opposite of what the sentence seems to convey. Sarcasm is usually meant to insult or to mock someone. According to Cambridge English dictionary, "Sarcasm is the use of remarks that undoubtedly means the contrary of what a person says, in order to hurt or criticize someone". The Oxford definition of sarcasm goes like this: "the use of irony to make or express contempt". Collins dictionary states sarcasm as "mocking, contemptuous or ironic language intended to convey scorn or insult". The following is an example of a sarcastic statement.

"Good fences make good neighbors lol"

This line points out, in a sarcastic way that distancing ourselves from our neighbors' will help in building a good relationship with them; though in real life it is not the case. Therefore, detection of sarcastic statements is very important in order to analyze the actual emotion of the statement.

The major contributions of this paper are as follows:

- So far, the existing literatures in the field of sarcasm detection aimed at determining whether a text is sarcastic or not. The main motivation behind this research work is to take a step ahead by not only detecting sarcasm present in the text but also identifying the level of sarcasm that the statement conveys.

- A Conditional Random Field model based Modified Expectation Maximization (CRF-MEM) algorithm has been proposed to overcome the randomness factor of the existing

*Corresponding Author: Karthik Sundararajan; E-mail: karthi24mar@gmail.com
DOI: 10.53106/160792642023012401005

Expectation Maximization (EM) algorithm.

2 Related Work

An ensemble feature selection approach for detecting irony and sarcasm from news and reviews was developed in [1]. An ensemble of unigrams, semantic, psycholinguistic, statistical features was used for figurative content detection and highlighted the common features of satire and irony. In [2], a sarcasm detection model based on fuzzy and Naïve Bayes approach was developed. Feature list includes function words, POS and n-grams. The naive Bayes algorithm was found to be effective than the fuzzy clustering techniques since only a small set of data was used for classification. A deep learning framework, Attention-of-Emoticons Based Convolutional Neural Network (AEB-CNN), for enhancing the performance of Sentiment analysis was developed by [3]. The authors carried out the research by integrating attention based entities along with emoticons. The overall accuracy of Sentiment analysis of Chinese microblogs was around 85-89%.

In [4], an Auto-WEKA tool to obtain a combination of machine learning algorithms on Spanish and Mexican corpus Twitter was developed. They developed a satire identification approach with psycholinguistic features that obtained encouraging results with an accuracy of 85%. A pattern-based approach for sarcasm detection was developed in [5]. The authors came up with four sets of features to cover the various types of sarcasm. The model achieved an accuracy of 83.1% with a precision equal to 91.1% using random forest classification technique. In [6], a set of discriminative features to automatically detect irony sentences using Naive Bayes and Decision tree approach were identified. Different types of conceptual features were used to identify verbal and situational irony based collected corpus words in tweets. This method obtained an accuracy of 72.3% and 80.44% for balanced distribution and imbalanced distribution respectively. In [7], the authors developed a discriminative model based on Ontology for sentiment classification called as Sentiment Oriented Terminological Ontology (SOTO). A Multi-rule based sarcasm detection framework was presented for sarcasm detection in [8]. The authors had developed a framework for detecting sarcasm as well as classifying it into multiple types. In [9], a Corpus based approach for sarcasm detection in twitter was developed by the authors. The data was compared with the ontology based emotion detection techniques to analyze the mood of a person. The approach used only Parts-of-speech tagging for feature extraction.

An approach combining Natural language processing and data mining was presented by [10]. Improvement was shown by finding the frequency of words used during the US elections. In [11], the authors developed a model based on Google BERT (Bidirectional Encoder Representations from Transformers) to handle volume, velocity and veracity of data. The performance of the model was compared with other classical and contemporary approaches such as Support Vector Machine, Logistic Regression, Long Short Term Memory etc. A Hidden Markov Continual Progression Cosine Similar (HM-CPCS) was developed by authors in [12] in order to figure out the effect of pre-processing and then to optimize the process of sentiment analysis. Cosine Similarity function was used for removing stop words and the method improved the accuracy by 9% over the existing methodologies. In [13], the authors developed a classifier model that detected Arabic-sarcasm tweets using Weka. The model achieved recall, precision and f-score of 0.659, 0.710, and 0.676 respectively. More features could have been included in detection since some of the tweets were misclassified.

A system to perform sentiment analysis of product reviews with expectation maximization algorithm was presented by authors in [14]. Latent Semantic Indexing (LSI) was used to analyze the text documents containing text reviews. There were problems like co-reference, ambiguity and anaphora resolution. In [15], the authors designed a variation of EM algorithm to improve the efficiency of the EM algorithm based on the sampling and online strategy. Unlike the author's algorithm, the proposed CRF-MEM model does not estimate the parameters by the random initial guess. The proposed model uses aspect term values obtained from the CRF model as the initial guess to create a normal distribution. The model does not converge at a local maximum like other traditional EM algorithms. The approach maximizes the likelihood estimation by improving the accuracy of the algorithm with minimum number of epochs.

An ensemble learning methodology for feature selection has been proposed in [16]. In order to obtain a robust feature subset, various feature lists were aggregated with the help of Genetic algorithm.

A deep learning approach was proposed in [17] for identifying sarcasm. The authors analysed the performance of the topic enriched word embedding with traditional word embedding techniques and found out that the former adds value to the process of Sarcasm detection. In [18], an inverse gravity moment based term weighted word embedding model has been proposed for sarcasm detection. A three layer stacked Bi-LSTM architecture was presented by the authors for identifying documents containing sarcasm. The model attained an accuracy of 95.30%.

In [19], the authors presented a feature ensemble methodology for sarcasm detection. The model incorporated various hyperbolic and pragmatic features for the purpose of identifying sarcastic statements. Authors inferred that ensemble feature selection yielded better results over the conventional feature selection approach. A probabilistic model based deep learning approach was presented in [20] for detecting sarcasm present in the texts. The authors augmented the term pair weightage information with the traditional word to vector model in order to improve the value of contextual information. The model used Convolutional neural network model for classification and attained an accuracy of 97.25%.

Various approaches have been proposed for the process of Sarcasm detection. Some of the works have used various types of features to detect sarcasm, some have used machine learning and deep learning algorithms to perform the same. Most of the existing works deals with supervised machine learning algorithms where manual labelling of inputs was required. It is difficult to perform manual labelling in real time as it requires a lot of computation time for training. This proves to be a real challenge when the data grows dynamically and the labels cannot be predefined. The main point of focus of this research is to exploit the contextual properties of sarcasm. This has been achieved by computing the aspect value from the given input. The traditional sentiment analysis based algorithms can predict the sentiment involved in a statement by just considering a particular word without bothering about the impact of other words on it. When sarcasm detection is concerned, a similar strategy is not applicable because it is essential to model the relationship of a word with its neighbouring words to determine the context in which it is used. For example, the word 'good' has different inference when used with different set of words and emoticons preceding and succeeding it.

3 Proposed Work

The detailed working of the proposed CRF-MEM algorithm is described in this section. The proposed model works in two layers. In the first layer the model determines the sarcastic statements and the second layer identifies the sarcastic levels present in the sarcastic tweets. The proposed architecture diagram is demonstrated in Figure 1.

The steps involved in the proposed model include preprocessing of raw data by using various preprocessing techniques. Further, the proposed sarcastic level detection model extracts the aspect level information of the text by building the term pairs based on CRF model. The aspect level information thus obtained replaces the initial random guess in the EM algorithm. The proposed algorithm then classifies the data as sarcastic or non-sarcastic. Finally, the classified sarcastic statements are categorized into different levels.

The proposed model consists of four phases:

(i) Data collection and preprocessing
(ii) Extraction of aspect information
(iii) Proposed CRF-MEM based sarcasm detection
(iv) Proposed sarcastic level categorization

3.1 Data Pre-processing

The first module in the proposed approach is data preprocessing, where the input tweets are processed by a sequence of steps. The initial step in preprocessing is data cleaning. In data cleaning, all the urls, links, misspelled words, hash tags are removed which is followed by tokenization. In tokenization, the character stream of words is divided into individual tokens. These tokens may contain abbreviations or acronyms which are normalized to their full forms. After normalization, stemming of words is done where the inflected or derived words are reduced to their root word. The preprocessed data is then fed into the model for further processing.

3.2 Extraction of Aspect Information

Most of the existing works deals with supervised machine learning algorithms where manual labeling of inputs is required. It is difficult to perform manual labeling in real time as it requires a lot of computation time for training. This can be a real challenge when the data is large and grows dynamically where the labels cannot be predefined. The authenticity of the persons labeling is also a factor to consider while labeling. Unsupervised learning algorithms have less complexity when compared with supervised learning algorithms since there is no need to label the inputs. Hence an unsupervised approach was attempted for the purpose of sarcastic detection. The main idea behind the proposed work is to utilize the aspect term pair relationship information to replace the randomness factor in the EM algorithm. An aspect is a part or feature of something. To calculate the aspect value of a sentence, it is essential to analyze the context of a sentence.

The Conditional Random Field (CRF) model identifies the- multivariate interdependencies between the words present in a statement. The CRF algorithm is a statistical modeling method for structured prediction. The traditional sentiment analysis based algorithms can predict the sentiment involved in a statement by just considering a particular word(s) without bothering about the impact of other word(s) on it. When sarcasm detection is concerned, a similar strategy is not applicable because it is essential to model the relationship of a word with its neighboring words to determine the context in which it is used. For example, the word 'good' can give different inference when used with different set of words and emoticons preceding and succeeding it. The proposed work aims to model this relationship between terms present in the text with the proposed CRF-MEM model as it is essential to figure out the dependencies between multiple words in the text.

The Conditional Random Field algorithm models multiple variables by considering the context of the text. It predicts the label sequence for the input samples. The CRF model assumes that the features are dependent on each other and predicts the future observations while learning a pattern. It is a combination of both Hidden Markov Model (HMM) and Maximum Entropy Markov Model (MEMM). In terms of performance, it outperforms the other models to solve the entity recognition problem.

The Conditional Random Field for each term pair is calculated by,

$$p(y \mid x) = \frac{1}{Z(x)} \prod {}_{(t=1)}^{T} exp\{\sum {}_{(k=1)}^{K} \theta_k f_k (y_t, y_{t-1}, x_t)\}. \quad \textbf{(1)}$$

'y' (label) denotes the hidden state and 'x' (entity) represents observed variable. The normalization constant Z(x) is a summation of all state sequences so that the summation results in one. 'f' denotes the feature functions and 'λ' denotes the weights. Then MLE is carried out by Weight estimation after which the features are defined.

The first step in CRF is annotation. Annotation is a process of tagging the words with the corresponding parts-of-speech tags or labels. Once all the words in the given sentence are assigned with the corresponding labels, they are extracted to construct the term pairs. Then the aspect value of each term pair is calculated. The aspect values obtained for each term pair in the sentence are then combined with a cumulative function to represent the value as a tweet. Finally, the aspect value of a tweet is served to replace the initial random guess in the traditional EM algorithm.

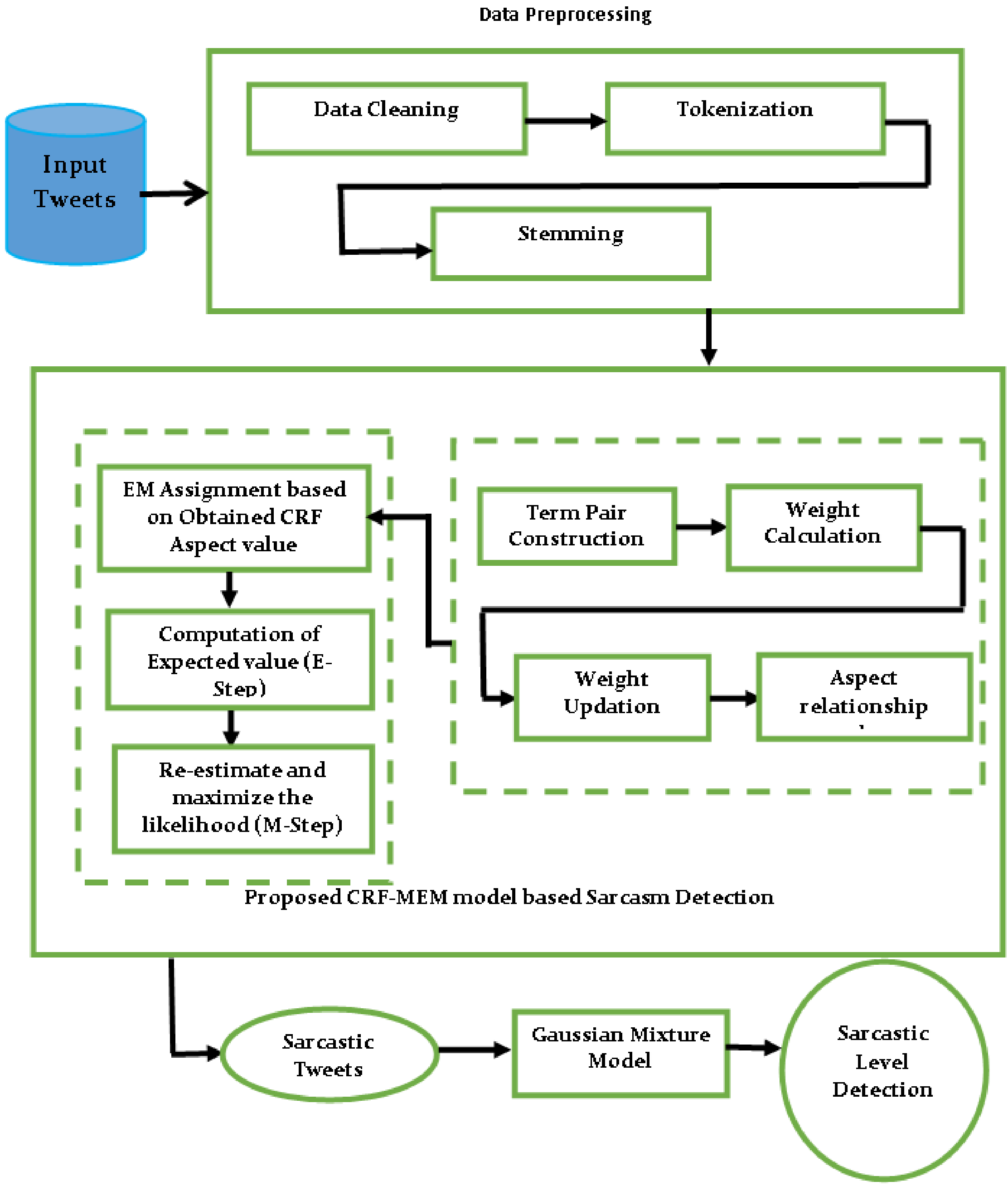


Figure 1. Architecture of the proposed CRF-MEM model

3.3 Proposed CRF-MEM based Sarcasm Detection

The objective of Expectation maximization algorithm is to estimate a maximum likelihood estimate of the parameters in statistical methods in which data does not have enough information. The Expectation Maximization algorithm contains two steps - the Expectation (E) step and the Maximization (M) step. Expectation step creates a feature function for the estimating the likelihood values for the parameters. Maximization step maximizes the likelihood until the convergence is satisfactory. The basic steps of the algorithm are:

1. E-Step: An initial assumption is developed for the parameters of the model. A probability distribution is created.
2. The model is fed with the newly observed data.
3. M-Step: The distribution obtained from the Expectation step is maximized for the new vales.
4. The process is repeated until stability is reached.

The parameter estimation is always improved through the multi-step process of the EM algorithm. In general EM algorithm, sample assignment is created based on the random initial guess for the model's parameters. Maximizing the likelihood function based on the random initial guess may not be optimal in all cases and the E step runs slowly as the procedure approaches a local maximum.

In the proposed work, instead of implementing EM algorithm with its random initialization step, initialization is achieved with aspect value computed from the proposed conditional random field model. The likelihood function and weight value for each text is calculated which is then iteratively maximized until convergence. Based on the likelihood obtained, the text is classified as sarcastic or not sarcastic. The proposed CRF-MEM model provides maximum likelihood estimation with a better accuracy than the traditional

EM algorithm. The sarcastic statements are then categorized into different levels based on the level of harshness present in the statement. The detailed working of the proposed methodology is as follows

3.3.1 Procedural & Mathematical Analysis

Let 'S' be a tweet containing n words.

$S = \{w_1, w_2, \dots, w_n\}$

Let 'l_i' be the corresponding label assigned for the word 'w_i'

Let 'H' be a vector of labels for the input tweet S,

$H = \{l_1, l_2, \dots, l_n\}$

Let 'L' be a set formed by constructing the term pairs from the vector H

$L = \{(l_1, l_2), (l_2, l_3), (l_3, l_4), \dots, (l_{n-1}, l_n)\}$

Let 'F' be the Feature Function which checks whether a particular label is present in the set of labels L

$$F1(a,b) = \begin{Bmatrix} 1, (a,b) \in L \\ 0, otherwise \end{Bmatrix}. \quad \mathbf{(2)}$$

The Aspect value of each term pair (a, b) is calculated based on the Conditional random field model as,

$$P(a,b) = \frac{1}{Z(a,b)} q(a,b). \quad \mathbf{(3)}$$

where,

$$q(a,b) = \exp(\prod\nolimits_{i=1}^{n} \prod\nolimits_{j} \lambda_j F1(a,b)). \quad \mathbf{(4)}$$

and the partition function z (a, b)

$$z(a,b) = \sum\nolimits_T \sum\nolimits_b q(a,b). \quad \mathbf{(5)}$$

where T denotes the total number of tweets.

Initially, random weights are assigned to each term pairs for training. Weights are then assigned to the term pairs. The weights are then updated through a learning rate until the difference in weights is smaller than negligible value. Let 'm' be the number of term pairs such that the learning rate 'e' is defined as

$$\textit{Learning Rate}\ (e) = (1/m). \quad \mathbf{(6)}$$

Let $\Delta\lambda$ be a small variance in λ calculation

$$\Delta\lambda = e * (E - D). \quad \mathbf{(7)}$$

where the empirical expectation is denoted as

$$E = \sum\nolimits_T \sum_{i=1}^{n} F1(l_{i-1}, l_i). \quad \mathbf{(8)}$$

and the predicted expectation is represented as

$$D = \sum\nolimits_{i=1}^{n} P(l_{(i-1)}, l_i) * \sum (F1(l_{(i-1)}, l_i))\}. \quad \mathbf{(9)}$$

The aspect value of each term obtained from the CRF model is then combined by a cumulative function to represent a tweet in the EM algorithm. The first step in the traditional EM algorithm is to create a uniform distribution with the random guess parameters. In the proposed model, the random guess parameters are replaced by the aspect values. A normal distribution is created in the proposed model with the aspect value and the sentiment score of a tweet using the Probability Density Function. The Expectation and the Maximization steps of the model are defined as,

E-STEP:

The Probability Density Function of normal distribution is calculated by the formula,

$$p(x_i \mid b) = \frac{1}{\sqrt{2\pi\sigma^2 b}} exp \frac{-(x_i - \mu_b)^2}{2\sigma^2 b}. \quad \mathbf{(10)}$$

For each data points the a_i and b_i are calculated to identify to which cluster the data point belongs to,

$$b_i = p(b|x_i) = \frac{p(x_i|b)p(b)}{p(x_i|b)p(b) + p(x_i|a)p(a)}. \quad \mathbf{(11)}$$

$$a_i = p(a|x_i) = 1 - b_i. \quad \mathbf{(12)}$$

M-STEP

For each class the parameters μ and σ are computed.

For class 'b' the parameters are calculated by using the formula,

$$\mu_b = \frac{b_1 x_1 + b_2 x_2 + \dots + b_n x_n}{b_1 + b_2 + \dots + b_n}. \quad \mathbf{(13)}$$

$$\sigma_b^2 = \frac{b_1(x_1 - \mu_1)^2 + \dots + b_n(x_n - \mu_n)^2}{b_1 + \dots + b_n}. \quad \mathbf{(14)}$$

Similarly, for class 'a' the parameters are calculated by using the formula,

$$\mu_a = \frac{a_1 x_1 + a_2 x_2 + \dots + a_n x_n}{a_1 + a_2 + \dots + a_n}. \quad \mathbf{(15)}$$

$$\sigma_a^2 = \frac{a_1(x_1 - \mu_1)^2 + \dots + a_n(x_n - \mu_n)^2}{a_1 + \dots + a_n}. \quad \mathbf{(16)}$$

The above procedure is repeated until the algorithm converges, giving a maximum likelihood estimate.

3.4 Proposed Sarcastic Level Categorization

The proposed model classifies the tweets into two: sarcastic and non-sarcastic. Sarcastic tweets are then grouped into different levels based on the Gaussian mixture model. GMM is a probability based model that considers all information developed from a limited number of distributions with unidentified parameters. The mixture models may be learnt with the GMM algorithm. GMM represents normally distributed subpopulations within an overall population. GMM doesn't require any prior knowledge about the data point. It automatically allows the model to learn the subpopulations and classifies the data accordingly. Since the data is assigned to the subpopulation without knowing any information about the data, it might be coming under unsupervised learning category.

A Gaussian mixture model consists of the mixture component weights as well as the component mean and variance or covariance. A univariate Gaussian mixture model has a mean and variance whereas a multivariate mixture model has mean and covariance matrix.

The probability density function of the multivariate GMM model is given by,

$$p(\vec{x}) = \sum_{i=1}^{k} \phi_i N(\vec{x} \mid \vec{\mu_i}, \textstyle\sum_i). \tag{17}$$

where

$$N(\vec{x} \mid \vec{\mu_i}, \textstyle\sum_i) = \dfrac{1}{\sqrt{(2\pi)^k \left|\sum_i\right|}} \exp\left(-\dfrac{1}{2}(\vec{x}-\vec{\mu_i})^T \textstyle\sum_i^{-1} (\vec{x}-\vec{\mu_i})\right). \tag{18}$$

$$\textstyle\sum_{i=1}^{k} \phi_i = 1. \tag{19}$$

The Gaussian Mixture Model can be classified into various types depending on the covariance of the difference classes. The types of model are spherical, diagonal, tied or full covariance GMM models. Covariance matrix and its mean are determined by the Gaussian distribution. The direction and lengths of axes are determined by Gaussians covariance matrix.

1. Full - the components may independently assume any shape and position.
2. Tied – clusters would have identical shapes and the shape can be any.
3. Diagonal - Clusters are tilted all along the coordinate axes else there might be a chance that eccentricities might diverge between the cluster components.
4. Spherical - the clusters are with circular contours.

Based on the cluster standard deviation of GMM model, sarcasm is categorized into four different levels as low, medium, high and very high. These levels of sarcasm will help in identifying the motivation or the level of harshness present in the statement.

4 Results and Discussion

The proposed work has been carried out on the dataset which contains sarcastic tweets and non-sarcastic tweets obtained from the Twitter Streaming API. Tweepy connects to the Twitter Streaming API and downloads the tweets. The main reason behind working with streaming real time data is that raw information can be obtained, without any alteration of the content, and the real time data will be much closer to reality than the regular datasets. The tweets were downloaded with the help of following hastags: #sarcasm, #sarcastic, #not. The raw data obtained from the Twitter is preprocessed through a series of steps such as removal of user names, URLs, punctuations and symbols. After preprocessing, the data is fed into the CRF model for finding the aspect value by constructing the term pairs between the adjacent words. Labels are assigned for each term pair and the aspect value for each term pair is calculated. The first step in the Conditional Random Field model is to split the given statement into individual words and then the labels are assigned to each word by using the POS tagger. The individual words with their labels are shown in Figure 2.

Sample tweet

Safe trip then use the same sarcastic tone when

u are pleading for votes in 2 years

POS tag assignment:

[('safe', 'JJ'), ('trip', 'NN'), ('then', 'RB'), ('use', 'VBP'), ('the', 'DT'), ('same', 'JJ'), ('sarcastic', 'JJ'), ('tone', 'NN'), ('when', 'WRB'), ('u', 'NN'), ('are', 'VBP'), ('pleading', 'VBG'), ('for', 'IN'), ('votes', 'NNS'), ('in', 'IN'), ('2', 'CD'), ('years', 'NNS')]

Figure 2. Label assignment using CRF

```
Initial Term pair weights

NN RB 0.5848178400653891
JJ JJ 0.569393114977848
NN VBP 0.860087776736498
DT JJ 0.916852282004149
VBG IN 0.050736790558971956
IN NNS 0.031317833898269365
NNS IN 0.38992128300023243
JJ NN 0.20091548713311924
IN CD 0.6950138931100126
NN WRB 0.9595648795990936
WRB NN 0.932120017437819
VBP DT 0.6208465440307055
VBP VBG 0.033817566676833525
CD NNS 0.16338001935839352
RB VBP 0.801909558823737
```

Figure 3. Weighted term pair values

Once the labels are assigned to the corresponding words, the words and the labels are separated into different vectors for usage. After the separation of labels from the words the term pairs are constructed by combining the adjacent labels. Weights are assigned to each term pair by using the random function in the Figure 3. The initial weights are then iterated with the learning rate to obtain the new weights as shown in Figure 4.

```
Current term pair:
NN RB
Old weight:      0.584818
Learning rate:   0.010000
Small variation in λ: 0.150000
New weight: 0.434818
```

Figure 4. Weight updation

After successful iterations, the aspect value of each term pair is obtained. The aspect value information of each term pair is shown in Figure 5.

```
After iteration Term pair weights

NN RB 0.43481784006538904
JJ JJ 0.4193931149778479
NN VBP 0.710087776736498
DT JJ 0.766852282004149
VBG IN -0.09926320944102804
IN NNS -0.11868216610173063
NNS IN 0.23992128300023244
JJ NN -0.09908451286688075
IN CD 0.5450138931100126
NN WRB 0.8095648795990936
WRB NN 0.782120017437819
VBP DT 0.47084654403070547
VBP VBG -0.11618243323166474
CD NNS 0.013380019358393353
RB VBP 0.651909558823737
```

Figure 5. Aspect value information of term pairs

The aspect values of each term pairs of a tweet are then combined by a cumulative function. A normal distribution is created with the aspect value of a tweet and the sentiment score of a tweet using the probability density function as seen in Figure 6.

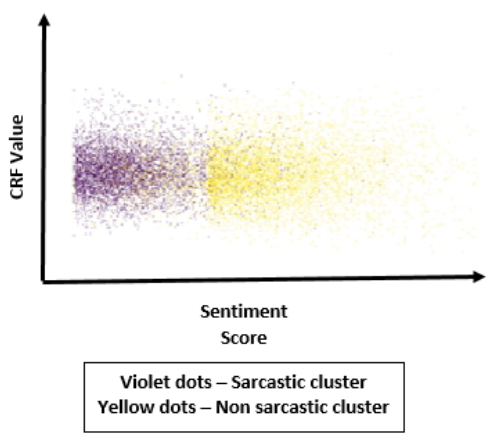


Figure 6. Initial CRF-MEM model based sarcastic tweet distribution

The final output of the proposed model where the initial distribution is classified into two clusters – sarcastic and non-sarcastic is shown in Figure 7. The *violet dots* represent the sarcastic cluster and the *yellow dots* represent the non-sarcastic cluster. The proposed model outperforms the traditional EM algorithm as it achieves a maximum likelihood with minimum number of epochs. The sarcastic tweets obtained from the CRF-MEM model is then fed into the GMM model for grouping into different levels. Based on the covariance, diagonal, full, spherical and tied GMM models are generated for the sarcastic tweets. The various covariance GMM models – diagonal, full, tied and spherical are shown in Figure 8.

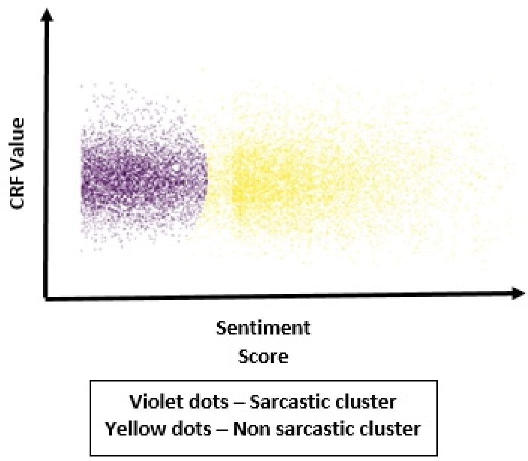
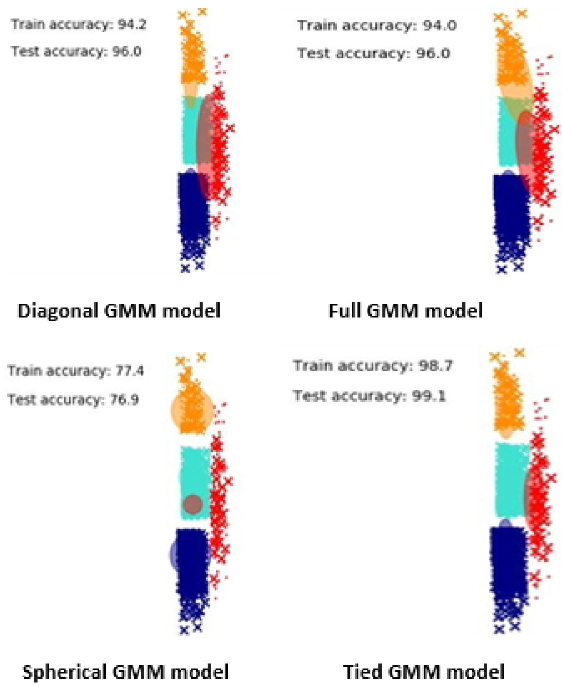


Figure 7. Final result of the proposed CRF-MEM model



Yellow – Low Level; Teal – Medium level;
Blue – High level; Red – Very high level

Figure 8. Results of GMM models

The different colors in the Figure 8 represent tweets belonging to various levels of sarcasm. At preliminary level sarcasm has been classified into four levels: low, medium, high and very high represented by yellow, blue, violet and red colors in the below figure. As this work is a first attempt in categorizing sarcasm into various levels, it is attempted with Gaussian model and the corresponding sentiment score values of tweets. Tweets with comparatively less harshness are represented by yellow crosses and tweets with comparatively high harshness index are represented by red crosses in the Figure 8. Some of the examples are listed below:

Dont you hate when you buy a bag of packaged air and there is chips inside it – Low Level

Don t judge a book by its cover my math textbook has a picture of someone loving maths but i did not enjoy myself – Medium Level

I just love being second choice and being a piece of garbage probably – High Level

Dude everyone hates you Really Because Im pretty sure I haven t even met all 7 billion people on this earth – Very High Level

5 Performance Analysis

The proposed CRF-MEM model is then compared with other EM models which are obtained by changing the distributions. The accuracy obtained for each EM models clearly shows that the proposed model outperforms all the other models. The accuracy comparison chart is shown in Figure 9.

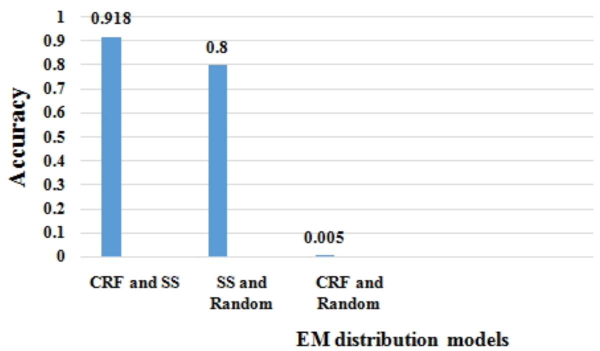


Figure 9. Accuracy comparison chart

Table 1. Evaluation metric for the different types of EM distribution

Model	Accuracy	Precision	Recall	F-score
Proposed CRF-MEM	0.918	0.81	0.99	0.89
SS & Random values	0.8	0.64	0.87	0.74
CRF & Random values	0.005	0.004	0.003	0.004

Table 1 depicts the performance achieved by the proposed method with different models. Out of the three distribution models – proposed CRF and Sentiment score (SS) based distribution, Sentiment score and traditional EM's random assignment distribution and CRF and traditional EM's random assignment distribution – the proposed CRF and Sentiment score based distributions achieves higher accuracy in classifying sarcastic statements.

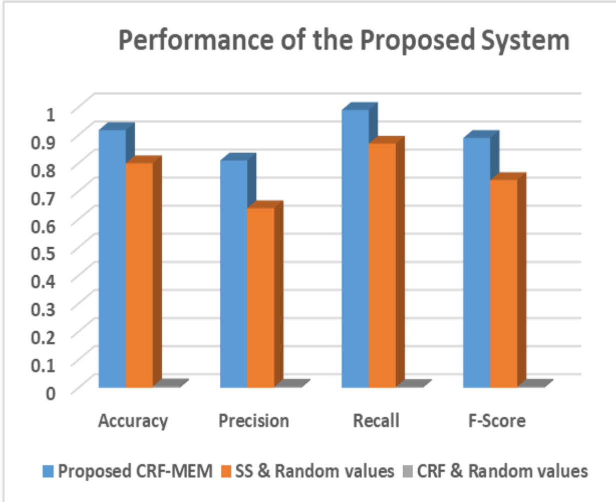


Figure 10. Performance of the proposed CRF-MEM method with other models

Figure 10 represents the performance of the proposed CRF-MEM algorithm with respect to other models. From the performance analysis of various EM distributions, it can be clearly seen that the proposed CRF-MEM model significantly performs better than all the other EM distributions with accuracy value of 91.89%, precision of 81%, recall of 99% and an F – score measure of 89%.

The performance of the proposed work has been compared with some of the related works existing in the literature and the data has been reflected in Table 2.

Table 2. Comparison of the proposed work with existing models

Model	Accuracy (%)
Reyes et al.	90.4
Zarate et al.	85
Bouazizi et al.	83.1
Proposed CRF-MEM	91.8

6 Conclusion

The main objective of this research is to find the levels of sarcasm with the proposed Conditional Random Field model based Modified Expectation Maximization (CRF-MEM) algorithm. As sarcasm is an implicit variety of sentiment, it is highly essential to understand the relationship between each words involved in the sarcastic statement rather than depending on individual words and its meaning. The proposed algorithm has been designed to exploit the power of aspect information for modelling the dependency between words. The traditional EM algorithm achieved an accuracy of 80% for detecting sarcasm. The proposed CRF-MEM model improves the traditional EM algorithm with 91.89% accuracy for the same. Classifying sarcasm into various levels will open the doors for further research which will help in understanding the impact created by sarcasm. In future, various other approaches will be analysed and incorporated for sarcasm detection in order to further improve the performance of the proposed system.

References

[1] K. Ravi, V. Ravi, A novel automatic satire and irony detection using ensembled feature selection and data mining, *Knowledge-Based Systems*, Vol. 120, pp. 15-33, March, 2017.

[2] S. Mukherjee, P. K. Bala, Sarcasm detection in microblogs using Naïve Bayes and fuzzy clustering, *Technology in Society*, Vol. 48, pp. 19-27, February, 2017.

[3] Y. J. Su, C. H. Chen, T. Y. Chen, C. C. Cheng, Chinese Microblog Sentiment Analysis by Adding Emoticons to Attention-Based CNN, *Journal of Internet Technology*, Vol. 21, No. 3, pp. 821-829, May, 2020.

[4] M. D. P. Salas-Zarate, M. A. Paredes-Valverde, M. A. Rodriguez-Garcia, R. Valencia-Garcia, G. Alor-Hernandez, Automatic detection of satire in Twitter: A psycholinguistic-based approach, *Knowledge-Based Systems*, Vol. 128, pp. 20-33, July, 2017.

[5] M. Bouazizi, T. O. Ohtsuki, A pattern-based approach for sarcasm detection on Twitter, *IEEE Access*, Vol. 4, pp. 5477-5488, August, 2016.

[6] A. Reyes, P. Rosso, T. Veale, A multidimensional approach for detecting irony in twitter, *Language resources and evaluation*, Vol. 47, No. 1, pp. 239-268, March, 2013.

[7] F. Colace, M. De Santo, L. Greco, V. Moscato, A. Picariello, Probabilistic approaches for sentiment analysis: Latent dirichlet allocation for ontology building and sentiment extraction, in: W. Pedrycz, S. M. Chen (Eds.), *Sentiment Analysis and Ontology Engineering*, Springer, 2016, pp. 75-91.

[8] K. Sundararajan, A. Palanisamy, Multi-rule based ensemble feature selection model for sarcasm type detection in twitter, *Computational intelligence and neuroscience,* Vol. 2020, pp. 1-17, January, 2020.

[9] M. Y. Manohar, P. Kulkarni, Improvement sarcasm analysis using NLP and corpus based approach, *IEEE International Conference on Intelligent Computing and Control Systems*, Madurai, Tamilnadu, India, 2017, pp. 618-622.

[10] F. Nooralahzadeh, V. Arunachalam, C. G. Chiru, 2012 Presidential Elections on Twitter--An Analysis of How the US and French Election were Reflected in Tweets, *19th IEEE International Conference on Control systems and Computer Science*, Bucharest, Romania, 2013, pp. 240-246.

[11] M. Shrivastava, S. Kumar, A pragmatic and intelligent model for sarcasm detection in social media text, *Technology in Society*, Vol. 64, pp. 1-9, February, 2021.

[12] P. Manivannan, C. K. Selvi, Optimizing Cross Domain Sentiment Analysis Using Hidden Markov Continual Progression, *Journal of Internet Technology*, Vol. 20, No. 3, pp. 781-788, May, 2019.

[13] D. Al-Ghadhban, E. Alnkhilan, L. Tatwany, M. Alrazgan, Arabic sarcasm detection in Twitter, *IEEE International Conference on Engineering & MIS*, Monastir, Tunisia, 2017, pp. 1-7.

[14] Y. Yengi, M. Karayel, An Alternative Method for Sentiment Classification with Expectation Maximization and Priority Aging, *International Journal of Scientific Research in Information Systems and Engineering*, Vol. 1, No. 2, pp. 91-96, December, 2015.

[15] N. M. Shelke, S. Deshpande, V. Thakre, Exploiting expectation maximization algorithm for sentiment analysis of product reviews, *IEEE International Conference on Inventive Communication and Computational Technologies*, Coimbatore, Tamilnadu, India, 2017, pp. 390-396.

[16] A. Onan, S. Korukoğlu, A feature selection model based on genetic rank aggregation for text sentiment classification, *Journal of Information Science*, Vol. 43, No. 1, pp. 25-38, February, 2017.

[17] A. Onan, Topic-enriched word embeddings for sarcasm identification, in: R. Silhavy (Eds), *Computer Science On-line Conference (CSOC), Advances in Intelligent Systems and Computing*, Springer, Cham, 2019, pp. 293-304.

[18] A. Onan, M. A. Toçoğlu, A term weighted neural language model and stacked bidirectional LSTM based framework for sarcasm identification, *IEEE Access*, Vol. 9, pp. 7701-7722, January, 2021.

[19] K. Sundararajan, J. V. Saravana, A. Palanisamy, Textual feature ensemble-based sarcasm detection in Twitter data, in: J. Peter, S. Fernandes, A. Alavi (Eds.), *Intelligence in Big Data Technologies—Beyond the Hype*, *Advances in Intelligent Systems and Computing*, Springer, 2021, pp. 443-450.

[20] K. Sundararajan, A. Palanisamy, Probabilistic Model Based Context Augmented Deep Learning Approach for Sarcasm Detection in Social Media, *International Journal of Advanced Science and Technology*, Vol. 29, No. 6, pp. 8461-8479, July, 2020.

Biographies

Anbarasu Sivalingam is currently working as an Associate Professor of Mathematics in the Department Applied Sciences and Humanities, MIT Campus, Anna University, Chennai. He has more than 15 years of teaching experience and has several Publications in National & International Journals and Conferences.

Karthik Sundararajan is currently an Associate Professor in the Department of Computer Science & Engineering, BNM Institute of Technology, Bangalore. He has 9 years of Teaching & Research experience and his areas of interest includes NLP, AI&ML, Data Science.

Anandhakumar Palanisamy is currently a professor in the Department of Information Technology, MIT Campus, Anna University Chennai. He has around 24 years of Teaching Experience and has published more than 120 articles in International Journals & Conferences. His areas of interest include AI, Image Processing, Robotics & Cloud Computing.