Proceedings of the 4th International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/54/20241434

Sentiment analysis based on machine learning models

Xinya Liao

Computer Science and Technology, Nanjing University, Nanjing, China

201220152@smail.nju.edu.cn

Abstract. Sentiment analysis represents a pivotal research domain within the realm of natural language processing (NLP). Its significance lies in its capacity to scrutinize vast volumes of data originating from social networks and to offer invaluable insights. While numerous studies center on the exploration and enhancement of diverse models and techniques for sentiment analysis tasks, there is a scarcity of research dedicated to evaluating and contrasting the performance of these models. This paper undertakes an investigation to assess the efficacy of four distinct machine learning models: k-nearest neighbor (KNN), random forest, multinomial naive Bayes, and logistic regression, with the aim of shedding light on their relative effectiveness. The data in this research comes from two datasets, SST-2 and IMDB. Data from SST-2 is used for training and testing, and data from IMDB is used for further testing. The term frequency-inverse document frequency (TF-IDF) feature extraction method is integrated with the models and applied to the datasets. Results show that all the four models do well on SST-2 dataset, but KNN and random forest model perform poorly on IMDB dataset.

Keywords: Sentiment Analysis, Natural Language Processing, Machine Learning.

1. Introduction

Natural Language Processing (NLP) holds significant significance in the field of Artificial Intelligence (AI). It aims at processing real human language through computers, facilitating human-computer interaction through natural language. NLP can be employed in machine translation, speech recognition, semantic understanding and many other applications.

Natural language processing was proposed in the 1950s, and developed with the accumulation of corpora and the continuous progress of computer technology, especially artificial intelligence. In the germination period, researchers focused on rule-based systems, where explicit linguistic rules were manually defined to process and understand language; statistical approaches appeared in the 1980s, and it mainly involved building probability models of existing data to analyze new data; in early 21st century, the rise of machine learning techniques brought new energy to NLP; with the advent of deep learning, the 2010s witnessed new breakthroughs in NLP [1].

One of the crucial branches of NLP is the sentiment analysis. It is a means of assessing whether the language sample is positive or negative and is widely used in various fields including e-commerce, news media and advertising [2]. The emergence of a large number of social platforms provides sentiment analysis with perfect application scenarios. These social networks enable people to freely express their opinions about any topic, generating a massive amount of data. The analysis of this data can lead to valuable information, and sentiment analysis is introduced to achieve this. For instance, the sentiment

DOI: 10.54254/2755-2721/54/20241434

analysis of Twitter data has the ability to capture public opinion about current affairs [3]. In the business world, sentiment analysis can help enterprises to conduct market research, consumer satisfaction survey and product evaluation [4]. Companies can get feedback based on NLP of online comments and adjust their commercial strategies wisely [5].

In the early 2000s, the mainstream sentiment analysis technique was lexicon-based approach. This approach utilizes a word dictionary, which contains words and their associated sentiment scores, and the sentiment of a text sample was calculated based on the sum or mean of sentiment scores from the lexicon. SentiWordNet, for example, is a popular lexicon in sentiment [6]. Later machine learning techniques were applied in sentiment analysis. Researchers separated a labeled dataset into two sections for training and testing, and then train machine learning models like Naive Bayes, Support Vector Machines (SVM) and others on the dataset [7]. Since the 2010s, deep learning algorithms have dominated the field of sentiment analysis. Deep learning models, notably Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN), were found having a very impressive performance when applied to sentiment analysis [8]. The majority of current studies in this area concentrate on strategies to improve the precision and effectiveness of sentiment analysis tasks, like doing finer analysis by applying a syntactic parser and sentiment lexicon, and adding the steps of weight preprocessing and word density weighting [9, 10]. However, few studies have discussed how several classic machine learning models perform differently on certain topic. This research aims to fill this gap by means of a comparative study.

This research aims to employ traditional machine learning models, including multinomial naive Bayes model, k-nearest neighbor (KNN) model, random forest model and logic regression model, and discusses the accuracy and applicability of these models when applied with Term frequency-inverse document frequency (TF-IDF) to sentiment classification. Additionally, imbalanced dataset is also used in this research.

2. Method

Figure 1 shows the basic steps of this research. The process comes across preprocessing the data, extracting features from data, developing models and evaluating models.

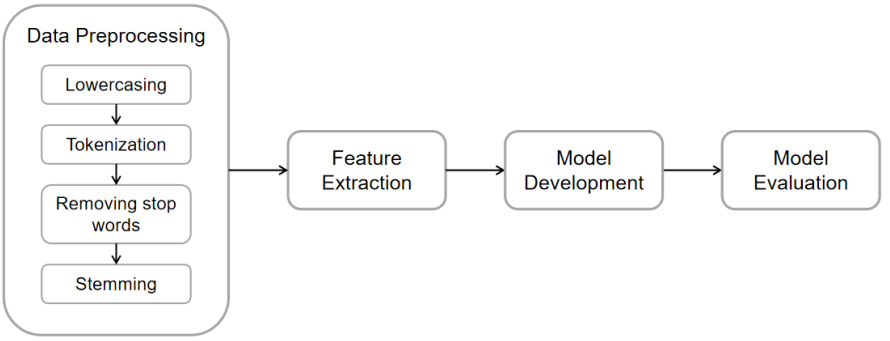

Figure 1. Basic steps of this research (Photo/Picture credit: Original).

2.1. Data Preparation

2.1.1. Dataset Introduction. The data used in certain sentiment analysis is either originally generated by the researchers or collected from available datasets. Given the challenge of generating a substantial volume of data, this study opts to utilize two pre-existing datasets that hold significant recognition within the domain of Natural Language Processing (NLP): the SST-2 dataset and the IMDB dataset.

Stanford Sentiment Treebank (SST) is one of the most popular datasets for training and testing NLP models for sentiment classification task, and the SST-2 dataset consists of thousands of sentences from SST that are labeled with binary sentiment labels, positive or negative [11]. 67, 349 sentences from this dataset are used in this research. Among them, 37, 569 sentences are labeled “positive” and 29, 780

DOI: 10.54254/2755-2721/54/20241434

sentences are labeled “negative”, so this dataset is imbalanced. The dataset is divided into two parts. One part is for training purpose and contains 50, 000 sentences, while the other part is for testing and contains the rest data.

To further assess the practicality of the models, this study proceeds to conduct testing on the IMDB dataset. IMDB comprises movie reviews sourced from the Internet Movie Database (IMDb) and is categorized into two sentiment classes: positive and negative. There are a total of 10000 reviews from this dataset that are used in this research.

2.1.2. Data Preprocessing. Data preprocessing is an essential link of NLP tasks. It cleans the text data and makes further analysis easier and more accurate. In this research, four main steps are done for data preprocessing, including lowercasing, tokenization, removing the stopwords and stemming.

Lowercasing: Since the datasets used are basically cleaned, this research doesn’t have to do text cleaning to remove the noise in the first place. The data preprocessing starts with converting all the text into lowercase, which reduces interference caused by capitalization.

Tokenization: Tokenization separates sentences into tokens, usually words or subwords. It is a crucial step to turn continuous text data into suitable format for analysis.

Removing the stopwords: In sentiment analysis, stop words such as “a”, “is”, “and” usually carry little significant information and nearly have nothing to do with emotion expression, so they are removed to reduce the volume of words and improve the efficiency. This research applies the stopwords dictionary from Natural Language Toolkit (NLTK) library in Python to detect stopwords.

Stemming: Stemming transforms words to their base form. For example, “testing” may be converted to “test”. This can lead to more accurate analysis.

2.1.3. Feature Extraction. In order to enable computers to comprehend textual data, it becomes necessary to employ feature extraction techniques such as word vectorization, which maps words into numerical vectors. In this study, the chosen method for feature extraction is Term Frequency-Inverse Document Frequency (TF-IDF). The formulas for calculating TF-IDF of a specific term are as follows.

$$TF = \frac{n}{N} \tag{1}$$

Here, n denotes the frequency with which this term appears in the sentence, and N represents how many terms there are in this sentence.

$$IDF = log(\frac{S}{S(t)}) \tag{2}$$

Here S represents the overall number of sentences in dataset, and S(t) represents the number of sentences containing term t.

$$TFIDF = TF \times IDF \tag{3}$$

After computing TF-IDF score for each term, this method manages to convert text data into matrix that shows the importance of terms.

2.2. Machine Learning Models

2.2.1. K-nearest Neighbor. K-nearest neighbor, or KNN, is a classification method that assigns class labels to data according to the majority category among its k-nearest elements by calculating the vector distance between the data to be predicted and the existing samples in the dataset. It is crucial to determine k, the number of nearest neighbors. Typically, as "k" increases, the model tends to become smoother, increasing the likelihood of the model underfitting the actual values. This research chooses 5 as the number of the nearest neighbors in order to get a robust model and avoid underfitting at the same time.

DOI: 10.54254/2755-2721/54/20241434

2.2.2. Random Forest. Random forest is a classifier that predicts samples through multiple decision trees. It works by randomly selecting data for input into decision trees and aggregating their outputs through voting to derive the final result. In this research, the random forest model is configured to have 8 decision trees and is built using bootstrapped samples.

2.2.3. Multinomial Naive Bayes. Multinomial Naive Bayes is a widely used algorithm that has a great performance on sentiment classification tasks for datasets with discrete features. It is based on Naive Bayes theorem, which describes how to calculate the probability that a sample falls into certain class.

$$P(C|F) = \frac{P(C)\times P(F|C)}{P(F)} \tag{4}$$

In this formula, P(C|F) means the probability that certain instance with feature F falls into class C. P(C) stands for the prior probability of C. P(F|C) stands for the probability that feature F appears when the class is C. P(F) is the probability that feature F appears. The class with the highest probability should be the predicted result.

When dealing with discrete data that has multiple features, multinomial naive Bayes is often used. It considers the probability distribution obeys the multinomial distribution.

2.2.4. Logistic Regression. Logistic regression extends linear regression by employing the sigmoid function to classify data into two classes depending on the features. Below is the logistic function:

$$f(x) = \frac{1}{1+e^{-\theta^T \times x}} \tag{5}$$

Here θ is the weight and x represents features. This function calculates the probability that the variable equals 1 given the feature x.

Additionally, this research sets the maximum of iterations for the algorithm as 1000 to avoid training indefinitely.

2.3. Implementation details

Four evaluation metrics are took to assess how well the four models in this research predict data from two datasets, covering accuracy, precision, recall and F1-score.

Accuracy reflects how many instances are correctly predicted out of all. Below shows the formula:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{6}$$

In this formula, TP stands for the true positives, TN for the true negatives, FP for the false positives and FN for the false negatives.

Precision shows the percentage of accurately anticipated positive instances to all positive predictions.

$$Precision = \frac{TP}{TP+FP} \tag{7}$$

Recall measures the proportion of true positive predictions to all real actual positive instances, which is able to reflect the completeness of the classification algorithm.

$$Recall = \frac{TP}{TP+FN} \tag{8}$$

The F1-score, which takes both precision and recall into account, is derived from the harmonic average of these two metrics.

$$F1 = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{9}$$

DOI: 10.54254/2755-2721/54/20241434

3. Results and Discussion

The models are first tested on the SST-2 dataset, the same dataset used for training these models. Then additional experiment is conducted, in which the models are tested on IMDB dataset that contains out-of-sample data. Table 1 shows the findings of the four models tested on both datasets.

Table 1. Comparison for models on two datasets

Model	Dataset	Accuracy	Precision	Recall	F1-score
KNN	SST-2	0.88	0.88	0.88	0.87
	IMDB	0.50	0.54	0.50	0.35
Random Forest	SST-2	0.89	0.89	0.89	0.89
	IMDB	0.67	0.68	0.67	0.67
Naive Bayes	SST-2	0.87	0.87	0.87	0.87
	IMDB	0.80	0.80	0.80	0.80
Logistic Regression	SST-2	0.88	0.88	0.88	0.88
	IMDB	0.81	0.81	0.81	0.81

It can be seen from the results that all four models have a good performance when predicting data from SST-2 dataset, since they get high accuracy, precision, recall and F1-score that is around 88%. This choice is attributed to the maturity and effectiveness of these four models in handling sentiment classification tasks. Additionally, the data from the SST-2 dataset exhibits distinct features, facilitating the models in accurately capturing underlying patterns and correctly classifying the text.

On the IMDB dataset, however, there is a noticeable difference in how well the four models perform. KNN performs worst given an accuracy of 0.50 and an F1-score of 0.35. There should be two possible explanations for this. Firstly, the model may have an overfitting problem. The model may be focused too much on the noise in the original dataset and thus fit the data from SST-2 dataset perfectly. But when it comes to unseen data, the model gets poor prediction accuracy. Secondly, in this research, the KNN algorithm's accuracy experiences a decline due to the imbalanced distribution of sample data, which is reflected in the training data. Similarly, the random forest model exhibits relatively lower accuracy and other metrics, which could be attributed to the potential overfitting, as the random forest algorithm has a propensity to overfit in the presence of noise during classification tasks. Naive Bayes model and logistic regression model perform well on the IMDB dataset with accuracy reaching 0.80 and 0.81. This may be because the naive Bayes algorithm usually has a stable performance on text classification tasks, especially for small-scale data, and logistic regression model is less likely to overfit than many other models like decision trees.

4. Conclusion

This study outlines the structural frameworks of four classical machine learning models and conducts a comparative analysis of their performance in sentiment analysis. The KNN model, random forest model, multinomial naive Bayes model, and logistic regression model are examined in conjunction with TF-IDF for handling sentiment classification tasks. Experiments are conducted to evaluate these four models on two datasets, SST-2 and IMDB. Results show that the four models all perform well on SST-2 dataset, which is used for training the models. For IMDB dataset, KNN model and random forest model have relatively poor performance while the other two models perform much better. This reveals that the four models all have the ability to handle sentiment analysis tasks, but KNN and Random Forest may be affected more when predicting unseen data. In the future, the comparative study can be extended to more models and datasets, and include more evaluation metrics. Different algorithms and techniques can be combined to advance the prediction accuracy of individual models in the subject of sentiment analysis.

DOI: 10.54254/2755-2721/54/20241434

References

[1] Zhao J Song M Gao X 2022 Research on Text Representation in Natural Language Processing (In Chinese) Journal of Software 33(01) 102-128

[2] Nandwani P Verma R 2021 A review on sentiment analysis and emotion detection from text Social Network Analysis and Mining 11(1) 81

[3] Elbagir S Yang J 2019 March Twitter sentiment analysis using natural language toolkit and VADER sentiment In Proceedings of the international multiconference of engineers and computer scientists Vol 122 p 16

[4] Wu Z 2023 Text Classification Based on Natural Language Processing and Machine Learning (In Chinese) Electronic Technology and Software Engineering216-219

[5] Zhang L Zhang B Kou H 2022 Tourist Landscape Perception of Jiangnan Ancient Town Based on Natural Language Processing of Network Comment Data (In Chinese) Journal of Chinese Urban Forestry 20(06) 125-132

[6] Sebastiani F Esuli A 2006 May Sentiwordnet: A publicly available lexical resource for opinion mining In Proceedings of the 5th international conference on language resources and evaluation pp 417-422 European Language Resources Association (ELRA) Genoa Italy

[7] Ye Q Zhang Z Law R 2009 Sentiment classification of online reviews to travel destinations by supervised machine learning approaches Expert systems with applications 36(3) 6527-6535

[8] Dang N C Moreno-García M N & De la Prieta F 2020 Sentiment analysis based on deep learning: A comparative study Electronics 9(3) 483

[9] Nasukawa T Yi J 2003 October Sentiment analysis: Capturing favorability using natural language processing In Proceedings of the 2nd international conference on Knowledge capture pp 70-77

[10] He K 2021 Research and Application of Text Classification Based on Natural Language Processing (In Chinese) Nanjing University of Posts and Telecommunications

[11] Munikar M Shakya S Shrestha A 2019 November Fine-grained sentiment classification using BERT In 2019 Artificial Intelligence for Transforming Business and Society (AITB) Vol 1 pp 1-5 IEEE