Analysis of sentiment analysis in tweets using machine learning algorithms

S.Kokila[1], K.Shiyamala[2]

Tagore Institute of Engineering and Technology, Salem, Tamil Nadu, India.

Abstract:

In modern society, the use of social networks is more than ever and they have become the most popular medium for daily communications. Twitter is a social network where users are able to share their daily emotions and opinions with tweets. Sentiment analysis is a method to identify these emotions and determine whether a text is positive, negative, or neutral. In this article, we apply four widely used data mining classifiers, namely K-nearest neighbor, decision tree, support vector machine, and naive Bayes, to analyze the sentiment of the tweets. The analysis is performed on two datasets: first, a datasetwith two classes (positive and negative) and then a three-class dataset (positive, negative and neutral). Furthermore, we utilizetwo ensemble methods to decrease variance and bias of the learning algorithms and subsequently increase the reliability. Also, we have divided the dataset into two parts: training set and testing set with different percentages of data to show the best train–test split ratio. Our results show that support vector machine demonstrates better outcomes compared to other algorithms, showing an improvement of 3.53% on dataset with two-class data and 7.41% on dataset with three-class data inaccuracy rate compared to other algorithms. The experiments show that the accuracy of single classifiers slightly outperforms that of ensemble methods; however, they propose more reliable learning models. Results also demonstrate that using 50% ofthe dataset as training data has almost the same results as 70%, while using tenfold cross-validation can reach better results.

Keywords: Social networks analysis, Sentiment analysis, Data mining, Text mining

Introduction

Social networks (SNs) are becoming increasingly popular platforms among people all across the world, and nowadays they are utilized even more than ever. With the growth of SNs like Twitter and increasing their popularity, people share more personal emotions and opinions about various issues in such networks. This rapid growth of SNs, combined with the accessibility of a large amount of data on a multitude of topics, provides a great research potential for a wide range of applications, such as customer analysis, product analysis, sector analysis and digital marketing (Bhatnagar and Choubey 2021; Fatehi, et al. 2022). In addition, identifying users' polarities and mining their opinions shared in various areas, especially SNs, have become one of the most popular and useful research fields. Social media platforms are able to build rich profiles from the online presence of users by tracking activities such as participation, messaging, and Web site visits (Cui, et al. 2020). By an increased growth in the number of users in the SNs and subsequently exponential rise in the interactions between them, large volumes of user-generated content are produced. It is difficult to analyze all these data since most of the social media data are unstructured and dynamic data which frequently alters. Social network analysis provides innovative techniques to analyze interactions among entities by emphasizing on social relationships (Kumar and Sinha 2021). Nowadays, analyzing SNs with data mining and machine learning algorithms has become a must-have strategy for obtaining useful data. Data mining is the process of extracting and identifyinguseful patterns and relationships from piles of data sets that may lead to the extraction of new information by using data analysis tools (Keyvanpour, et al. 2020).

Among different SNs, twitter is one of the most studied SNs for social networks' research. Twitter is a SN that enables users to share their daily emotions and opinions. It isconsidered a convenient platform for users to share personalmessages, pictures, and videos. One of the main advantagesof platforms like twitter is that users are organized in these platforms, making this possible to investigate groups of people or communities who are united by common interests, rather than individual profiles. Furthermore, this is possible through extensive use of hashtags, mentions, and retweets that form a complex network, which can provide us witha rich source of data to analysis. Twitter is known to be a novel source of data for those studying attitudes, beliefs, and behaviors of consumers and opinion makers (Islam, et al. 2020; Kwak and Grable 2021).

Among all various forms of communications, text messages are considered one of the most conspicuous forms, since users can express their opinions and emotions on various and diverse topics using text. Text mining is the process of exploring and transforming unstructured text data into structured data to find meaningful insights. It is defined as a multi-purpose research method to study a wide range of issues by systematically and objectively identifying characteristics of large sample data. Text mining is a sub-field of data mining and an extension of classical data mining methods, which can be applied when making sophisticated formulations using text classification and clustering procedures(Yang, et al. 2021). Hossny, et al. 2020 listed the key challenges for analyzing the text on Twitter including the tweet's length, frequent use of abbreviations, misspelled words and acronyms, transliterating non-English words using Roman scripts, ambiguous semantics and synonyms.

Information in several social media platforms, like blogs, reviewing SNs, and Twitter, is being processed for extracting people's opinions about a particular product, organization, or situation. The attitude and feelings comprise an essential part in evaluating the behavior of an individual that is known as sentiments. These sentiments can further be analyzed using a field of study, known as sentiment analysis (SA) (Singh, et al. 2021). SA belongs to the area of natural language processing (NLP) (Chen, et al. 2020) and it has been an active research topic in NLP, which is a cognitive computing study of people's opinions, sentiments, emotions, appraisals, and attitudes toward entities such as products, services, organizations, individuals, issues, events, topics, and their attributes (Dai, et al. 2021). Also, it aims to analyze and extract knowledge from the subjective information published on the Internet (Basiri, et al. 2021). Sentiment analysis of user-generated data is very useful to know the opinionof the crowd. Two main approaches for sentiment analysis of text documents are described in the literature, specifically approaches based on machine learning and approaches basedon symbolic techniques. Symbolic techniques use lexicons and other linguistic resources to determine the sentimentof a given text. Some research has used machine learning for classifying the sentiment of a given text, sometimes following the approach of most symbolic techniques and seeking to identify positive, negative and neutral categories, butsometimes also considering other sentiment categories suchas anger, joy and sadness (Moutidis and Williams 2020).

The SA plays significant role in many domain by extracting the people's emotions which then assist business industry to be developed accordingly. In this study, we investigatethe performance of different ML models to

analyze the sentiment of two real datasets.
So, the contributions in this paper are summarized as follows:

(1) We generate and preprocess two real datasets extracted with Twitter Application Programming Interface (API)—binomial and polynomial—to investigate the sentiment analysis. Binomial dataset incorporating two polarities of positive and negative which is the typical dataset used in the literature, polynomial dataset, however, includes three positive, negative, and neutral polarities.
(2) We investigate the performance of sentiment classification in terms of accuracy /AUC and accuracy/kappa for four classifiers on both binomial and polynomial datasets, respectively.
(3) To increase the reliability of SA and reduce variance and bias of learning models, we apply ensemble methods on both the binomial and polynomial datasets and then report the accuracy values for these methods.
(4) To find out the best train–test split ratio in addition to K-fold cross-validation, we divide the dataset into two parts: training set and testing set with different percent-ages of data.

The rest of this paper is structured as follows: Sect. 2 reviews some of the related works in the literature. A description of the methodology that includes data collection, preprocessing for sentiment analysis, sentiment detection, and classification modelling is presented in Sect. 3. The results are presented and discussed in Sect. 4, and eventually, the conclusion is detailed in Sect. 5.

1 Related work

Researchers in the field of sentiment analysis have been mostly used supervised machine learning algorithm for primary classification, such as the work done by Chauhan et al. (2020). Furthermore, many of the recent studies use Twitter as the primary source of data (Al-Laith, et al. (2021), Yadav, et al. (2021)).
Henríquez and Ruz (2018) used a non-iterative deep random vectorial functional link called D-RVFL. They analyzed two different datasets. Dataset 1 contains a collection of 10,000 tweets from the Catalan referendum of 2017 and dataset 2 contains a collection of 2187 tweets from the Chilean earthquake of 2010. They consider the datasets as a two-class classification problem with the labels of positive and negative. By the use of D-RVFL, results show the best performance compared to SVM, random forest, and RVFL. Ankit and Saleena (2018) proposed an ensemble classification system formed by different learners, such as naive Bayes, random forest, SVMs, and logistic regression classifiers. Their system employs two algorithms: the first algo-rithm calculates a positive and a negative score for the tweet, and the second algorithm utilizes these scores to predict the sentiment of that tweet. Furthermore, the dataset consists of
43,532 negative and 56,457 positive tweets.
Symeonidis et al. (2018) evaluated the preprocessing techniques on their resulting classification accuracy and the number of features they produce. However, this paper worked on lemmatization, removing numbers, and replacing contractions techniques, while the detection accuracy is low. For this task, they used four classification algorithms named logistic regression, Bernoulli Naive Bayes, linear SVC, and convolutional neural networks on two datasets with the classes of positive, negative, and neutral.
Sailunaz and Alhajj (2019), used a dataset to detect sentiment and emotion from tweets and their replies and measured the influence scores of users based on various user- based and tweet-based parameters. The dataset also includes replies to tweets, and the paper introduces agreement score, sentiment score and emotion score

of replies in influence score calculation.

Ruz, et al. (2020), reviewed five classifiers and assessed their performances on two Twitter datasets of two different critical events. Their datasets were Spanish, and they concluded that there is no difference between the behavior of support vector machine (SVM) and random forest in English and Spanish. In order to automatically control the number of edges supported by the training examples in the Bayesian network classifier, they adopt a Bayes factor approach, yielding more realistic networks.

Wang et al. (2021) proposed a system for general population sentiment monitoring from a social media stream (Twitter), through comprehensive multilevel filters, and improved latent Dirichlet allocation (LDA) method for sentiment classification. They reached an accuracy of 68% for general sentiment analysis using real-world content. Also, they used a dataset with three categories (positive, negative, and neutral) and a dataset with four categories (positive, negative, neutral and junk).

Ali et al. (2021) utilized the bilingual (English and Urdu) data from Twitter and NEWS websites to do the sentiment and emotional classification using machine learning and deep learning models. Kaur and Sharma (2020) used API to collect beneficial-related corona virus tweets and then cate- gorized them in three groups (positive, negative, and neutral) to investigate the feeling of people about the COVID-19 pan- demic. Nuser et al. (2022) proposed an unsupervised learn- ing framework based on serial ensemble of some hierarchi- cal clustering methods for sentiment analysis on a binomial dataset collected from Twitter.

Machuca et al. (2021) used English COVID-19 pandemic tweets to do the sentiment analysis using a logistic regres- sion algorithm on a binomial dataset including positive and negative labels.

In Table 1, we present a review of the state-of-the-art and their reported accuracy for the sentiment classification with data structures of binomial (positive and negative) and polynomial (positive, negative, and neutral).

2 Methodology

This section introduces our research framework in four phases: data collection, preprocessing, sentiment detection, and classification modeling (Fig. 1).

2.1 Data collection

Twitter is among the most popular social networking plat- forms nowadays. It provides its users with a platform to share their daily lives with other users and express their opinions about different national, international issues from various perspectives. Every user can write a short text called tweet with a maximum length of 140 characters. These opin- ions and comments can be used to raise public awareness to help the government and enterprises understand the views of the public. Twitter also can be used to predict event trends. Therefore, tweets are an important resource to study public awareness.

Table 1 Comparison of sentiment analisys approches

Paper	Dataset structure	Reported accuracy (%)
Henríquez and Ruz (2018)	Binomial	82.90

Ankit and Saleena (2018)	Binomial	75.81
Symeonidis, et al. (2018)	Polynomial	67.30
Sailunaz, et al. (2019)	Polynomial	66.86
Ruz, et al. (2020)	Binomial	81.20
Wang, et al. (2021)	Polynomial	68.00
Al-Laith, et al (2021)	Polynomial	69.40
Nuser, et al. (2022)	Binomial	73.75
Ali, et al. (2021)	Polynomial	80.00
Machuca, et al. (2021)	Binomial	78.50

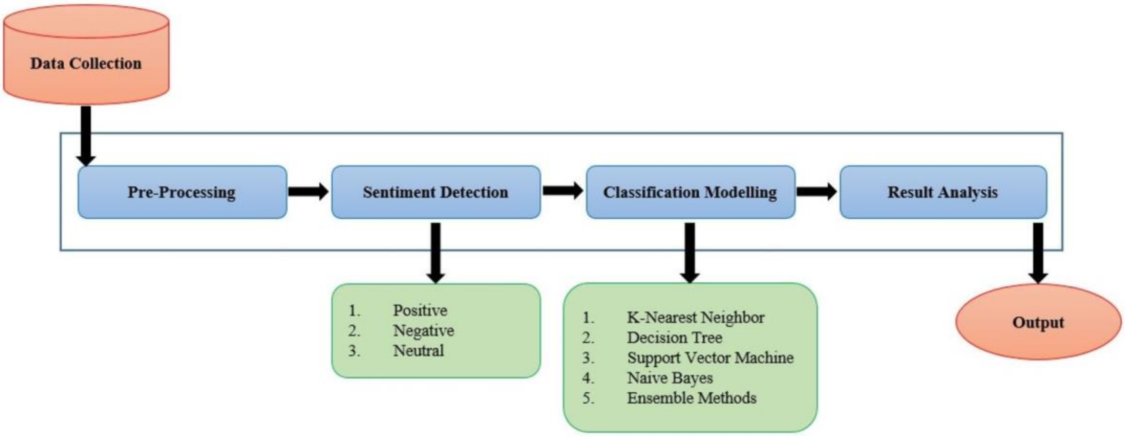

Fig. 1 Overview of proposed sentiment classification workflow

Researchers and practitioners can access Twitter data using Twitter API. Search and streaming APIs allow them to collect Twitter data using different types of queries, includ- ing keywords and user profiles, which has offered them opportunities to access the data needed to analyze challeng- ing problems in diverse domains. Thus, many researchers and practitioners have begun to focus on Twitter data mining to obtain more research value and business value from this research (Li et al. 2019).

For our experiments, in order to collect tweets, we selected a few recent events and issues; search keywords about corona virus like #covid-19, #coronavirus. For our experiments, in order to collect tweets, we selected a few recent events and issues; search keywords about corona virus like #covid-19, #coronavirus, #covid19vaccine, etc. A total of 14,000 tweets were extracted using Twitter API. 6980 of which were written in English; therefore, we picked these tweets. These tweets were sentences; consequently, we had to preprocess these sentences and convert them to a set of words. Then, these words were classified to be understood by the classifier. In the following sections, we elaborate the mentioned procedure.

2.2 Preprocessing

Tweets are sometimes not in a usable format, for instances they include characters, symbols or emoticons. Therefore, we need to format them in an appropriate usable form to be able to extract meaningful opinions from them. As a first step in preprocessing, most (if not all) studies apply tokeni- zation. Tokenization is a task for separating the full text string into a list of separate words. Tokenization is defined as a kind of lexical analysis that breaks a stream of text up into words, phrases, symbols, or other meaningful elements called tokens. At its core, the process of tokenization is a standard method for further natural language processing (NLP) transformation in preprocessing (Symeonidis, et al. 2018). For the preprocessing steps, various methods have been proposed and can be applied for data cleaning. Fol- lowing are the steps in the data preprocessing that we used in this article:

- All non-English tweets are eliminated.
- User names preceded by '@' and external links are omit-ted.
- All hashtags (only the # symbol) are removed.
- Stop-words or useless words are removed from the tweet.
- All emoticons were removed (i.e.,:-),:-(etc.).

All the tweets were converted to lower case to make thedataset uniform.

2.3 Detection

Each tweet should be labeled with sentiment with three pos-sible values: negative, neutral, or positive. The first step to label the tweets is to apply unsupervised methods due to the large dataset we have. For this purpose, we used the Text- Blob library in the python programming language to label tweets. This library assigns each tweet a number between

– 1 and + 1 (-1 is the most negative and + 1 is the most posi-tive value). Then, we double-checked the labels manually. Tweets between [− 1, − 0.1], [− 0.1, + 0.1] and [+ 0.1, + 1] were labeled negative, neutral, and positive, respectively. Figure 2 illustrates the results from the sentiment analysis. Also, the number of tweets in each class is shown in Table 2. We have a total of 6980 tweets: 977 of which are negative, 3689 of which are neutral and positive tweets are 2314.

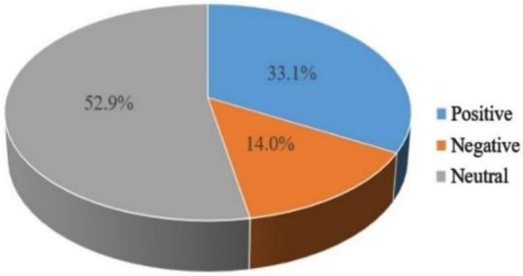

Fig. 2 Sentiment proportion of dataset

Table 2 Dataset structure

	Number of tweets in dataset
Positive	2314
Neutral	3689
Negative	977
Total	6980

2.4 Classification modelling

For our experiment and in order to make a comparative analysis, we employed four classifiers, which are the most widely used classifiers for sentiment analysis, namely (1) K-nearest neighbor (KNN), (2) decision tree (DT), (3) sup- port vector machine (SVM), (4) Naive Bayes (NB), and also two ensemble methods including voting and bagging.

2.4.1 K-nearest neighbor

The logic behind KNN classification is that we expect a test sample X to have the same label as the training sample located in the local region surrounding X denoting by K. Training a KNN classifier simply consists of determining K. KNN simply memorizes all samples in the training set and then compares the test sample with them.

2.4.2 Decision tree

The decision tree is a particularly efficient method of pro- ducing classifiers from data. It is a tree-like collection of nodes intended to create a decision on values affiliation to aclass or an estimate of a numerical target value. Each node represents a splitting rule for one specific attribute. For clas- sification, this rule separates values belonging to different classes. The building of new nodes is repeated until the stop- ping criteria are met. A prediction for the class label attrib- ute is determined depending on the majority of examples which reached this leaf during generation.

2.4.3 Support vector machine

An SVM is a supervised learning algorithm creating learn- ing functions from a set of labeled training data. Support vector machine solves the traditional text categorization problem effectively. The main principle of SVMs is to deter- mine a linear separator that separates different classes in thesearch space with a maximum distance. SVM’s classifica- tion function is based on the concept of decision planes that define decision boundaries between classes of samples. Themain idea is that the decision boundary should be as far away as possible from the data points of both classes. Thereis only one that maximizes the margin.

2.4.4 Naive Bayes

The naive Bayesian method is one of the most widely used methods for text data classification. The naive Bayesian is a simple probabilistic classifier that uses the concept of mix- ture models to perform classification. The mixture model relies on the assumption that each of the predefined classes is one of the components of the mixture itself. The componentsof the mixture model denote the probability of belonging-ness of any term to the particular component. Naive Bayes classifier uses the concept of Bayes theorem and finds the maximum prospect of the probability of any word fitting to a particular given or predefined class. This algorithm assumes that the elements in the dataset are independent from each other and their occurrences in different datasets indicate their relevance to certain data attributes (Desai and Mehta 2016).

This method is a high-bias, low-variance classifier, and it can build a good model even with a small data set. Typical use cases involve text categorization, including spam detection, sentiment analysis, and recommender systems.

2.4.5 Ensemble methods

Ensemble methods are learning algorithms which by try to improve the predicted performance by employing a set of learning algorithms. They reduce bias and variance of the model and so are more reliable compared to the single clas- sifier (Dietterich 2000). The voting method can be used with different combination sets of the classifiers; therefore, we applied the voting method with the combination set of all classifiers to get the maximum value for accuracy. We also used the bagging method with DT (generally this amalga- mation has shown a better performance) and bagging with SVM, KNN, and NB.

2.4.6 Evaluation metric

2.4.6.1 Accuracy

$$\text{accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}$$

TP, TN, FP, and FN are the number of true positive, true negative, false positive, and false negative.

2.4.6.2 AUC The area under the curve (AUC) is the meas- ure of the ability of a classifier to distinguish between classes and is used as a summary of the receiver operator characteristic (ROC) curve. The higher the AUC, the better the performance of the model at distinguishing between the positive and negative classes.

2.4.6.3 Kappa Kappa is a metric that provides a compari- son between observed accuracy and expected accuracy.

To start the classification, we divided the dataset into a training set and a testing set with different percentages of data. Common ratios used are 70% or 60% of the dataset for training and 30% or 40% for testing. In our experiment, we used different train–test split percentage, which are 10% to 70%. Continuing the classification, we also used K-fold cross-validation (K-FCV) with $K = 10$ to generate the train- ing set and the testing set and compare the results with above-mentioned split ratios.

In this paper, first, the above-mentioned classifiers were applied to a dataset with just negative and positive tweets (binomial), and then, the classifiers were applied to a dataset including negative, positive, and neutral tweets (polynomial).

3 Result analysis

This section gives an overview of the accuracy rates of dif- ferent trained classifiers. All the calculations are done in the RapidMiner Studio application.

Table 3 shows the predicted accuracy of all classifiers when the tweets are binomial. Our results in Table 3 demon- strate that K-FCV with $k = 10$ has the highest accuracy rate, except DT, besides the accuracy when we use the train–test split procedure. SVM with 86.42% in single methods and voting with 86.75% in ensemble methods has the best accu-racy rates. In Table 4, we can see the differences between the accuracy rates. In most algorithms, there is some decrease in accuracy rate when we used 60% of the dataset for training data. Also, this decrease can be seen when 40% of the dataset is used for training in some methods. Furthermore, in all methods when the ratio is 20%, there is the most increase in accuracy rate in comparison with the ratio of 10%. NB

Table 3 Sentiment accuracy comparison on binomial dataset

Algorithm	Train–test split percentage							10-FCV
	20%	30%	40%	50%	60%	70%	10%	
KNN	73.26	76.45	78.86	78.98	82.13	81.47	82.37	82.89
DT	74.34	75.96	76.65	76.34	77.26	76.23	77.61	76.39
SVM	76.47	78.58	80.90	83.28	83.65	84.13	85.21	86.42
NB	71.54	76.79	77.30	79.03	80.67	80.49	81.05	81.43
Voting	76.23	80.14	82.16	83.89	85.35	85.35	85.71	86.75
Bagging (KNN)	73.63	76.95	78.82	79.38	81.95	82.08	82.57	82.86
Bagging (DT)	75.15	76.34	77.34	76.75	78.05	76.84	78.32	76.85
Bagging (SVM)	76.33	78.05	80.86	82.57	83.53	83.90	85.31	86.08
Bagging (NB)	71.64	76.98	77.21	79.28	80.67	80.64	81.26	81.46

Table 4 Sentiment accuracy

Algorithm	Variation between train–test split percentages differences on binomial dataset					
	10–20%	20–30%	30–40%	40–50%	50–60%	60–70%
KNN	+ 3.19	+ 2.41	+ 0.12	+ 3.15	– 0.66	+ 0.90
DT	+ 1.62	+ 0.69	– 0.31	+ 0.92	– 1.03	+ 1.38
SVM	+ 2.11	+ 2.32	+ 2.38	+ 0.37	+ 0.48	+ 1.08
NB	+ 5.25	+ 0.51	+ 1.73	+ 1.64	-0.18	+ 0.56
Voting	+ 3.19	+ 2.02	+ 1.73	+ 1.46	0.00	+ 0.36
Bagging (KNN)	+ 3.32	+ 1.87	+ 0.56	+ 2.57	+ 0.13	+ 0.49
Bagging (DT)	+ 1.19	+ 1.00	– 0.59	+ 1.30	– 1.21	+ 1.48
Bagging (SVM)	+ 1.72	+ 2.81	+ 1.71	+ 0.96	+ 0.37	+ 1.41
Bagging (NB)	+ 5.34	+ 0.23	+ 2.07	+ 1.39	– 0.03	+ 0.62

algorithm with + 9.15% and bagging with NB with + 9.62% have the most variation in accuracy rate from 10 to 70% train–test split percentages of the dataset.

Table 5 shows the predicted AUC for binomial dataset. SVM and bagging with SVM have the best values. We can also see that the 10-FCV has better results than the split procedure. From Table 6, the results show that there is some reduction when we use 60% of the dataset for training data than 50%. An increase in AUC from 10 to 20% of the dataset is more than other ratios. The classification continued with the polynomial dataset. So we applied classifiers to the dataset with three categories including positive, negative, and neutral tweets. Tables 7, 8, 9, 10 show the comparison between classifiers in terms of accuracy and kappa metrics when the tweets are polynomial. According to Tables 7, 8, 9, 10, there is some reduction in accuracy and kappa rates when we use 60% of the dataset for training data than 50% in most classifiers, and in some cases we have just a little increase in the accuracy and kappa rates. SVM and bagging with SVM have better results com-pared to other classifiers. SVM with an accuracy of 73.91%

Table 5 Sentiment AUC comparison on binomial dataset

Algorithm	Train–test split percentage							10-FCV
	10%	20%	30%	40%	50%	60%	70%	
KNN	0.749	0.800	0.828	0.845	0.863	0.871	0.868	0.876
DT	0.579	0.579	0.610	0.604	0.619	0.601	0.625	0.604
SVM	0.793	0.847	0.878	0.897	0.917	0.913	0.929	0.932
NB	0.495	0.550	0.556	0.608	0.643	0.637	0.655	0.601
Voting	0.598	0.670	0.704	0.731	0.779	0.745	0.761	0.794
Bagging (KNN)	0.741	0.792	0.825	0.839	0.861	0.865	0.861	0.877
Bagging (DT)	0.618	0.637	0.641	0.624	0.652	0.651	0.647	0.638
Bagging (SVM)	0.795	0.849	0.879	0.898	0.918	0.917	0.929	0.934
Bagging (NB)	0.706	0.753	0.768	0.787	0.821	0.813	0.817	0.824

Table 6 Sentiment AUC differences on binomial dataset

Algorithm	Variation Between Train–Test Split Percentages					
	10–20%	20–30%	30–40%	40–50%	50–60%	60–70%
KNN	+ 0.051	+ 0.028	+ 0.017	+ 0.180	+ 0.008	-0.003
DT	0.000	+ 0.031	– 0.006	+ 0.015	– 0.018	+ 0.024
SVM	+ 0.054	+ 0.031	+ 0.019	+ 0.020	– 0.004	+ 0.016
NB	+ 0.055	+ 0.006	+ 0.052	+ 0.035	– 0.006	+ 0.018
Voting	+ 0.072	+ 0.034	+ 0.027	+ 0.048	– 0.034	+ 0.016
Bagging (KNN)	+ 0.051	+ 0.033	+ 0.014	+ 0.022	+ 0.004	– 0.004
Bagging (DT)	+ 0.019	+ 0.004	– 0.017	+ 0.028	– 0.001	– 0.004
Bagging (SVM)	+ 0.054	+ 0.030	+ 0.019	+ 0.020	– 0.001	+ 0.012
Bagging (NB)	+ 0.047	+ 0.015	+ 0.019	+ 0.034	– 0.008	+ 0.004

Table 7 Sentiment accuracy comparison on polynomial dataset

Algorithm	Train–test split percentage							10-FCV
	10%	20%	30%	40%	50%	60%	70%	
KNN	57.02	59.55	61.46	62.79	63.72	64.09	64.95	66.50
DT	54.08	54.94	54.89	54.72	55.47	55.25	55.73	55.49
SVM	61.73	65.29	67.56	69.09	70.69	71.00	71.97	73.91
NB	54.14	57.27	57.90	58.90	60.69	60.08	60.89	61.69
Voting	58.48	61.19	62.98	64.60	65.93	65.81	66.76	68.30
Bagging (KNN)	56.37	59.06	60.52	62.17	63.32	63.62	64.37	66.54
Bagging (DT)	54.36	54.96	54.56	55.22	55.47	55.25	55.87	55.56
Bagging (SVM)	61.72	65.28	67.56	68.98	70.72	70.96	71.97	73.87
Bagging (NB)	54.17	57.18	58.00	58.97	60.74	60.08	61.03	61.92

Table 8 Sentiment accuracy

Algorithm Variation between train–test split percentages
differences on polynomial
dataset

Table 9 Sentiment Kappa

	10–20%	20–30%	30–40%	40–50%	50–60%	60–70%
KNN	+ 2.53	+ 1.91	+ 1.33	+ 0.93	+ 0.37	+ 0.86
DT	+ 0.86	– 0.05	– 0.17	+ 0.75	– 0.22	+ 0.48
SVM	+ 3.56	+ 2.27	+ 1.53	+ 1.60	+ 0.31	+ 0.97
NB	+ 3.13	+ 0.63	+ 1.00	+ 1.79	– 0.61	+ 0.81
Voting	+ 2.71	+ 1.79	+ 1.62	+ 1.33	– 0.12	+ 0.95
Bagging (KNN)	+ 2.69	+ 1.46	+ 1.65	+ 1.15	+ 0.30	+ 0.75
Bagging (DT)	+ 0.60	– 0.40	+ 0.66	+ 0.25	– 0.22	+ 0.62
Bagging (SVM)	+ 3.56	+ 2.28	+ 1.42	+ 1.74	+ 0.24	+ 1.01
Bagging (NB)	+ 3.01	+ 0.82	+ 0.97	+ 1.77	-0.66	+ 0.95

Algorithm Train–test split percentage 10-FCV
comparison on polynomial
dataset

Table 10 Sentiment Kappa

		10%	20%	30%	40%	50%	60%	70%
KNN	0.108	0.173	0.221	0.253	0.275	0.284	0.306	0.341
DT	0.042	0.064	0.063	0.058	0.077	0.070	0.083	0.077
SVM	0.225	0.310	0.363	0.398	0.433	0.441	0.463	0.504
NB	0.247	0.298	0.315	0.335	0.369	0.362	0.377	0.399
Voting	0.150	0.218	0.261	0.300	0.330	0.328	0.351	0.384
Bagging (KNN)	0.090	0.160	0.196	0.237	0.265	0.272	0.292	0.343
Bagging (DT)	0.051	0.066	0.053	0.073	0.077	0.070	0.087	0.079
Bagging (SVM)	0.225	0.310	0.363	0.396	0.434	0.440	0.463	0.503
Bagging (NB)	0.247	0.296	0.316	0.336	0.370	0.362	0.379	0.398

Algorithm differences on polynomial dataset	Variation between train–test split percentages					
	10–20%	20–30%	30–40%	40–50%	50–60%	60–70%
KNN	+ 0.065	+ 0.048	+ 0.032	+ 0.022	+ 0.009	+ 0.022
DT	+ 0.022	– 0.001	– 0.005	+ 0.019	– 0.007	+ 0.013
SVM	+ 0.055	+ 0.053	+ 0.035	+ 0.035	+ 0.008	+ 0.022
NB	+ 0.051	+ 0.017	+ 0.020	+ 0.034	– 0.007	+ 0.015
Voting	+ 0.068	+ 0.043	+ 0.039	+ 0.030	– 0.002	+ 0.023
Bagging (KNN)	+ 0.070	+ 0.063	+ 0.041	+ 0.028	+ 0.007	+ 0.020
Bagging (DT)	+ 0.015	-0.013	+ 0.020	+ 0.004	-0.007	+ 0.017
Bagging (SVM)	+ 0.085	+ 0.053	+ 0.033	+ 0.038	+ 0.006	+ 0.023
Bagging (NB)	+ 0.049	+ 0.020	+ 0.020	+ 0.034	– 0.008	+ 0.017

is the better choice for polynomial classification. However, the bagging with SVM is a more reliable model compared to SVM, employing the ensemble method. This technique makes the learning model more reliable by reducing vari- ance and bias. Tables 7 and 10 show that the most positive variation has happened from 10 to 20% of the dataset in bothaccuracy and kappa terms.

From the results of accuracy and AUC on the binomial dataset (Tables 3, 4, 5, 6) and the results of accuracy and kappa on the polynomial dataset (Tables 7, 8, 9, 10), we can observe that SVM and bagging with SVM have better results compared to other classifiers. However, the accuracy of pol- ynomial classification is less than binomial. The reason of over-performing of SVM can be the fact the text data have asparse nature. In such type of data, there are few irrelevant features that tend to have a correlation with each other. This leads those features to turn into some distinct categories, which can be separated by linear separators. Also, we can see most of the classifiers in 50% train–test split percentage have almost the same results as 70% in accuracy (Figs. 3 and 4), AUC and kappa rates, while using 10-FCV can reachbetter results.

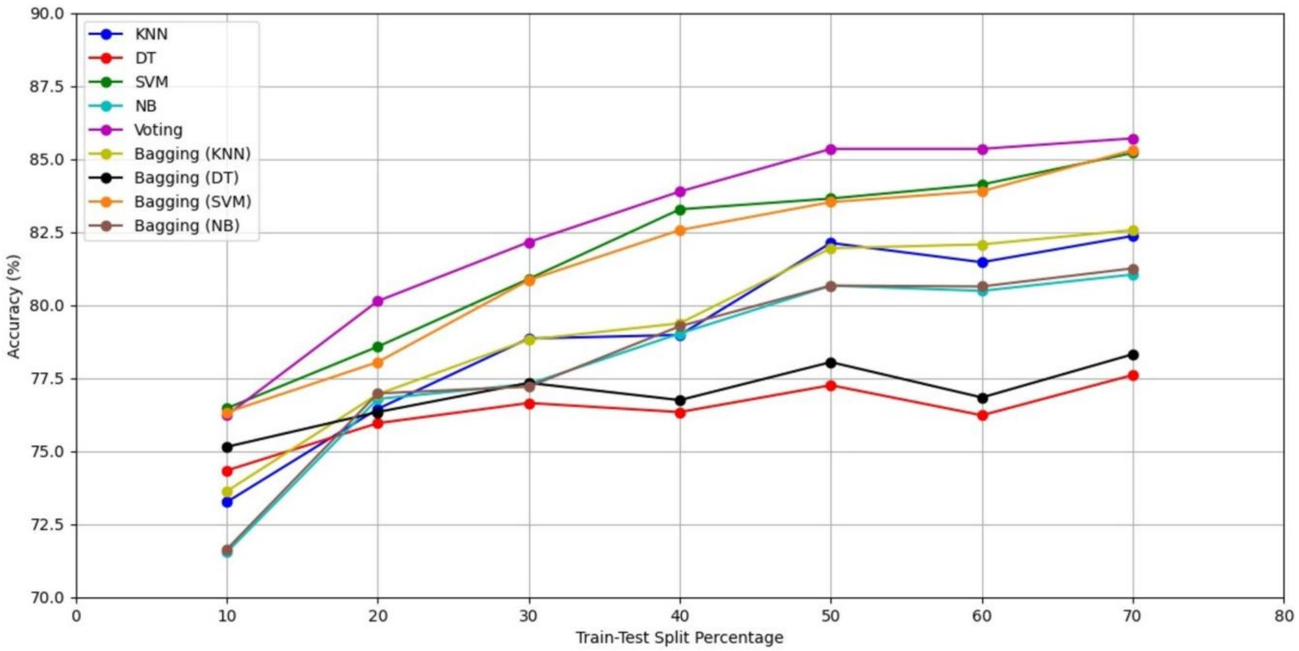

Fig. 3 Classification accuracy on binomial dataset

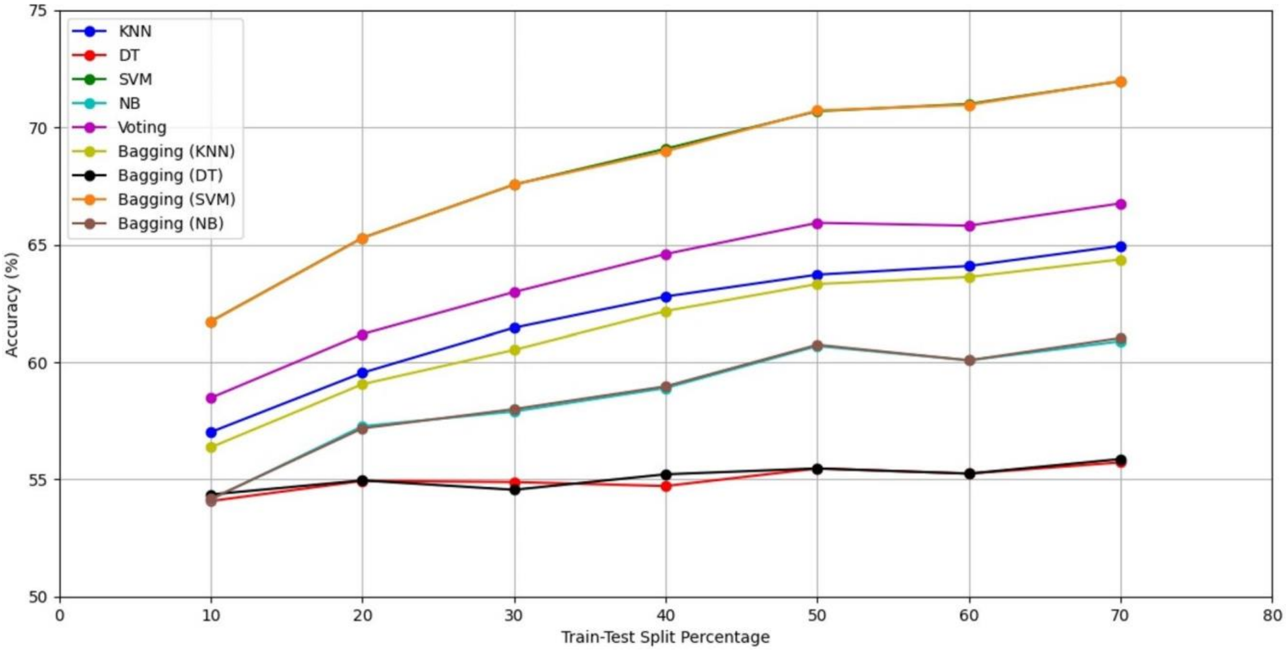

Fig. 4 Classification accuracy on polynomial dataset

We also compared the performance of SVM, when 10-FCV is imposed, with state of the art presented in Table 1. The results showed that overall accuracy has improved at least 3.52% and 5.91% on binomial and polynomial datasets, respectively. This improvement can be a result of using the training and testing data divided throughthe K-fold cross-validation method.

4 Conclusion

In this paper, we aimed to analyze the sentiment of social media data, specifically Twitter, using both single classifiers and ensemble models combined with single classifiers on two datasets including binomial (positive and negative) andpolynomial (positive, negative, and neutral) datasets.

From the results, we observed that data mining is a good choice for sentiment prediction since the accuracy rates are relatively high values. We also reviewed four classifiers, including SVM, K-nearest neighbor, decision tree and naiveBayes and two bagging ensemble methods.

From the results, we concluded that among single classifi- ers and their combination with the ensemble methods, SVM reached 3.53% and 7.41% over performances on binomial and polynomial datasets, respectively. Although ensemble methods do not show over performance compared to singlemethods, they are able to decrease the bias or variance of the learning models and also decrease the generalization error. Therefore, there is a trade-off between reliability of the algo-rithm and accuracy.

Our results show that using 50% of the dataset as train- ing data has almost the same results as 70%; however, using10-FCV has better results. This conclusion can be seen bothin the accuracy and AUC rates in the binomial dataset and accuracy and kappa rates in the polynomial dataset.

In future studies, we will apply other ensemble meth- ods, such as boosting and stacking combined with other classifiers, along with single classifiers. Furthermore, we will attempt to improve our dataset by selecting other keywords including both negative and positive sentiments and increasing the size of the dataset by extracting more tweets.

References

1. Ali MZ, Javed K, Tariq A (2021) Sentiment and emotion classifica- tion of epidemic related bilingual data from social media. arXiv preprint arXiv:2105.01468
2. Al-Laith A, Shahbaz M, Alaskar HF, Rehmat A (2021) Arasencorpus: a semi-supervised approach for sentiment annotation of a large arabic text corpus. Appl Sci 11(5):2434
3. Ankit, Saleena N (2018) An ensemble classification system for Twitter sentiment analysis. Procedia Comput Sci 132:937–946. https:// doi.org/10.1016/j.procs.2018.05.109
4. Basiri ME, Nemati S, Abdar M, Cambria E, Rajendra AU, (2021) ABCDM: an attention-based bidirectional CNN-RNN deep model for sentiment analysis. Futur Gener Comput Syst 115:279–294. https://doi.org/10.1016/j.future.2020.08.005
5. Bhatnagar S, Choubey N (2021) Making sense of tweets using senti- ment analysis on closely related topics. Soc Netw Anal Min 11:44. https://doi.org/10.1007/s13278-021-00752-0
6. Chauhan UA, Afzal MT, Shahid A, Abdar M, Basiri ME, Zhou X (2020) A comprehensive analysis of adverb types for mining user sentiments on amazon product reviews. World Wide Web 23(3):1811–1829
7. Chen J, Hossain MS, Zhang H (2020) Analyzing the sentiment cor- relation between regular tweets and retweets. Soc Netw Anal Min 10:13. https://doi.org/10.1007/s13278-020-0624-4
8. Cui R, Agrawal G, Ramnath R (2020) Tweets can tell: activity recogni- tion using hybrid gated recurrent neural networks. Soc Netw Anal Min 10:16. https://doi.org/10.1007/s13278-020-0628-0
9. Dai Y, Liu J, Zhang J, Fu H, Xu Z, (2021) Unsupervised Sentiment Analysis by Transferring Multi-source Knowledge. Cogn Comput.https://doi.org/10.1007/s12559-020-09792-8
10. Desai M, Mehta MA (2016) Techniques for sentiment analysis of Twit- ter data: A comprehensive survey. In: 2016 International Confer- ence on Computing, Communication and Automation (ICCCA). 149–154 https://doi.org/10.1109/CCAA.2016.7813707
11. Dietterich TG, (2000) Ensemble methods in machine learning. In: Mul- tiple Classifier Systems. MCS 2000. Lecture Notes in Computer Science. 1857, 1–15. https://doi.org/10.1007/3-540-45014-9_1
12. Fatehi N, Shahhoseini HS, Wei J, Chang CT (2022) An automata algorithm for generating trusted graphs in online social networks. Appl Soft Com- put 118:108475. https://doi.org/10.1016/j.asoc.2022.108475
13. Henríquez PA, Ruz GA (2018) Twitter Sentiment Classification Based on Deep Random Vector Functional Link. In: 2018 International Joint Conference on Neural Networks (IJCNN), 1–6 https://doi.org/10.1109/IJCNN.2018.8489703
14. Hossny AH, Mitchell L, Lothian N, Osborne G, (2020) Feature selection methods for event detection in Twitter: a text mining approach. Soc Netw Anal Min 10:61. https://doi.org/10.1007/ s13278-020-00658-3
15. Islam MR, Liu S, Wang X, Xu G, (2020) Deep learning for misinfor- mation detection on online social networks: a survey and new perspectives. Soc Netw Anal Min 10:82. https://doi.org/10.1007/ s13278-020-00696-x
16. Kaur C, Sharma A, (2020). Twitter sentiment analysis on coronavirus using textblob (No. 2974). EasyChair.
17. Keyvanpour M, Karimi Zandian Z, Heidarypanah M (2020) OMLML: a helpful opinion mining method based on lexicon and machine learning in social networks. Soc Netw Anal Min 10:10. https://doi.org/10.1007/s13278-019-0622-6
18. Kumar P, Sinha A (2021) Information diffusion modeling and analy- sis for socially interacting

networks. Soc Netw Anal Min 11:11. https://doi.org/10.1007/s13278-020-00719-7

19. Kwak EJ, Grable JE (2021) Conceptualizing the use of the term finan- cial risk by non-academics and academics using twitter messages and science direct paper abstracts. Soc Netw Anal Min 11:6. https://doi.org/10.1007/s13278-020-00709-9
20. Li X, Xie Q, Jiang J, Zhou Y, Huang L, (2019) Identifying and moni-toring the development trends of emerging technologies using patent analysis and Twitter data mining: The case of perovskite solar cell technology. Technol Forecast Soc Chang 146:687–705.https://doi.org/10.1016/j.techfore.2018.06.004
21. Machuca CR, Gallardo C, Toasa RM (1828) 2021, Twitter sentiment analysis on coronavirus: Machine learning approach. J Phys Conf Series 1:012104
22. Moutidis I, Williams HTP (2020) Good and bad events: combining network-based event detection with sentiment analysis. Soc Netw Anal Min 10:64. https://doi.org/10.1007/s13278-020-00681-4
23. Ruz GA, Henríquez PA, Mascareño A (2020) Sentiment analysis of Twitter data during critical events through Bayesian networks classifiers. Futur Gener Comput Syst 106:92–104. https://doi.org/10.1016/j.future.2020.01.005
24. Sailunaz K, Alhajj R (2019) Emotion and sentiment analysis from Twitter text. J Comput Sci 36:101003. https://doi.org/10.1016/j. jocs.2019.05.009
25. Singh M, Jakhar AK, Pandey S (2021) Sentiment analysis on the impact of coronavirus in social life using the BERT model. Soc Netw Anal Min 11:33. https://doi.org/10.1007/s13278-021-00737-z
26. Symeonidis S, Effrosynidis D, Arampatzis A (2018) A comparative evaluation of pre-processing techniques and their interactions for twitter sentiment analysis. Expert Syst Appl 110:298–310. https://doi.org/10.1016/j.eswa.2018.06.022
27. Wang D, Al-Rubaie A, Hirsch B, Pole GC, (2021) National happiness index monitoring using Twitter for bilanguages. Soc Netw Anal Min 11:24. https://doi.org/10.1007/s13278-021-00728-0
28. Yadav N, Kudale O, Rao A, Gupta S, Shitole A (2021) Twitter senti- ment analysis using supervised machine learning. In: Hemanth
29. J, Bestak R, Chen JI-Z (eds) Intelligent Data Communication Technologies and Internet of Things. Springer, Singapore, pp 631–642
30. Yang Y, Hsu JH, Löfgren K, CHo W, (2021) Cross-platform com- parison of framed topics in Twitter and Weibo: machine learn- ing approaches to social media text mining. Soc Netw Anal Min 11:75. https://doi.org/10.1007/s13278-021-00772-w
31. Nuser M, Alsukhni E, Saifan A, Khasawneh R, Ukkaz D, (2022) Senti- ment analysis of COVID-19 vaccine with deep learning. J Theor Appl Inf Technol. 100(12):4513-4521.