



Article

Cooking Is Creating Emotion: A Study on Hinglish Sentiments of Youtube Cookery Channels Using Semi-Supervised Approach

Gagandeep Kaur ^{1,†,‡}, Abhishek Kaushik ^{2,*}  and Shubham Sharma ^{3,*} ¹ School of Computing, Dublin Business School, D02 WC04 Dublin, Ireland² ADAPT Centre, School of Computing, Dublin City University, D09 W6Y4 Dublin, Ireland³ School of Food Science and Environmental Health, TU Dublin, D01 HV58 Dublin, Ireland

* Correspondence: abhishek.kaushik2@mail.dcu.ie (A.K.); shubham.sharma@dit.ie (S.S.)

† Current address: 13/14 Aungier St, D02 WC04 Dublin, Ireland.

‡ These authors contributed equally to this work.

Received: 14 May 2019; Accepted: 28 June 2019; Published: 3 July 2019



Abstract: The success of Youtube has attracted a lot of users, which results in an increase of the number of comments present on Youtube channels. By analyzing those comments we could provide insight to the Youtubers that would help them to deliver better quality. Youtube is very popular in India. A majority of the population in India speak and write a mixture of two languages known as Hinglish for casual communication on social media. Our study focuses on the sentiment analysis of Hinglish comments on cookery channels. The unsupervised learning technique DBSCAN was employed in our work to find the different patterns in the comments data. We have modelled and evaluated both parametric and non-parametric learning algorithms. Logistic regression with the term frequency vectorizer gave 74.01% accuracy in Nisha Madulika's dataset and 75.37% accuracy in Kabita's Kitchen dataset. Each classifier is statistically tested in our study.

Keywords: sentiment analysis; hinglish; cookery channels; machine learning

1. Introduction

Youtube is a platform where users can upload, rate, view, share, report, add to favourites, comment on the videos and subscribe to the channels. The content available on Youtube includes TV show clips, video clips, music videos, documentary films, movie trailers, full movies, audio recordings, video blogging and educational videos. Youtube is the world's second largest search engine and third most visited site after Google and Facebook. Every minute on Youtube 400 hours of video are uploaded [1]. People watch over 1 billion hours of Youtube videos a day, more than Netflix and Facebook videos combined [1]. Around 70% of views come from mobile devices. Youtube is present in 80 different languages, covering 95% of the total population present on the internet [1]. Youtube is very famous in India as 225 million people use Youtube monthly [2]. As many Indians live abroad, they find Indian cookery channels an easy medium to learn the basic process of cooking Indian cuisines. This motivated us to investigate the Indian cookery channels and find ways to assist them and support them to understand their viewers. On Youtube people share their thoughts about the video through comments. The useful patterns in unstructured Youtube comments may help the Youtubers to understand the expectations of the user and deliver better content [3]. The current study investigates the patterns and trains machine learning models over the patterns to understand and analyze the viewers' requirements from the video or YouTube.

Sentiment analysis is a study to analyze sentiments, opinions, attitudes, evaluations and the users' emotions, which they convey on social media. A large number of users' comments represent

the current form of the feedback. It is a complex task for humans to get the latest trends and sum up the users' opinions due to the massive amount of data present on social media and this generates the need for real time opinion mining. Deciding the sentiment of the users' comments is a challenging task due to the individuality element which is basically what people think. Sentiment analysis is also considered as the classification task as it classifies the text's orientation. For the classification of sentiments, machine learning techniques are widely used. Machine learning has two parts, supervised learning and unsupervised learning. In Supervised learning the labels are known and the machine tries to map the input to these labels. Unsupervised learning consists of unlabeled input and the machine tries to learn structures from the data.

The challenges in sentiment analysis are subject detection and emotion detection. It is also difficult to find sarcasm or context through the text. Doing the sentiment analysis on multi-lingual languages, such as Hinglish, Chinese, Urdu, etc., is also a great challenge as it mainly addresses mapping the sentiment resources from English to any morphological language. Hinglish is morphologically rich and is a free order language as compared to English, which adds complexity while handling the user-generated content. The scarcity of resources for the Hinglish language brings challenges ranging from collection to generation of datasets. We took up this challenge and work on the Hinglish language.

Hinglish means Hindi language (language of India) written in English script with many words from English vocabulary. For instance, "rice me oil dale to chalega". Here, rice and oil are English words and "me dale to chalega" are hindi words written in English script meaning "is it fine to put oil in rice?" Most of the comments in the datasets are of this type.

The semi-supervised learning technique has been used in our work which includes the unsupervised learning technique as well as the supervised learning technique. The unsupervised learning technique "density-based spatial clustering of applications with noise" (DBSCAN) has been used to label the data into classes and the supervised machine learning classifiers such as random forest, decision trees, multinomial Naive Bayes, Bernoulli Naive Bayes, gaussian Naive Bayes, logistic regression, linear support vector machine, polynomial support vector machine and gaussian support vector machine have been used for sentiment classification on Hinglish datasets. In this study, Hinglish comments are considered, as not much work has been done on the Hinglish language and there are very few datasets available for Hinglish. Moreover, there are no standard stemming algorithm and stopword lists for Hinglish datasets. Indian cookery channels are considered as they are popular and, to the best of our knowledge, this work has not been done previously. This study is divided into five sections: Related work, methodology, results, discussions and conclusion and future work. Related work includes the literature survey done on the different papers. Methodology includes the detail about the dataset and the experimental methodology. In the results section, the results obtained by various machine learning classifiers with different vectorizers are presented. In the discussion section, limitations and findings to our work have been added. In the conclusion and future work section, the work which ought to be done in future is outlined.

Research Questions

The first hypothesis is that the machine learning algorithms work for Hinglish datasets. Based on this hypothesis, the following questions have been formulated.

- RQ1. Which machine learning classifier works best for classifying the Hinglish text?

The second hypothesis is that the patterns in the unstructured comments of viewers from the Youtube channels are useful, and they could be more useful with classification algorithms. The following Research Questions (RQs) have been formulated based on the second hypothesis.

- RQ2. What are the useful patterns in the viewers' comments?
- RQ3. What are the potential capabilities of using machine learning techniques in favour of Youtuber perspectives?
- RQ4. Do we find that the prospective digital approach supports the provider in the long run?

During our study, we investigated the above-mentioned RQs and were able to explore the insight through the current study. The answers to the above mentioned RQs are discussed in Section 5.

2. Related Work

Before starting our investigation, we did background studies. The background studies were divided into five sections:

1. Text pre-processing
2. Text categorization
3. Machine learning
4. Sentiment analysis
5. A study on cookery channels

2.1. Text Pre-Processing

Text pre-processing [4] includes removing the white spaces, punctuation, stopwords and numbers and converting all the letters to either lower or upper case. After pre-processing, feature extraction methods can be applied. There are a lot of feature extraction methods like part of speech (POS), n-gram, bi-gram and bag of words (BoW). Pang et al. did the document classification by using three machine learning classifiers, Naive Bayes, maximum entropy and support vector machine (SVM) classifier, with different types of features like uni-grams, bi-grams and POS; in their study, it was found that SVM is an appropriate tool for handling features sets comprising of bags of uni-gram and bi-gram [5]. Martineau et al. [6] proposed a delta tf-idf technique which gives weight scores to the words before classification. For doing this, SVM as a machine learning classifier with the delta term frequency and inverse document frequency (tf-idf) to improve the accuracy were used for the sentiment analysis problem. Pang, Bo and Lee [7] examined the relation between polarity classification and subjectivity detection. It was found in their study that subjectivity detection compressed the reviews into short extracts but still they maintained polarity information at a level comparable to full text. Subjectivity extracts worked well for the Naive Bayes.

2.2. Text Categorization

Text categorization [8] is an important task of assigning the prefixed labels to the text. There are three types of text categorization:

1. Single-label versus multi-label text categorization
2. Document-pivoted versus category-pivoted categorization
3. Hard categorization versus Ranking categorization

In the single-label text categorization, one category is assigned to the text, while in multi-label more than one category is assigned. Binary text classification is a special case of single-label categorization, where the text either belongs to a category or not. In category-pivoted categorization, the classifier finds all the documents that could be fit under the category. For a given document, the classifier has to find all the categories that could be labelled under the document, and is called document-pivoted categorization [9]. Assigning a probability to an instance is hard categorization. Explicitly assigning a label to an instance is ranking categorization [10]. In our study, we used the multi-label text categorization in which we categorized our data in seven categories. All the seven categories are discussed in detail in Section 3.

Xia et al. [11] use the ensemble framework in their study on movie reviews taken from Amazon. Multi-labelling was done and ensemble framework was applied to increase the accuracy of the classification. Different machine learning techniques like Naive Bayes, maximum entropy and SVM and different feature selection techniques like uni-gram, bi-gram, dependency grammar and joint feature were used in their paper.

2.3. Machine Learning

For the classification of text, machine learning techniques use a training and testing set. The training set has feature vectors and their corresponding labels. A classification model is developed which classifies the feature vectors into the corresponding labels. To validate the model, the test set is used which predicts the labels of unseen feature vectors. There are various machine learning techniques which could be used for sentiment analysis. They are parametric learning models and non-parametric learning models such as Naive Bayes Classifier, Support Vector Machine (SVM), Random Forest, Decision Trees and Logistic Regression.

Density-based spatial clustering of applications with noise (DBSCAN) is a well-known density-based non-parametric clustering algorithm used in machine learning and data mining. DBSCAN is used to group together the points that are close to each other based on Euclidean distance.

The purpose of the DBSCAN algorithm is to find the association between the data points that are hard to find manually and create clusters or groups of points based on the parameters to find patterns in datasets.

Past research that has been done based on the machine learning algorithms is as follows:

Neethu et al. [12] used the Support vector machine (SVM) classifier, Naive Bayes Classifier, maximum entropy classifier and ensemble classifier to find the polarity in the text. It was concluded from their study that all these classifiers gave equal accuracy for their proposed new feature vector using the uni-gram approach. Domingos et al. [13] in their study found out that, for certain problems, Naive Bayes works well for dependent features, contradicting the Naive Bayes main assumption that all the features must be independent.

Gupte et al. [14] used various machine learning models like Naive Bayes, maximum entropy classifier, boosted tree classifier and random forest classifier for doing sentiment analysis. In their study, random forest classifier gave the highest accuracy, though it takes large training time and processing power; still, they considered it as the best classifier for doing the sentiment analysis. Da Silva et al. [15] used the bag of words and feature hashing for extracting the features. The ensemble model formed by multinomial Naive Bayes, random forest, SVM and logistic regression was used for classification. They checked the accuracies of the stand alone classifiers and ensemble classifier with bag of words and feature hashing. Bag of words gave the highest accuracy on their dataset.

Deep Learning

Abdi et al. [16] used a deep learning method called RNSA to classify a user's opinion expressed in reviews. The RNSA employs the Recurrent Neural Network (RNN) which is composed by Long Short-Term Memory (LSTM) to take advantage of sequential processing and overcome several flaws in traditional methods, where order and information about the word are vanished. The datasets were taken from the Movie Review and DUC4 2001 and 2002 datasets. The RNSA is divided into two main parts: Sentiment analysis to extract the useful features in order to determine the sentence polarity and pre-processing which includes the basic linguistic functions. In their study, RNSA (full: word embedding feature (WEF), sentence-level features (SLF), Word-level features (WLF)) obtained the best performance.

Arora and Kansal [17] proposed an architecture which embeds the character level convolutional neural network (CNN) for performing sentiment analysis (SA) of unstructured data and thereby performs text normalization and classification of sentiments. This thereby helps in determining the actual polarity of the text message like whether the text indicates a positive, negative or neutral point of view. While the authors adopted the standard techniques for normalization, like tokenization, lemmatization and stemming, they have also implemented the out of vocabulary (OOV) detection and replacement process. This workflow is aimed at dealing with typos and noisy contents found in the raw text of tweets. Further, this processed data is then combined with the convolutional deep network architecture for performing sentiment classification of raw tweets. The CNN system built in this study is composed of convolutional layer, max pooling layer and a fully connected soft max

classification layer. The input features for this model were the words received after pre-processing, and the Softmax layer is used to receive a multi-class classification of tweets into a three or five scale polarity of categories. This is labelled as strongly positive, weakly positive, neutral, strongly negative and weakly negative. This is indicated by the levels 2, 1, 0, -1 and -2, respectively. Using the accuracy and F-score metrics, a comparative result of accuracy is depicted by showcasing paired t test values for which the proposed approach is paired with the existing methods. These results directly indicate that they perform better than SVM or traditional architectures in multi-class classification of texts. It also deals with fewer parameters to train the CNN network.

2.4. Sentiment Analysis

In their paper, Vinodhi et al. [18] did a survey on opinion mining and sentiment analysis. The major criteria for improving the quality of the services is the user's opinions. Review sites, blogs, data and microblogs provide a better understanding of products and services. Review sites were taken and discussion about the feature selection techniques like uni-gram, bi-grams, dependency grammar, joint feature and tf-idf were done. Different machine learning techniques were applied in order to find their credibility. According to them, finding the sentiments on the movie reviews is a challenging task as most of the users use ironic words in writing the reviews of a movie. Product review is easier than the movie review domain as it is based on the features of the product. Some people may like some features of the product that others do not; so the categorization is easily done in positive and negative. The comparative analysis on movie review and product review is done. From their study, we could say that the combination of different types of classifiers and features could overcome the drawbacks of the individual ones and get benefits from each other's advantages and enhance the performance of the sentiment classification.

Zhang et al. [19] proposed a new entity-level sentiment analysis method. A lexicon-based approach was used for performing the entity-level sentiment analysis. This method gave high precision value but low recall value. In order to improve the recall value, additional texts that would help in opinion mining were selected automatically from the results of the first method. For assigning the polarities to the entities in newly selected text, a classifier was trained. The training samples were given by the lexicon-based approach rather than by labelling manually.

Bilal et al. [20] discussed the results obtained by the Naive Bayes classifier, decision tree and K-NN for the sentiment analysis on the Roman-Urdu text. In their study, the Naive Bayes classifier gave the best result.

Uysal [21] used the different feature selection methods with the supervised classification techniques in the YouTube comments. Sharma et al. [22] used different supervised machine learning classifiers with different feature selection methods on Hinglish text. In their paper, it was observed that the highest accuracy was achieved by using a support vector machine with n-gram at 95.07% followed by Naive Bayes with n-gram at 94.45%. Chi-square feature selection method was employed on our data but it did not give positive results; so we did not include the feature selection method in our research.

Timoney et al. [23] did sentiment analysis on the Youtube videos of the top songs from the British chart since 1960. Only two machine learning techniques: Naive Bayes and Decision Trees were applied in their work. It was observed in their paper that decision trees gave higher accuracy of 86.09% followed by Naive Bayes with 79%. In their paper, only two machine learning techniques were used but in our work we used many machine learning techniques covering both the parametric as well as non-parametric techniques.

Trinto et al. [24] used the Bangla, English and Romanized text for the sentiment analysis. They used the two groups of classes; the first was positive, negative and neutral, and the second was strongly positive, positive, neutral, negative and strongly negative. In their paper, three class multi-label datasets achieved more than 10% accuracy from the baseline approach. In our work, we categorized our data into seven labels.

2.4.1. Sentiment Analysis on Hinglish

Ravi Kumar and Ravi Vadlamani [25] used the different feature selection methods like information gain, gain ratio, chi-squared and correlation on Hinglish Facebook comments. Different supervised machine learning classifiers with TF-IDF vectorizer were used in their paper. They got the best accuracy of 86% by using the combination of TF-IDF, gain ratio and radial basis function neural network. Kaur et al. [26] did dictionary-based sentiment analysis of Hinglish text. Hinglish comments on movie reviews from different sources were taken in their study. Two different dictionaries were made based on English and Hindi language. A stopwords-removal list was created and some pre-processing techniques were done in their study.

2.4.2. Sentiment Analysis Using Semi-Supervised Approach

Khan et al. [27] performed the sentiment analysis on the English movie data and Amazon product review data using the semi-supervised approach. Lexicon-based methodology was combined with machine learning to improve the performance of the sentiment analysis in their study. In SentiWord-Net, the senti scores were revised using the cosine and gain similarity. A comparison between the proposed technique with the state-of-the-art techniques was carried out, proving that the proposed technique is better than other techniques. Silva et al. [28] did a survey and comparative study on sentiment analysis of the Twitter comments by using semi-supervised approach. Different methods, like graph-based, wrapper-based, and topic-based methods, for labelling the data were compared in their work. Support vector machine with linear kernel was used in their work in the classification process. According to their study, the self-training approach is considered to be best when significant amounts of data are available. In addition, it was observed to be more useful when irony and sarcasm are present.

2.5. A Study of Cookery Channels

Benkhelifa et al. [29] discussed the opinion extraction and classification of real-time Youtube cooking recipe comments. A real-time system was proposed in their study, which automatically extracts and classifies the Youtube cooking recipes. After collecting the data, it filtered the comments and classified the comments into positive and negative by using the model built by the SVM classifier.

Bianchini et al. [30] proposed PREFer, which recommends menus on the basis of the user's preferences using the recipe dataset and annotation. Here, any choice made by the user automatically generates recommendations that might affect the user's health. Filtering algorithms that help in recommending things to the users were used in their study.

Pugsee et al. [31] did the sentiment analysis on the food based on the SentiWordNet. Polarity lexicon was generated after collecting the subjectivity words about the food. They proposed a tool that analyzes much content on the recipe's comments text. This helps the user to make their decision about the food recipe.

Yu et al. [32] proposed a method that helps in predicting the user ratings of online recipes. Information about the ingredients of the recipe, instructions to make the recipe and reviews are taken into consideration. The multi-class SVM was used to examine how reliable those pieces of information are. In their study, it was found that the information about the reviews gave the most reliable predictions.

As per the literature review studied, more work has been done on the English language, whereas very few studies have been done on the Hinglish language. Moreover, the studies have been primarily done on news channels and political channels but not on cooking channels on Youtube. Furthermore, other researchers worked on finding the spam or non-spam comments and negative or positive comments, whereas our study targeted finding different patterns in comments. This makes our work novel, as we have found the patterns in Hinglish text that indicate the viewers' expectations using the unsupervised clustering technique and those patterns further used for classification using

supervised machine learning techniques. The below table shows the different methods, languages and datasets used (Table 1).

Table 1. Literature survey table.

Authors	Language	Datasets	Methods Used	Results
Neethu et al. [12]	English	Twitter posts about electronic products	SVM, Naive Bayes, MaxEnt and ensemble classifier	90 (SVM, MaxEnt and ensemble classifier)
Domingos et al. [13]	English	Multiple sources	Naive Bayes	96.9 (Naive Bayes)
Da Silva et al. [15]	English	Twitter posts about products	NB-M, RF, SVM and LR	84.89 (Ensemble with lexicon)
Zhang et al. [19]	Cantonese	Internet Restaurant reviews	Naive Bayes and SVM	95.6 (Naive Bayes)
Bilal et al. [20]	Roman-Urdu and English	Blogs	Naïve Bayes, CART and KNN	97.50 (Naive Bayes)
Sharma et al. [22]	Hinglish	Whatsapp	Different machine learning and feature selections	95.07 (SVM with n-gram)
Timoney et al. [23]	English	Youtube videos	Naive Bayes and CART	86.09 (CART)
Trinto et al. [24]	Bangla, English and Romanized text	Youtube comments	3 class multilabel and 5 class multilabel	65.97 (3 class multilabel)
Ravi Kumar and Ravi Vadlamani [25]	Hinglish	Facebook comments	Information gain, gain ratio, chi-squared and correlation with supervised machine learning algorithms	86 (TF-IDF, gain ratio and radial basis function neural network)
Kaur et al. [26]	Hinglish	Movie reviews from different sources	Classification algorithms	Prospective future work
Khan et al. [27]	English	Movie data and Amazon product review data	Lexicon based with the machine learning	86.50 (Senti-Cosine with model selection)
Silva et al. [28]	English	Twitter comments	Graph-based, wrapper-based, and topic-based methods with SVM-L	67.34 (SVM-L)
Benkhelifa et al. [29]	English	Youtube cooking recipes comments	Proposed algorithms for the subjectivity and sentiment classification	95.3 (Sentiment classification)
Bianchini et al. [30]	English	Recipe dataset	Filtering algorithms	PREFer system is developed
Pugsee et al. [31]	English	Food data	SentiWordNet	Tool is made for the user to make their decision about the food recipe
Yu et al. [32]	English	Online recipes	Multi-class SVM	Reviews information is reliable
Pang et al. [5]	English	Movie review	Naive Bayes, MaxEnt, SVM with uni-grams, bi-grams and POS	82.9 (SVM with unigram)
Pang, Bo and Lee [7]	English	Movie review	Naive Bayes, MaxEnt, SVM	86.4 (Naive Bayes)
Xia et al. [11]	English	Movie reviews from Amazon	Naive Bayes, MaxEnt, SVM with feature selection techniques like uni-gram, bi-grams, dependency grammar and joint feature	88.65 (WR-based ensemble)
Martineau et al. [6]	English	Movie review data	SVM with delta tf-idf	99.8 (Subjectivity classification)

3. Methodology

In this section, the methodology used for the sentiment analysis is discussed. The methodology is divided into various sections as shown in Figure 1.

1. Data gathering: The data was gathered from the Youtube API. The top two cookery channels, named Nisha Madhulika Cooking Channel and Kabita's Kitchen, were taken.
2. Preprocessing: Preprocessing was done after gathering the data. Preprocessing includes the removal of stopwords, null values, numbers, special characters and punctuation, converting the document into lower case, tokenization and stemming.
3. Clustering techniques: Clustering was done on our dataset to label the data. DBSCAN was used to cluster the data and then categorizations were made from the clusters.
4. Sentiment categorization: The seven categorizations, as shown in Table 2, were made using the thematic analysis.

5. Machine learning: Machine learning techniques were employed on the dataset. Cross validation was done on 70% of the training dataset and testing was done on the remaining 30% test dataset.
6. Resulting opinion: We got the validation score after applying the machine learning models to the test dataset.
7. Statistical testing: Statistical testing was done on the training score to be assured that the results were not got by chance.

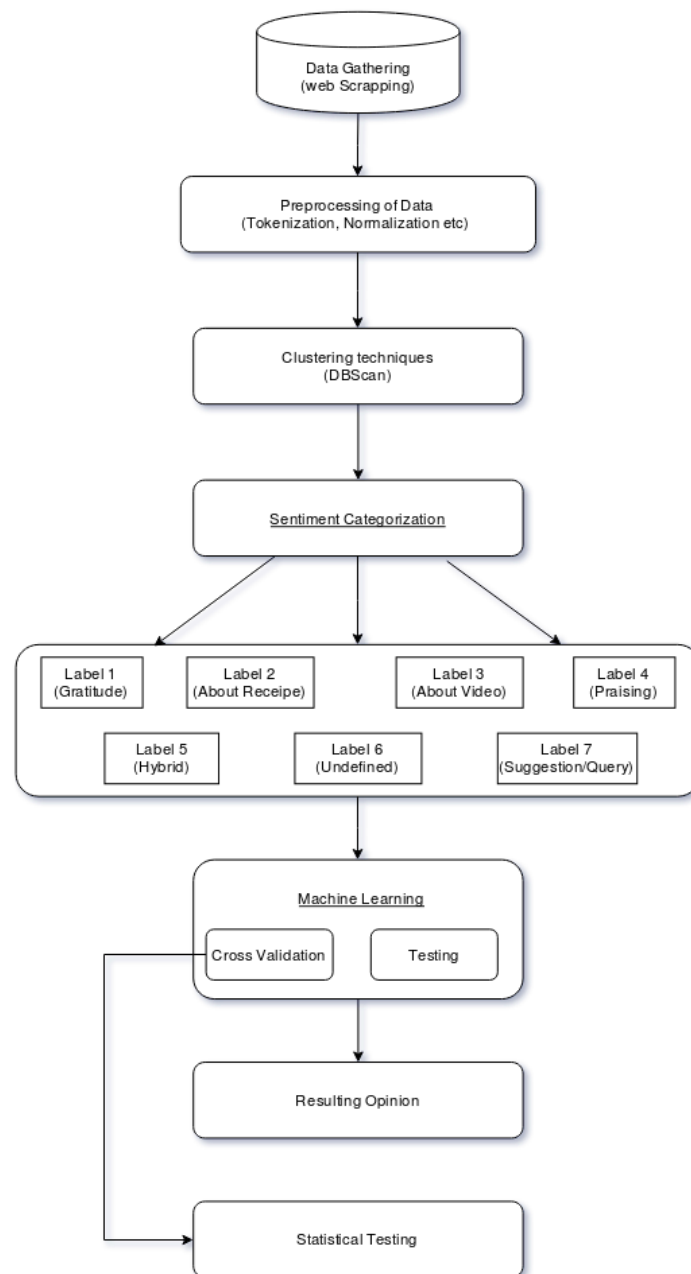


Figure 1. Conceptual diagram showing the methodology.

3.1. Datasets

The datasets were collected from the Youtube through its API in march 2019 [33]. The top two cookery channels of India, named Nisha Madhulika Cooking Channel [34] and Kabita's Kitchen [35], were chosen.

On each dataset, the comment text section is present. This comment text section includes the comments of the user. After looking at the comments in the comment text section in the dataset, we realized that there were no spam comments, as most of the users come on cookery channels to see the cookery videos only. However, we found that this dataset is good enough to do the further analysis. Therefore, the unsupervised technique density-based spatial clustering of applications with noise (DBSCAN) clustering was employed to cluster the comments present in the comments text section. After doing the clustering, two voluntary coders coded the dataset independently using the thematic analysis. At the time of the conflict or confusion, they used the Cohen's kappa coefficient. The data was categorized into seven labels. The data was manually labelled as per the categories, as shown in Table 2.

Table 3 presents the datasets collected and used in the experiments reported in this paper, along with the amount of samples in each class and the total number of samples.

The dataset was categorized into seven labels with an equal number of samples, 700, each. The total sample of all the datasets is 9800.

Table 2. Categories of comments.

Labels	Description
Label-1 (Gratitude)	This gives a description about gratitude. Here the users show their gratitude to the chef. For instance: thank aunty g, thank you, thank you much, thank you so much madam, thank mem, thank u much mem.
Label-2 (About recipe)	This gives a description about the recipe. Here the users express their views about the recipe, whether it is good, tasty, delicious, etc. For instance: yummy, very delicious, delicious, yummy nice one, nice yummy, very nice yummy, very testi, very testy, testy, tasty, so tasty, very tasty, very nice recipe mam, nice recipe.
Label-3 (About video)	This gives a description about the video. Here the users express their views about the video, whether the video is good or not, long or short, etc. For instance: nice, superb, awesome, wow, nyc, super mam, very nice mam, super, nice aunty, wow nice mam, nice video, very nice, good, so good, nice Nisha mam, mast, nic, really nice, awesome nisha ji, wonderful, nice mem, nyc aunty ji.
Label-4 (Praising)	This gives a description about praising the chef. Here the users express their admiration to the chef. For instance: you sweet, sweet, r great, great, u r good mam, u best, u amazing, u r awesome mam.
Label-5 (Hybrid)	In this label we combined two or more labels. Suppose users are expressing their gratitude and admiration to the chef, then it is labelled as hybrid. For instance: thank you for this nice recipe!!
Label-6 (Undefined)	Those comments made by the user which are not defined in any category are kept in this label. Here the user is not talking about the recipe, video or not paying gratitude to the chef. They are also not praising the chef or asking any questions of the chef. For instance: please reply mam.
Label-7 (Suggestion or queries)	This label describes the questions asked by the users. Here users either ask for suggestions or put their queries about the recipe to the chef. For instance: Which flour to be used? What is the substitute for this or that? What if we do this or that?

Table 3. Data comments distribution.

Dataset	Label 1	Label 2	Label 3	Label 4	Label 5	Label 6	Label 7	Total Comments
Nisha Madhulika	700	700	700	700	700	700	700	4900
Kabita's Kitchen	700	700	700	700	700	700	700	4900

3.2. Experimental Methodology

3.2.1. Preprocessing

Our investigation was done only on the comments. Keyword extraction is not easy in Youtube due to the misspellings present in the comments. In order to avoid this problem we did some preprocessing on both the datasets, which included removal of stopwords, null values, numbers, special characters and punctuation, converting the entire text into lower case, tokenization (creating tokens from sentences) and stemming (eliminate the tense and repeated words from sentences).

Tokenization is the process of splitting the text into tokens by removing commas, white spaces, etc. The numbers were removed from the comments text as they are of no use. Stopwords refer to the most common words in the English language like 'is', 'the', 'at', etc. For removing the stopwords in the Hinglish language, we created the stopwords list containing the Hinglish stopwords like 'hain', 'yeh', etc. We removed the Hinglish stopwords from the comments text as they do not play any positive role in the sentiment analysis. Stemming reduces the tokens that are relevant to a single type. There are no standard stemming algorithms for the Hinglish language. Therefore, Porter stemmer algorithm, which is widely used for the English language, was employed assuming that it would work for Hinglish text.

3.2.2. Clustering Techniques

Clustering is a process of grouping all the objects together that are similar. Clustering was done on our dataset to label the data. The k-means clustering was employed by giving different values of k (number of clusters) on the dataset. By doing this we could not find any useful results; therefore, the DBSCAN algorithm was employed. In the DBSCAN algorithm there is no need to provide the number of clusters to the model. By doing this we got 80 clusters and out of 80 clusters seven categories were made, as described in Section 3.1.

3.2.3. Bag of Words

Machine learning algorithms do not work with text data. So, there is a need to convert the text into vectors known as feature extractions. A popular method used for feature extraction in text is called bag of words (BoW). The bag of words model is a way of presenting the textual data while modelling the text with the machine learning algorithms. BoW is considered to be best for classification. The BoW model can be built by using

1. Count occurrence: This counts the number of times each word token appears in the document. The reason behind the usage of this approach is that keywords or important signals occur repetitively. The importance of the word is represented by the number of occurrences of that word. The higher the frequency, the more important.
2. Term frequency and inverse document frequency: In this approach, it is assumed that high frequency might not provide much information gain. In other words, more weight is contributed to the model by rare words. In tf-idf, words that appear regularly in few documents are given the highest rating and words that appear regularly in every document are given the lowest rating.
3. Term frequency: Term Frequency (TF) is simply the ratio of the occurrence of each word token to the total number of word tokens in the document. The condition becomes more important for the summary presentation when the term has higher frequency.

We then employed feature vectors like count vectorizer, tf-idf vectorizer [20] and term frequency vectorizer to convert the text into vectors.

3.2.4. Building the Machine Learning Model

The supervised classification is categorized into parametric and non-parametric learning algorithms. In the parametric learning algorithms, the number of features are fixed, whereas in non-parametric learning algorithms, the number of features are infinite. The number of features in the non-parametric learning algorithms grows when the training data increases. Examples of parametric learning algorithms are logistic regression, Naive Bayes and linear support vector machine, whereas examples of non-parametric learning algorithms are decision trees and gaussian support vector machines. In our work, both the parametric as well as the non-parametric learning algorithms are covered.

In the Table 4, the classification models which were selected are shown.

Table 4. Applied machine learning techniques.

Algorithm	Classification Techniques
CART	Decision tree
LR	Logistic regression
NB-B	Bernoulli Naïve Bayes
NB-G	Gaussian Naïve Bayes
NB-M	Multidimensional Naïve Bayes
RF	Random forests
SVM-L	Support vector machine with linear kernel
SVM-P	Support vector machine with polynomial kernel
SVM-R	Support vector machine with gaussian kernel

Cross validation was performed on the training data and the accuracy of the model was evaluated on the test data. If the training score is high and validation score is low, then the model is overfitting. If the training score is low and validation score is high, then the model is underfitting; 10 k-folds cross validation was done on the 70% of the training data. From the training scores we got, we can say that our model is well-generalized. It is neither underfitted nor overfitted.

For comparing the different algorithms, well-known measures like accuracy (ACC), F1-score, precision, recall and Matthews Correlation Coefficient (MCC) were taken.

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{[(TP + FP) * (FN + TN) * (FP + TN) * (TP + FN)]}} \quad (1)$$

In order to find the best parameter for our algorithm, we employed the grid search. For analyzing, we used 10^i as the search range, where i is between -3 to 3 ; α (alpha) for Bernoulli Naive Bayes and multinomial Naive Bayes; C for linear SVM and LR; and C and γ (gamma) for gaussian SVM and polynomial SVM. The number of trees of the RF technique was fitted with the search range 10 to 100, with a step size of 10. The best values found for each dataset are reported in Table 5.

The pre-processing on the dataset, classification algorithms, grid search and experiments were implemented and performed in Python 3.7.0 [36] using scikit-learn v.0.20.3 library [37]. All other parameters that were not set by grid search kept with their default values. For reproducibility purposes, the seed of the random number generator for random forests and decision trees was set to 0.

Table 5. Parameters chart.

Method	Parameter	Nisha Madhulika	Kabita's Kitchen				
		tf-idf	Count	Term	tf-idf	Count	Term
LR	C	10	10	10	10	10	10
NB-B	α	0.1	0.1	0.1	0.1	0.1	0.1
NB-M	α	1.0	0.1	0.1	1.0	0.1	0.1
RF	# trees	80	50	100	100	100	100
SVM-L	C	1.0	1.0	1.0	1.0	1.0	1.0
SVM-P	C	0.1	0.001	0.001	0.01	0.1	0.01
	γ	100	10	100	10	10	100
SVM-R	C	1000	100	100	1000	1000	1000
	γ	0.001	0.01	0.001	0.001	0.001	0.1

4. Results

The results were obtained by different classification algorithms over both the datasets using different feature vectorizers like tf-idf vectorizer, count vectorizer and term frequency vectorizer.

In Table 6, it is shown that SVM linear kernel (SVM-L) with the tf-idf vectorizer has the highest accuracy of 73.74% and precision of 75.15%. It is very close to logistic regression (LR) with tf-idf vectorizer with 73.46%. SVM gaussian kernel (SVM-R) with the count vectorizer has the highest accuracy of 73.40% and precision of 74.11%, followed by logistic regression with 72.65%. Logistic regression with term frequency vectorizer gave the highest accuracy of 74.01% and the precision of this is 74.70%.

Table 6. Validation results on Nisha Madhulika's dataset.

Vectorizer	Algorithm	Accuracy	F1-Score	Recall	Precision	MCC
Tf-IDf	RF	70.61	70.61	70.61	71.63	65.89
	NB-M	70.27	69.57	70.27	71.13	65.64
	NB-B	67.75	66.22	67.75	66.50	62.63
	NB-G	54.96	52.83	54.69	52.10	47.27
	LR	73.46	73.73	73.46	74.54	69.15
	SVM-L	73.74	73.94	73.67	75.15	69.48
	SVM-P	60.95	62.84	60.95	72.33	56.69
	SVM-R	73.33	73.42	73.33	74.03	68.98
	CART	64.89	64.88	64.89	65.28	59.12
Count vectorizer	RF	68.43	69.21	69.52	70.24	64.64
	NB-M	69.76	69.02	69.79	69.30	64.90
	NB-B	67.75	66.22	67.75	66.50	62.63
	NB-G	55.44	53.47	55.44	53.50	48.38
	LR	72.65	72.77	72.65	73.26	68.15
	SVM-L	72.44	72.48	72.44	73.07	67.95
	SVM-P	64.76	64.52	64.76	68.64	59.69
	SVM-R	73.40	73.56	73.40	74.11	69.03
	CART	66.59	66.45	66.59	67.27	61.20
Term frequency vectorizer	RF	72.31	72.41	72.31	72.88	67.75
	NB-M	71.97	71.62	71.97	71.97	67.40
	NB-B	68.43	67.07	68.43	67.36	63.40
	NB-G	56.12	54.16	56.12	53.91	49.03
	LR	74.01	74.04	74.01	74.70	69.79
	SVM-L	73.26	73.47	73.26	74.57	68.98
	SVM-P	67.27	68.89	67.27	74.55	63.03
	SVM-R	73.87	73.95	73.87	74.44	69.59
	CART	64.55	64.94	64.55	65.80	58.74

In Table 7, support vector machine linear kernel with tf-idf vectorizer achieved the highest accuracy of 75.30% and precision of 76.56%, followed by support vector machine gaussian kernel with 74.96% accuracy. Support vector machine linear kernel with count vectorizer achieved the highest accuracy of 74.55% and precision of 75.95%. Logistic regression with the term frequency vectorizer achieved the highest accuracy of 75.37% and precision of 76.19%.

From Figures 2 and 3, it is seen that, in the dataset of Nisha Madhulika, logistic regression with term frequency gave the best accuracy 74.01%. In the Kabita's Kitchen dataset, logistic regression with term frequency yielded the best accuracy with 75.37%. From these results, we could say that logistic regression worked well with term frequency on our dataset.

Table 7. Validation results on Kabita's Kitchen dataset.

Vectorizer	Algorithm	Accuracy	F1-Score	Recall	Precision	MCC
Tf-IDf	RF	72.24	72.56	72.24	74.51	67.96
	NB-M	71.29	70.72	71.29	72.10	66.76
	NB-B	66.53	65.28	66.53	66.46	61.35
	NB-G	49.59	48.32	49.59	49.41	41.57
	LR	74.62	74.74	74.62	75.63	70.53
	SVM-L	75.30	75.41	75.30	76.56	71.38
	SVM-P	63.53	65.48	63.53	73.38	59.35
	SVM-R	74.96	75.03	74.96	75.77	70.91
	CART	67.82	67.84	67.82	68.85	62.62
Count vectorizer	RF	70.61	70.08	70.61	73.32	62.62
	NB-M	69.31	68.67	69.31	68.98	64.32
	NB-B	66.53	65.28	66.53	66.46	61.35
	NB-G	47.61	46.47	47.61	48.84	39.34
	LR	73.67	73.71	73.67	74.45	69.39
	SVM-L	74.55	74.63	74.55	75.95	70.51
	SVM-P	64.96	64.25	64.96	70.91	60.33
	SVM-R	74.01	74.03	74.01	75.15	69.85
	CART	67.82	67.74	67.82	69.48	63.74
Term frequency vectorizer	RF	72.31	72.61	72.31	73.94	67.90
	NB-M	71.29	70.90	71.29	71.16	66.58
	NB-B	66.87	65.64	66.87	66.70	61.71
	NB-G	50.00	48.87	50.00	50.32	42.05
	LR	75.37	75.42	75.37	76.19	71.38
	SVM-L	74.21	74.20	74.21	75.09	70.06
	SVM-P	69.38	70.53	69.38	75.12	65.29
	SVM-R	74.48	74.64	74.48	75.88	70.43
	CART	66.87	66.98	66.87	67.58	61.41

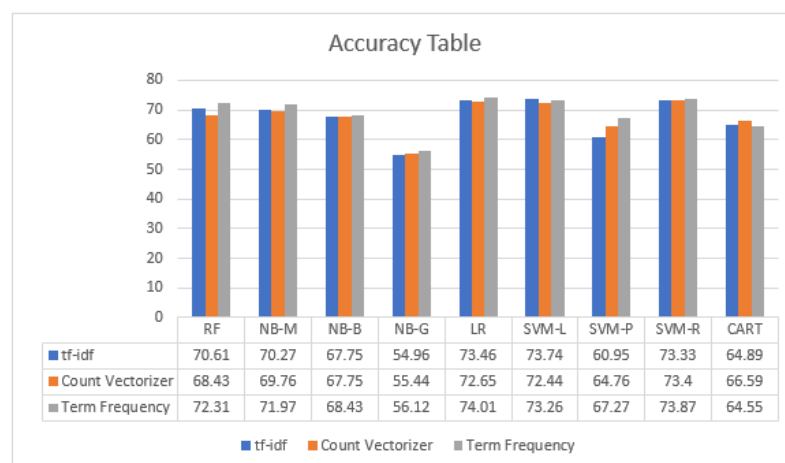


Figure 2. Bar graph showing accuracies on Nisha Madhulika's dataset using tf-idf, count and term frequency vectorizer.

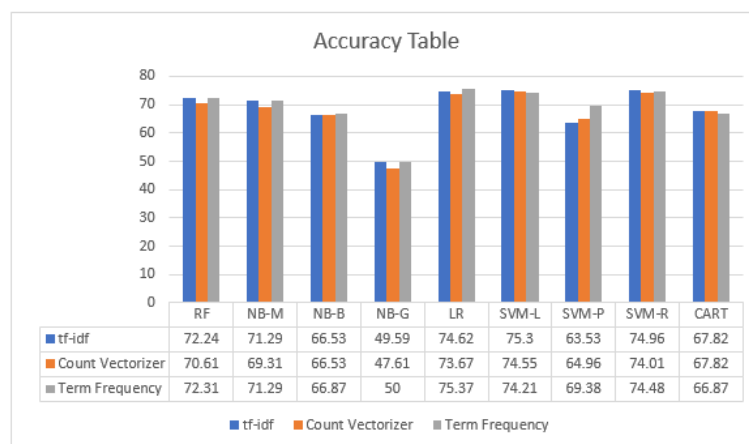


Figure 3. Bar graph showing accuracies on Kabita's Kitchen dataset using tf-idf, count and term frequency vectorizer.

Statistical Testing

In order to ensure the results obtained from different classifiers are accurate and are not produced by chance, Friedman statistical testing [38] was performed. According to Friedman statistical testing, the null hypothesis assumes there is no significant difference between the performances achieved by the evaluated classifiers. The other hypothesis assumes there is significant difference between the performances achieved by the evaluated classifiers. Here, for the Nisha Madhulika dataset, we got a p-value less than the value of $\alpha = 0.001$, this means there is statistically significant difference among the classifiers with 99.9% confidence level. After the rejection of the null hypothesis of the Friedman test, least significant difference (LSD) test was done as shown in Figures 4 and 5.

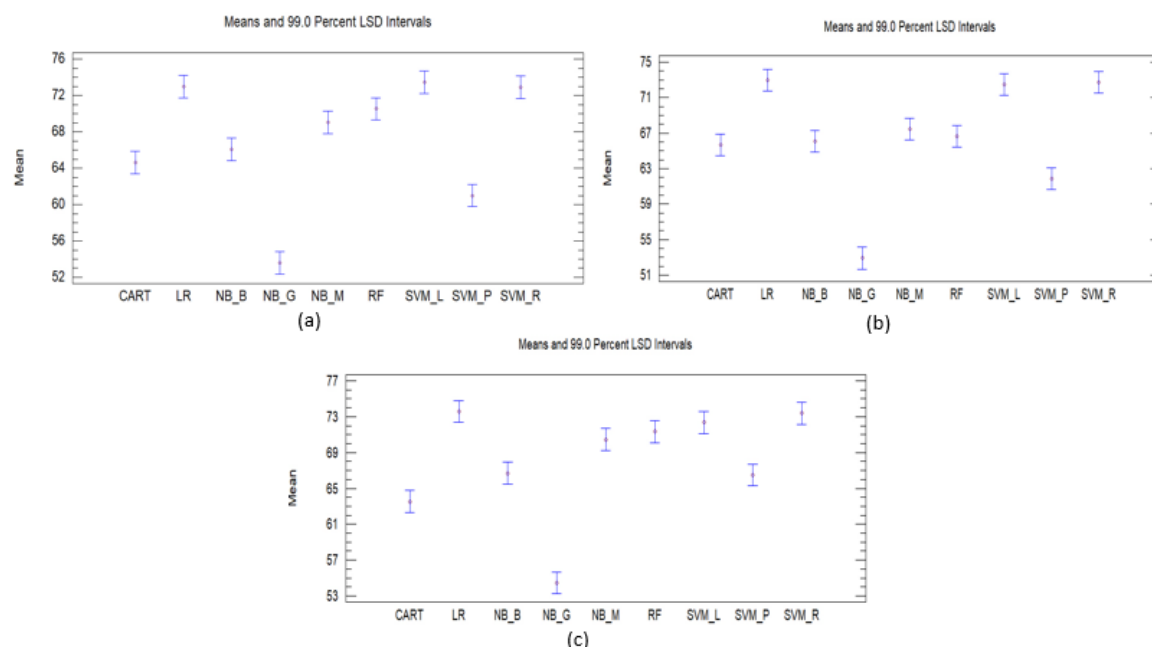


Figure 4. Box plot showing least significant difference (LSD) results on Nisha Madhulika's dataset using (a) tf-idf, (b) count and (c) term frequency vectorizer.

In Figure 4, (a) shows the LSD results on Nisha Madhulika's dataset using tf-idf vectorizer, (b) shows the LSD results using the count vectorizer and (c) shows the LSD results using the term frequency vectorizer. From the results we got on Nisha Madhulika's dataset, we can say that decision trees and Bernoulli Naive Bayes, Bernoulli Naive Bayes and multinomial Naive Bayes, and multinomial

Naive Bayes and random forest are statistically equivalent ($p < 0.001$). Random forest, gaussian SVM, logistic regression and linear SVM are statistically equivalent ($p < 0.001$). After employing the least significance difference on Nisha Madhulika's dataset (using count vectorizer), we can say that decision trees, Bernoulli Naive Bayes, random forest, multinomial Naive Bayes and linear SVM, gaussian SVM and logistic regression are statistically equivalent ($p < 0.001$). Results on Nisha Madhulika's dataset (using term frequency vectorizer) show that decision trees, polynomial SVM and Bernoulli Naive Bayes are statistically equivalent. multinomial Naive Bayes, random forest, linear SVM, gaussian SVM and logistic regression are statistically equivalent ($p < 0.001$).

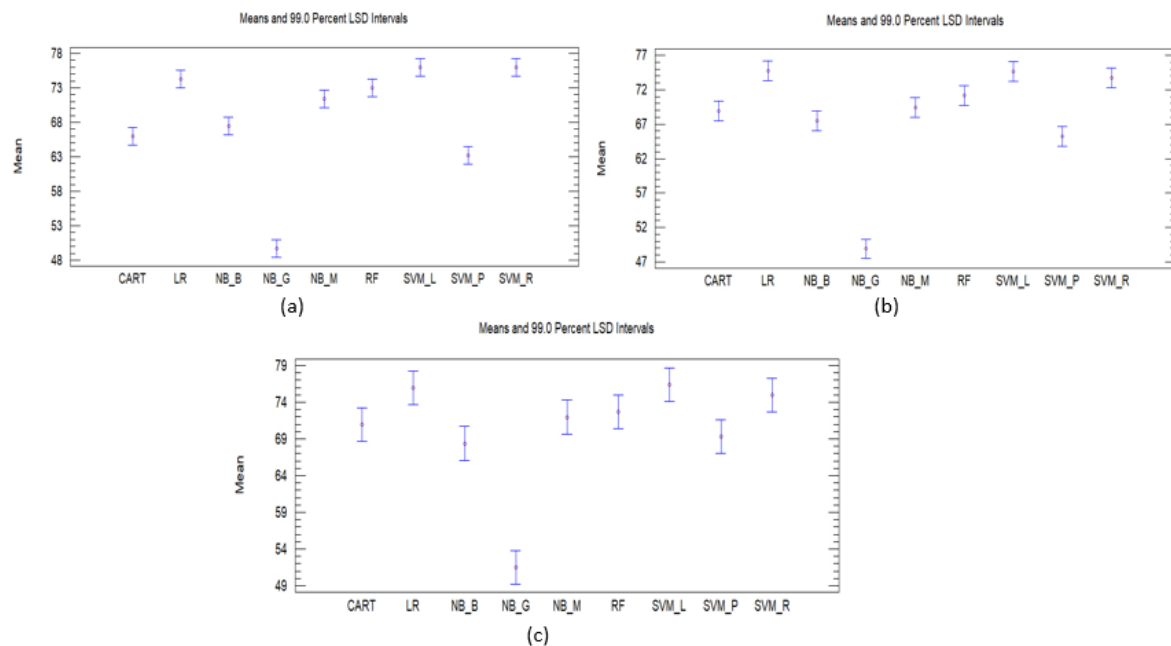


Figure 5. Box plot showing LSD results on Kabita's Kitchen dataset using (a) tf-idf, (b) count and (c) term frequency vectorizer.

In Figure 5, (a) shows the LSD results on the Kabita's Kitchen dataset using tf-idf vectorizer, (b) shows the LSD results using the count vectorizer and (c) shows the LSD results using the term frequency vectorizer. From the results we got on the Kabita's Kitchen dataset (using tf-idf vectorizer), we can say that decision trees and polynomial SVM, Bernoulli Naive Bayes and decision trees are statistically equivalent ($p < 0.001$). Multinomial Naive Bayes, random forest and logistic regression are statistically equivalent ($p < 0.001$). Random forest, gaussian SVM, logistic regression and linear SVM are statistically equivalent ($p < 0.001$). After employing the least significance difference on the Kabita's Kitchen dataset (using count vectorizer), we can say that decision trees, Bernoulli Naive Bayes, polynomial SVM and random forest, multinomial Naive Bayes, decision trees and Bernoulli Naive Bayes are statistically equivalent ($p < 0.001$). In addition, random forest, gaussian SVM, linear SVM and logistic regression are statistically equivalent ($p < 0.001$). On Kabita's dataset (using term frequency vectorizer), result shows that decision trees, polynomial SVM, Bernoulli Naive Bayes, multinomial Naive Bayes and random forest are statistically equivalent ($p < 0.001$). Decision trees, polynomial SVM, multinomial Naive Bayes, random forest and gaussian SVM are statistically equivalent ($p < 0.001$). Multinomial Naive Bayes, random forest, gaussian SVM, logistic regression and linear SVM are statistically equivalent ($p < 0.001$).

5. Discussion

The current section discusses the limitations and findings of our study.

5.1. Limitations

Hinglish is a language used for communication on social media and is not officially supported by the linguistic society; so there are a number of limitations and challenges while dealing with the Hinglish language. There is no in-built list of stopwords for the Hinglish language; therefore, a list of Hinglish stopwords was manually made. Additionally, there are no stemming algorithms for the Hinglish language. Therefore, Porter stemming algorithm was used in our research, which is the standard algorithm for the English text. This affects the accuracy of the machine learning model. There are a couple of threats that might have impacted our research. Firstly, the Youtube API used to extract the information in this research might not have provided us the latest and right datasets. Secondly, machine learning models have been trained on a small number of records (4900) and if on a given day a cooking channel has more than 15,000 comments, then results might be affected based on the size of the data.

5.2. Findings

As part of this research, we addressed multiple research questions based on our assumptions mentioned above. The primary goal was to find the best machine learning algorithm that works in the Hinglish language and, based on our analysis, it was found that the logistic regression worked well for both of our datasets. Many studies have been done using two or three machine learning algorithms [22,23], but in our study nine machine learning algorithms covering both parametric and non-parametric machine learning models were employed. The reason behind using these algorithms was to broaden the scope of our work.

The primary focus of our study was food science in the digital world. Cookery shows in the digital world [39] can be highly influential. Our approach is in contrast with Ketchum's [39] approach; we were more interested in examining the patterns of viewers' views to enhance the essences and capabilities of the cookery channels to the benefit of the Youtubers and the users.

Our study also threw some light on the insight of the cookery channels. Online Channels using an online platform such as Youtube are a great source of sharing knowledge and providing the ease to do business. We were also able to find different patterns in viewers' comments through the dynamic clustering that enabled us to label the data to train our machine learning model. These labels also helped us to capture the perspective of different viewers while viewing the cookery channels on online media.

The trained model that we built during our study can help the Youtuber to predict the right label and enable them to automatically separate the comments, which can ease the analysis and help understand their viewers' requirement. This can help them improve their channel and increase their subscriber base. In order to improve our study and provide all the other cookery channels on Youtube to increase their subscribers, we are planning to embed the training model with the Rest API to make the whole process robust and automatic. Building the Rest API is the future scope of our present study.

6. Conclusions and Future Work

Youtube is the most popular website where large numbers of videos are shared worldwide. People can share their knowledge, ideas and thoughts by putting videos on Youtube. Users watch these videos as means of entertainment or to learn skills or gain knowledge. Here we chose cookery channels, as more Indians are living abroad, and they find the Indian cookery channels very useful in learning cuisines. Therefore, our study was done on Indian cookery channels. Two channels, one is purely based on vegetarian (vegan) Indian cuisines, while the other is based on non-vegetarian as well as vegetarian Indian cuisines, were chosen. This study would help the channel makers to build up their channels by adding those things in the videos which are frequently asked by the users.

Our main objective was to find a promising classifier that can help us to find the sentiments of the comments made on Youtube. Logistic regression with the term frequency vectorizer obtained

74.01% on Nisha Madhulika's dataset. Logistic regression with term frequency vectorizer yielded the best accuracy with 75.37% on Kabita's dataset. By this we could say that Logistic regression classifier worked well with term frequency vectorizer in both our datasets.

For future work, we are planning to apply the deep learning models to these datasets. We will compare the results and find out the better models. After employing the deep learning models, we plan to build the Rest API, which would help Youtubers to automatically separate the viewers' comments and help them to understand the needs of the viewers.

Author Contributions: The following work has been categorized under different authors name. Conceptualization: G.K. and A.K.; Methodology: G.K.; Software part: G.K.; Validation: G.K. and A.K.; Formal analysis: S.S.; Investigation: G.K. and S.S.; Data curation: G.K. and A.K.; writing—original draft preparation: G.K.; Writing—review and editing: G.K., A.K. and S.S.; Visualization: G.K.; Supervision: A.K. and S.S.; Project administration: G.K. and A.K.; Resources: Youtube online Channels; Funding acquisition: NA.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

RF	Random Forest
NB-M	Multinomial Naive Bayes
NB-B	Bernoulli Naive Bayes
NB-G	Gaussian Naive Bayes
LR	Logistic Regression
SVM-L	Linear Support Vector Machine
SVM-P	Polynomial Support Vector Machine
SVM-R	Gaussian Support Vector Machine
CART	Decision Trees
DBSCAN	Density-based spatial clustering of applications with noise
LSD	Least significant difference
POS	Part of speech
BoW	Bag of words
RQ	Research Questions
MaxEnt	Maximum Entropy

References

1. Smith, K. Youtube Statistics. 2019. Available online: <https://www.brandwatch.com/blog/Youtube-stats/> (accessed on 26 March 2019).
2. Mitter, S. Youtube Monthly User Base Touches 225 Million in India, Reaches 80 pc of Internet Population. 2018. Available online: <https://yourstory.com/2018/03/Youtube-monthly-user-base-touches-225-million-india-reaches-80-pc-internet-population> (accessed on 26 March 2019).
3. Maniou, T.A.; Bantimaroudis, P. Hybrid salience: Examining the role of traditional and digital media in the rise of the Greek radical left. *Journalism* **2018**. [CrossRef]
4. Zhao, J.; Gui, X. Comparison research on text pre-processing methods on twitter sentiment analysis. *IEEE Access* **2017**, *5*, 2870–2879.
5. Pang, B.; Lee, L.; Vaithyanathan, S. Thumbs up?: Sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2002; Volume 10, pp. 79–86.
6. Martineau, J.C.; Finin, T. Delta tfidf: An improved feature space for sentiment analysis. In *Proceedings of the Third International AAAI Conference on Weblogs and Social Media*, San Jose, CA, USA, 17–20 May 2009.

7. Pang, B.; Lee, L. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics, Barcelona, Spain, 21–26 July 2004; Association for Computational Linguistics: Stroudsburg, PA, USA, 2004; p. 271.
8. Kaushik, A.; Naithani, S. A Study on Sentiment Analysis: Methods and Tools. *Int. J. Sci. Res. (IJSR)* **2014**, *4*, 287–292.
9. Kaushik, A.; Naithani, S. A comprehensive study of text mining approach. *Int. J. Comput. Sci. Netw. Secur. (IJCSNS)* **2016**, *16*, 69.
10. Liu, B. Sentiment Analysis and Opinion Mining. 2012. Available online: <https://www.cs.uic.edu/~liub/FBS/SentimentAnalysis-and-OpinionMining.pdf> (accessed on 10 April 2019).
11. Xia, R.; Zong, C.; Li, S. Ensemble of feature sets and classification algorithms for sentiment classification. *Inf. Sci.* **2011**, *181*, 1138–1152. [[CrossRef](#)]
12. Neethu, M.; Rajasree, R. Sentiment analysis in twitter using machine learning techniques. In Proceedings of the 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT), Tiruchengode, India, 4–6 July 2013; pp. 1–5.
13. Domingos, P.; Pazzani, M. On the optimality of the simple Bayesian classifier under zero-one loss. *Mach. Learn.* **1997**, *29*, 103–130. [[CrossRef](#)]
14. Gupte, A.; Joshi, S.; Gadgul, P.; Kadam, A.; Gupte, A. Comparative study of classification algorithms used in sentiment analysis. *Int. J. Comput. Sci. Inf. Technol.* **2014**, *5*, 6261–6264.
15. Da Silva, N.F.; Hruschka, E.R.; Hruschka, E.R., Jr. Tweet sentiment analysis with classifier ensembles. *Decis. Support Syst.* **2014**, *66*, 170–179. [[CrossRef](#)]
16. Abdi, A.; Shamsuddin, S.M.; Hasan, S.; Piran, J. Deep learning-based sentiment classification of evaluative text based on Multi-feature fusion. *Inf. Process. Manag.* **2019**, *56*, 1245–1259. [[CrossRef](#)]
17. Arora, M.; Kansal, V. Character level embedding with deep convolutional neural network for text normalization of unstructured data for Twitter sentiment analysis. *Soc. Netw. Anal. Min.* **2019**, *9*, 12. [[CrossRef](#)]
18. Vinodhini, G.; Chandrasekaran, R. Sentiment analysis and opinion mining: A survey. *Int. J.* **2012**, *2*, 282–292.
19. Zhang, L.; Ghosh, R.; Dekhil, M.; Hsu, M.; Liu, B. *Combining Lexicon-Based and Learning-Based Methods for Twitter Sentiment Analysis*; Technical Report HPL-2011; HP Laboratories: Palo Alto, CA, USA, 2011.
20. Bilal, M.; Israr, H.; Shahid, M.; Khan, A. Sentiment classification of Roman-Urdu opinions using Naive Bayesian, Decision Tree and KNN classification techniques. *J. King Saud Univ. Comput. Inf. Sci.* **2016**, *28*, 330–344. [[CrossRef](#)]
21. Uysal, A.K. Feature Selection for Comment Spam Filtering on Youtube. *Data Sci. Appl.* **2018**, *1*, 4–8.
22. Sharma, A.; Nandan, A.; Ralhan, R. An Investigation of Supervised Learning Methods for Authorship Attribution in Short Hinglish Texts using Char and Word N-grams. *arXiv* **2018**, arXiv:1812.10281.
23. Timoney, J.; Davis, B.; Raj, A. Nostalgic Sentiment Analysis of Youtube Comments for Chart Hits of the 20th Century. In Proceedings of the 26th AIAI Irish Conference on Artificial Intelligence and Cognitive Science, Dublin, Ireland, 6–7 December 2018; pp. 386–395.
24. Trinto, N.I.; Ali, M.E. Detecting Multilabel Sentiment and Emotions from Bangla Youtube Comments. In Proceedings of the 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), Sylhet, Bangladesh, 21–22 September 2018; pp. 1–6.
25. Ravi, K.; Ravi, V. Sentiment classification of Hinglish text. In Proceedings of the 2016 3rd International Conference on Recent Advances in Information Technology (RAIT), Dhanbad, India, 3–5 March 2016; pp. 641–645.
26. Kaur, H.; Mangat, V.; Krail, N. Dictionary based Sentiment Analysis of Hinglish text. *Int. J. Adv. Res. Comput. Sci.* **2017**, *8*, 816–823.
27. Khan, F.H.; Qamar, U.; Bashir, S. A semi-supervised approach to sentiment analysis using revised sentiment strength based on SentiWordNet. *Knowl. Inf. Syst.* **2017**, *51*, 851–872. [[CrossRef](#)]
28. Silva, N.F.F.D.; Coletta, L.F.; Hruschka, E.R. A survey and comparative study of tweet sentiment analysis via semi-supervised learning. *Acm Comput. Surv. (CSUR)* **2016**, *49*, 15. [[CrossRef](#)]
29. Benkhelifa, R.; Laallam, F.Z. Opinion extraction and classification of real-time Youtube cooking recipes comments. In *International Conference on Advanced Machine Learning Technologies and Applications*; Springer: Cairo, Egypt, 2018; pp. 395–404.

30. Bianchini, D.; De Antonellis, V.; De Franceschi, N.; Melchiori, M. PREFer: A prescription-based food recommender system. *Comput. Stand. Interfaces* **2017**, *54*, 64–75. [CrossRef]
31. Pugsee, P.; Niyomvanich, M. Comment analysis for food recipe preferences. In Proceedings of the 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Hua Hin, Thailand, 24–27 June 2015; pp. 1–4.
32. Yu, N.; Zhekova, D.; Liu, C.; Kübler, S. Do good recipes need butter? Predicting user ratings of online recipes. In Proceedings of the IJCAI Workshop on Cooking with Computers, Beijing, China, 3–9 August 2013.
33. Kaushik, A.; Kaur, G. Youtube Cookery Channels Viewers Comments in Hinglish. 2019. Available online: <https://zenodo.org/record/2841848> (accessed on 3 July 2019).
34. Maudhika, N. Youtube Nisha's Madhulika Cookery Channel. 2019. Available online: <https://www.Youtube.com/user/NishaMadhulika> (accessed on 6 March 2019).
35. Kabita. Youtube Kabita's Kitchen Cookery Channel. 2019. Available online: <https://www.Youtube.com/channel/UCCChqsCRFePrP2X897iQkyAA> (accessed on 15 March 2019).
36. Python. Python Link. 2019. Available online: <https://www.python.org/> (accessed on 2 March 2019).
37. Scikit. Scikit Version Link. 2019. Available online: <https://scikit-learn.org/stable/> (accessed on 7 March 2019).
38. Alberto, T.C.; Lochter, J.V.; Almeida, T.A. TubeSpam: Comment Spam Filtering on Youtube. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; pp. 138–143. [CrossRef]
39. Ketchum, C. The essence of cooking shows: How the food network constructs consumer fantasies. *J. Commun. Inq.* **2005**, *29*, 217–234. [CrossRef]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).