# Detection of inappropriate anonymous comments using NLP and Sentiment Analysis

N. Sai Nikhita [1*], V. Hyndavi [1] and Trupthi M.[2]

Department of IT, Chaitanya Bharathi Institute of Technology, Hyderabad, India

sainikhitanayani@gmail.com

**Abstract—**The world became interactive and socially active now-a-days because of the increase in different types of content sharing applications. These content sharing applications are social media platforms which provide various features so that users can effectively interact and share their thoughts and ideology. One such platform is a discussion forum which promises the anonymous posting of user's views and complaints. As growth In craze of the forum, they are being targeted by spammers for their work. Though these platforms act as medium of knowledge sharing, all of the users don't use these platforms for a positive cause. They are also being used to abuse or bully targeted people taking advantage of their anonymous feature. Spamming and cyber bullying has grown rapidly to a limit that social media is being termed harmful. By reading spam and vulgar comments, readers will be in the diverted and these results in several misconceptions which are harmful. Main aim is to detect these bad comments which are vulgar, inappropriate or not related to the specific context. The research is not based on the static contents but it live streams the comments and the entire research is being done. The research is based on NLP, Sentiment calculation and topic detection.

**Keywords—** PPM Algorithm · TEM Algorithm · SAM Algorithm · Latent Dirichtlet Allocation (LDA) · Natural Language Processing (NLP) · Machine Learning · Topic extraction.

---

N. Sai Nikhita (✉) · V. Hyndavi · Trupthi M.

Department of IT, Chaitanya Bharathi Institute of Technology, Hyderabad, India

e-mail: sainikhitanayani@gmail.com

V. Hyndavi

e-mail: venkatreddygarihyndavi.3@gmail.com

Trupthi M.

e-mail: trupthijan@gmail.com

# 1. Introduction

The social media platforms allow us to interact, share our ideas and opinions by means of sharing posts, tweets, comments and so many other possible ways. By reading comments we get to know the ideology of the people and it is mostly useful in case of online shopping where we try to buy a product by reading the comments and we can come to a view on that product. These comments are allowed to be posted anonymously to get more genuine views. Though we have the access of reading the comments and coming to a decision but they may be spam and they cause irrelevant and irresponsible impact on the readers brain. As we already know the feature in one of the social media site YouTube and the feature is it will delete a comment based on the number of dislikes until and unless reached a particular number. By this action we can understand the real motive is that to not entertain any spam comments. In our research in this paper our approach is to deal not only with the spam comments but also to look after the bad, vulgar and irrelevant comments which manipulate the readers mind and are out of topic which are of no use. The first elimination is based on the removing all the extra spaces and tabs in order to make them into tokens. The above mentioned preprocessing is done only after checking the vulgarity of the comment, also based on the topic relevancy. After checking the above conditions the comments are deleted. Then we deal with topic extraction and topic similarity. We built a mechanism to identify the spam comments and apply the natural processing techniques in the later stages along with the machine learning algorithms.

### 1.1. Objective

With this paper, we intend to create a system where forums, websites and all the social media sites will be spam free. The society now-a-days is fully dependent on social media and it is responsible for changing and routing the behavior of the people. This attention paves a way for the spammers to promote irrelevant content and promote malicious behavior. The main idea of this paper is to provide a system where spam will detected and how the sentiment of the people is depended based on the comments is calculated.

### 1.2. Existing System

In the existing system in now-a-days forums we can see people commenting randomly and we face a number of cases where the comments are

(i)Topic irrelevant
(ii)Which are either offensive or vulgar
(iii)Spam comments

But we can see that in the forum pages all the comments are displayed and no effort is made to remove these comments which are in the above mentioned list until someone reports them to be vulgar. But by the time people will be seeing the comments and it is waste of time to remove them after getting reports.
We may even observe cases where tweets are extracted from twitter and they are further processed by using different techniques to find out the profanity but this is only applied only a particular hashtag or using any username and extracting tweets from that username so this is not efficient in case of common platforms like web forums where people access them everywhere across the world for different reasons. So we are using nlp and different machine learning algorithms to provide a solution to this problem.

### 1.3. Related Work

The initial research in this area was with the recognition of the email spam by Sahami etc. al. [1].He used probabilistic learning methods to produce filters which are useful in this task. Domain specific features are considered for detecting spam. Carreras etc. al. [2] proved that AdaBoost is more effective than Naïve Bayes and decision trees. As increased growth in World Wide Web, the need for spam detection also has grown simultaneously. So they started to detect spam comments by analytics, the decision trees algorithm was used in [3] by Davison to identify link based web-spam. Drost etc. al. [4] used SVM to classify web spam with content based features. Their method is based on the link spam detection and the solution is based on training data by revealing the effectiveness of classes of intrinsic and relational attributes.

In [5] Mishne etc. al. used machine learning techniques the detection of spam comments. Comments in the posts are checked with the comments of linked pages. In [6] Bhattari etc. al. used the data corpus from the same set to detect spam comments. In [7] the authors made study on the comments and commenting behavior of the videos that have more than 6M views and comparison in sentiment analysis is made on finding the influence of the comments by other people's comments. In [8] Ruihai Dong etc. al. worked on topic detection using nlp.

### 1.4. Problem Statement

The online reviews, comments mostly influence the viewers, this behavior can be observed for online shopping, movies, videos and posts in social networking sites. Detection and deletion of the spam comments is big challenge faced by internet now-a-days, which states the need of a system where the comments are streamed continuously instead of taking the static data then to detect whether it is spam or not and based on the result obtained the action should be taking. Further the sentiment of the comments is analyzed to depict the mindset of the people.

## 2. Proposed methodology

The main goal of the paper is to remove spam from all social networking sites. As we know that social media is responsible for influencing the peoples mind and spamming results in change of perspective and is harmful. So we proposed a system where the comments will be streamed from the forums and these comments will be assessed based on the vulgarity using a profanity module.

The preprocessing and later stages will be continued based on the profanity check results and further topic detection and sentiment of the people is calculated.

### 2.1. System Architecture

In our research we have dealt with detecting the spam comments based on NLP and Machine learning. The first is detecting the profanity of the comment then it goes for preprocessing where we tokenize, lemmatize, stemmetize and it continues with other preprocessing finally we reach to find out the topic similarity and topic detection using algorithms. The results are found out using sentiment analysis and then displaying the sentiment of the comments using word cloud and barplot.

Our approach consists of four modules. The first module is about finding the profanity. The second module deals with all the preprocessing. The third module comes with the algorithms where we find out the topic similarity and topic detection by forming dictionary and corpus formation. Finally comes our sentiment analysis.

Generally we can stop the research after detection and blocking of the spam comments but as we developed it for an educational institution we continued it further to calculate sentiment. By doing sentiment analysis we can extract the features of the institution is working and what are the required changes to be made and what students are in need of by calculating the positives and negatives.

The results are visualized using wordcloud. Wordcloud allows us to visualize the most frequently talked issues whose representation or the weight of the word is more.

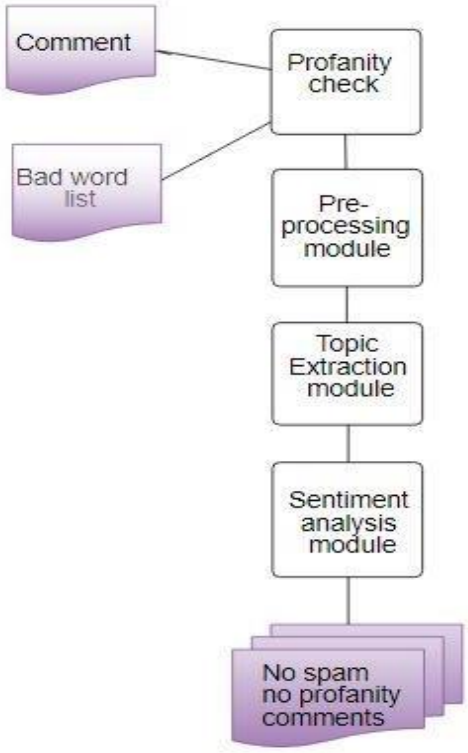


Fig. 1. Proposed Architecture

### 2.2. System Components

A. Profanity Check Module
B. Preprocessing Module
C. Topic Extraction Module
D. Sentiment Analysis Module

The first step we deal here is about profanity check of the comments which can be done in different ways of but we are using some modules here. SentiWordnet is the module which we used where we can check our comments based on the profanity consisting list of words. If the word consisting profanity is found out then we have to stop further preprocessing and as this comment is not for any use in the forum we try not displaying them.

This research then deals with the preprocessing before that they should be in form of tokens. So the comments are split and tokens are formed. Then we see for lemmatization and stemmatization. Then there comes our POS tagging which is a crucial step in the preprocessing. Though we preprocess all the tokens based on POS tagging but our main aim is to find the topic detection and topic similarity so we need only the words which are like nouns and adjectives which are used to describe these nouns. So we focused on these parts of speech After POS tagging we arrive at a stage of topic detection and topic similarity detection.

So the third module deals with TEM algorithm which is depended on the values which are normalized i.e. the sum of the frequencies of the topics from the comment in the post. We understand by topic the main theme which is discussed in the comment and it is given by a set of unigrams and bigrams with a predominant number of occurrences in the comment. Here we applied to LDA to find out the topic similarity. Based on the results of the LDA we can find out the topic which the comments are based on.

Next modules deal with the SAM algorithm which gives the sentiment of the users and finally when we get the results displaying the positivity and negativity of comments and how they affect the further comments on the forum.

*A) PROFANITY CHECK MODULE*

As growth in number of web users the presence of inappropriate content from users becomes more problematic. Social news sites, forums, and any online community must manage user-generated content, censoring that which is not in line with the social norms and expectations of a community. Failure to remove such content not only deters potential users/visitors, but can also signal that such content is acceptable.

The research on previous works shows that the current systems are not up to the mark. The general practice is they use a static list each time they check for profanity. This dot works if the vulgarity is in form of misspelled words, different languages and other reasons. These drawbacks make the current systems to detect profanity obsolete, some even depend on outsiders so that they are assigned with the detection of spam comments for the posts. This is suitable and doable upto a particular stage but when the task becomes huge this is not applicable. So all the comments are profanity checked based on vulgarity, abusive words and irrelevant topic discussion.

(i) LIST BASED APPROACH:

This is the most standard approach where That is, in order to determine if a comment contains profanity in a particular forum, these systems simply examine each word in the document. If any of the words are present on a list of profane terms, then the comment is labeled as profane. Basically we introduced a system where as soon as the comment is introduced in the forum the comment is being checked for the profanity and the profanity module runs in the background and if it is found to be profane we stop further pre-processing. The profanity module is from Google where they update the list on periodic basis and we make sure that the list is updated in our profanity module which takes care of all the spellings, partially censored and other issues taken care of.

*B) PRE-PROCESSING MODULE*

Preprocessing is an important stage in natural language processing because the words, tokens and other sentences identified in this stage are used for further preprocessing to find ngrams and applying algorithms.

(i)STOP WORD REMOVAL:

Many words in a sentence are used as joining words but they themselves do not provide any sense unless combined and framed grammatically to form a sentence. So we can say that their presence do not contribute to the content or the context of the document. Removal of these stop words is necessary because of their high frequency causes obstacles in understanding the actual content of the document.

One can use their own stop word module to detect them and remove but it is not suitable because it is important to update all the stop words and check each word of the sentence to verify and leads to a hectic task. It's better to use stop word modules and try to eliminate the stop words so that we get the actual words which are then taken to the next step for further preprocessing.

(ii)TOKENIZATION:

Tokenization is the process where the sentence is broken into tokens. The main aim behind tokenization is to explore the meaning of the tokens formed and how they are preprocessed further to make meaningful outcomes after performing nlp.

We may have doubt that text is already in readable format then why is the use of tokenizing but still we are left out with many punctuation words and expressions which are needed to be separated. So the main aim of tokenizing is to obtain tokens which are meaning full and remove inconsistency. Tokenizing is based on the delimiter which is further depended on the language where different languages have different delimiters .Space is a delimiter in English where as some languages do not have a particular boundaries like Chinese. So in case of those languages extra care is needed.

(iii)STEMMETIZATION:

The word is the token is reduced to its root word. The root form is not necessarily a word by itself, but it can be used to generate words by concatenating the right suffix. Stemming is a type of normalization. The sequence is returned with its root word

(iv)LEMMETIZATION:

The lemma of the word is found. So we can see that the extra endings are removed in lemmatization. The word which is returned is called lemma. These two terms are not same, Stemming is just finding the root word but most times it's not preferable but lemming is a technique where morphological analysis of the words. It returns a lemma.

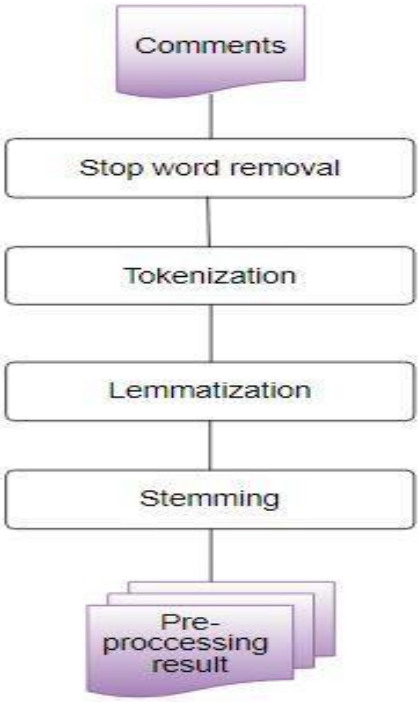


Fig. 2. Flow of Pre-processing module

Algorithm : PPM

Input : Comments entered in the forum
Output : Filtered Tokens

a) Get profanity list from profanity module
b) If comment contains profanity then
    (i)Delete the comment and don't display in the forum
c) Else
    (i)Get the stopwords of English language present in nltk module
    (ii)Remove stopwords present in the comments
    (iii)Tokenize the comment using any tokenizer (eg PunktTokenizer) present in nltk module
    (iv)Stemmatize the tokens
    (v)Lemmatize the tokens using WordnetLemmatizer

Fig. 3. Pseudo code Pre-processing module

*C) TOPIC EXTRACTION MODULE*

The tokenized comments now need to be preprocessed in such a way that we can extract the topic of discussion and categorize the comments based on the topic being extracted.

(i)POS TAGGING:

In this process, for each token that has been formed after the preprocessing, we assign the part of speech to which it belongs. The words are classified based on their part of speech, tense, number, case etc..

The set of all the pos tags used in the POS tagging are called tag sets where they differ from language to language .Basic tagsets are like N for noun ,A for adjective etc. Like that we will have a list of tagsets but each one of them is not useful for topic extraction because as we know that mostly to obtain the topic being discussed in a given sentence or paragraph or document we rely on the nouns being discussed. Not only the nouns but we are also deciding it based on the adjectives and verbs being discussed as they describe the nouns and the situation which is being talked about in a sentence.

(ii)UNIGRAMS AND BIGRAMS:

As we have our pos tagged words further we can group the words based on the distance and then we can conclude what topic is being discussed.

N-grams are sequence of items in a sequence can take values 1, 2 etc. but we no need large value for N. In our project we are generating bi grams to further application. Bi grams are formed by considering adjacent tokens and grouping

them together. Before forming them to bi grams they are in uni grams stage and then we form the bi grams. Unigrams and bigrams are generated as they are essential to proceed towards LDA

(iii)TOPIC EXTRACTION:

As we have our bi grams there is a need to apply an algorithm to extract topic. Here we used LDA(Latent Dirichlet Allocation) to extract the topic .The input to this algorithm needs to be in form of dictionary and corpus. LDA is a topic extraction model which is vastly used for this purpose. This model is used to identify the topic of the document by classifying text which is present in a document. This algorithm is used to build dirichlet distributions (i.e. a topic per document model and words per topic model).

First we try to analyze the frequency of terms (i.e. the number of occurrences of a term present in the document) using a document term matrix. After this has been done we generate a LDA model upon the document. When we apply it, each token is given a unique integer id, which is transformed to a bag of words known as a corpus then the LDA is applied. Based on the frequency of the words obtained topic is extracted. Based on the extracted topic our forum main discussion will be obtained daily basis and weekly basis.

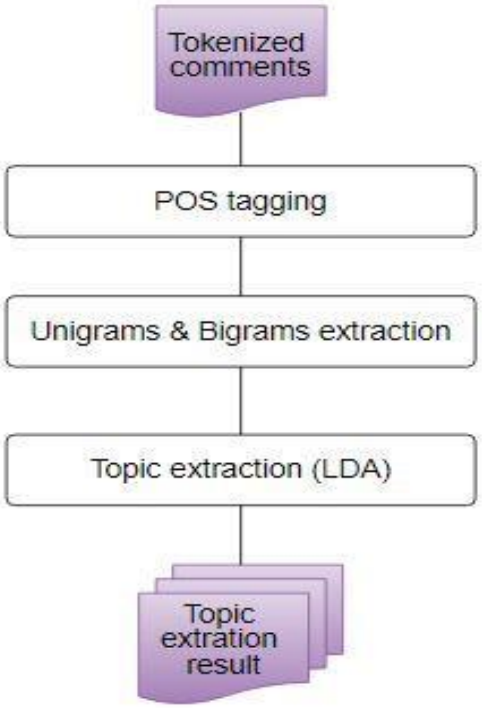


Fig. 4. Flow of Topic extraction module

Algorithm: TEM

Input: Filtered Tokens
Output: Topic of the comment

a) POS Tagging using nltk module
b) Find the candidate nouns (i.e. tokens which are
singular nouns (NN), plural nouns (NNS), singular proper nouns (NNP), plural proper nouns (NNPS))
c) Generate Bi-grams using nltk module
d) Dictionary formation using gensim module
e) Corpus formation using gensim module
f) Apply LDA algorithm using gensim module
g) If topic of the comment is not relevant to the topic
of the forum then
(i)Delete the comment and don't display in the forum
h) Create a wordcloud of most discussed topics using
wordcloud module

Fig. 5. Pseudocode Topic extraction module

*D) SENTIMENT ANALYSIS MODULE*

Sentiment calculation shows the sentiment of the people based on the topic being discussed and how this results in future comments commented and how the people's opinions are based. This is a classification where the inserted phrase is decided based on the negative, positive and neutral sentiment.

In our research we used sentiwordnet. SENTI WORDNET is a document containing all the synsets of WORDNET along with their "positivity", "negativity", and "neutrality". Each synset is has three scores Positive score, Negative score, and Objective score which represent how positive, negative, and "objective" (i.e., neutral) the synset is. These scores may range from 0.0 and go upto 1.0, the sum of all the three scores being 1.0. Each score for a synset term has a non-zero value. SentiWordNetsynset term scores have been computed semi automatically based on a semi-supervised algorithm.

So on the result obtained which shows the sentiment of the phrase describes how the opinion of the people is and also the opinion of the topic being discussed which helps a lot in case of our forum where students will be discussing all their issues which paves a way for the management and the teachers to look after the issues which needed to be taken care and how they need to be handled are also discussed as we provided their suggestions section also so they can reach the staff and be resolved. This system helps not only the faculty and institution but also the students who want their issues to be solved.

Algorithm: SAM

Input: Filtered tokens from preprocessing module, SentiWordnet module which contains synset terms along with their positive and negative sentiment scores

a) For each token in the filtered token list:
    (i)If token= 'not':
        positivescore= 0
        negativescore= thresholdvalue
    (ii)Else:
        positivescore= positive score of synset
          term in sentiwordnet
        negativescore= negative score of synset
          term from sentiwordnet
b) Create a plot visualizing the positive and negative
   sentiments using matplotlib module

Fig. 6. Pseudocode of Sentiment Analysis Module

## 3 Results

The results are showed in the form of plots which are word clouds and barplots which depict the sentiment and they will updated each time the comment is being posted so we can observe the change in the plots and how the sentiment is being changed and how the requirements are to be done and we can observe that topic being discussed upon and the topic which we extracted are evaluated to see how our research has been done.

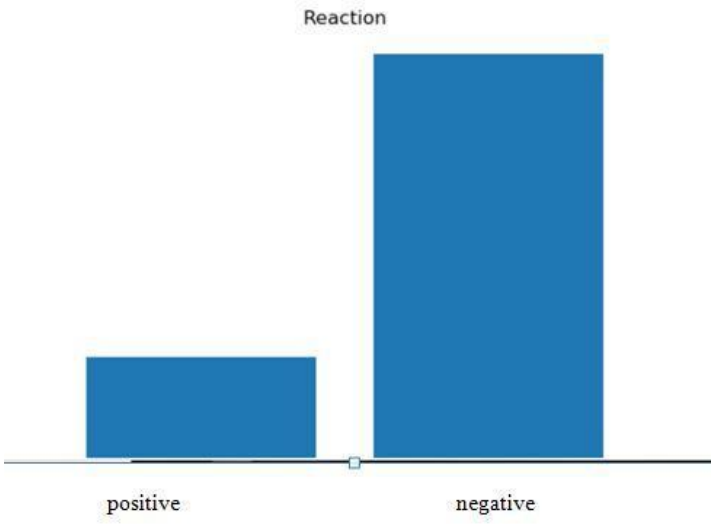


Fig. 7. Barplot visualizing the sentiment

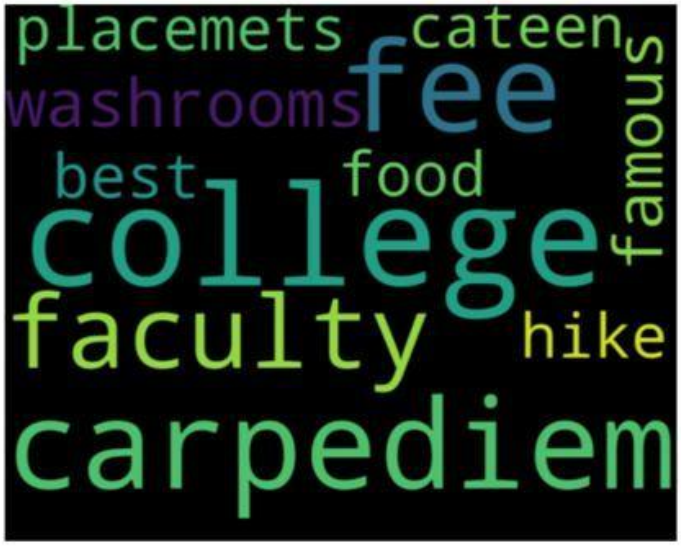


Fig. 8. Wordcloud depicting the most discussed topics

## 4 Conclusion and Future scope

Our research has overcome the problem with some comments and all the disadvantages which were in the existing system and proposed the system where spam comments will be detected based on finding out its features and also the problem where topic irrelevant comments which lead to misconception are also dealt with.

Future enhancements can be made to this project as we are streaming the comments not just taking the static content which provides a great scope not only to remove the spam comments but to make this evaluation of topic to be applicable in other areas of interest

## References

[1] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, “A Bayesian approach to filtering junk e-mail”. In AAAI-98 Workshop on Learning for Text Categorization, July 1998, Madison, Wisconsin, pp. 98-105

[2] Carreras, X. and Marquez, L., “Boosting trees for anti-spam email filtering”. In Proceedings of RANLP-01, 4th International Conference on Recent Advances in Natural Language Processing, 2001, pp. 58-64

[3] Davison B. D., “Recognizing Nepotistic Links on the Web”. In AAAI 2000 Workshop on Artificial Intelligence for Web Search, 2000, pp.23- 28.

[4] I. Drost and T. Scheffer., “Thwarting the nigritude ultramarine: Learning to identify link spam”. In ECML’05 Proceedings of the 16th European conference on Machine Learning, 2005, Berlin, Germany, pp.96-107.

[5] Gilad Mishne, David Carmel, and Ronny Lempel, “Blocking blog spam with language model disagreement”. In Proceedings of the First International Workshop on Adversarial Information Web (AIRWeb), Chiba, Japan, May 2005, pp. 1-6

[6] A. Bhattari and D. Dasgupta, “A Self-supervised Approach to Comment Spam Detection based on Content Analysis”. In International Journal of Information Security and Privacy (IJISP), Volume 5, Issue 1, 2011, pp. 14-32M.

[7] Stefan Siersdorfer and Sergiu Chelaru, “How useful are your comments? analyzing and predicting YouTube comments and comment ratings”. In Proceedings of the 19th international conference on World wide web, 2010, pp. 891-900.

[8] Ruihai Dong, Markus Schaal and Barry Smyth, “Topic extraction from online reviews for classification and recommendation”. In