Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.9854890

IMPLEMENTING CUTTING-EDGE TECHNIQUES AND TOOLS FOR SENTIMENT ANALYSIS: A PRACTICAL ASSESSMENT

Diksha Wagh[1], Rupali A.Mahajan[2], Pallavi R. Rege[3], Vikas B. Maral[4], Nagaraju Bogiri[5]
Computer Engineering Department, Vishwakarma Institute of Information Technology, Pune India, diksha.22020099@viit.ac.in

Abstract
Sentiment analysis is a process which involves identifying and extracting emotions, feelings, and opinions from textual data. This technology has gained prominence in recent years, as more and more companies realize the possibility of understanding the emotions and opinions of their customers in order to improve products and their services. There are many sentiment analysis tools and applications on the market to meet different business needs. The applications of sentiment analysis tools are widespread and varied. In marketing, sentiment analysis tracks customer feedback and improves branding. It is also used in customer service to understand customer needs and provide individualized assistance. In politics, sentiment analysis tracks public opinion and predicts election results. It is also used in the healthcare industry to monitor the mood of patients and detect potential health problems. This study will compare different techniques of analyzing sentiments by implementing them.

Introduction

Sentiment analysis, in easy terms, is nothing but taking a polarity score that you take out of any form of input. The polarity score in sentiment analysis is negative, positive, and neutral of the received information. You can derive sentiments from an image or text. Any application nowadays will ask you for feedback in the form of sentiments. The feedback is received from users in the form of emojis. Whether you like, dislike, or love the application is the briefest way of giving your opinion of anything. On the internet today many people give their opinions on new topics. A product may get launched and people will give their strong opinions on it. Overall, determining the likeness of your product holds in people's opinion gets important for marketing and improving upon the issues of the product. Your product can be anything but feedback generally is (on the internet or especially on social media) via text. Since 2002 there have been many advances in this field. Introducing machine learning to sentiment analysis will sentiment categorization from online reviews that will also lead to creating useful decision-making, detecting false reviews, and predictive decision-making.

Literature review

Monalisha Sahoo et al. [1], experimented with the lexicon, machine learning, and deep learning approach. They determined the polarity or sentiments of a dataset by these approaches. They got an accuracy of 88.38%, 92.31%, and 67.46% and concluded that the machine learning approach was better than the deep learning and lexicon-based approach. Jaspreet Singh et al. [2], exploited classifiers such as Naive Bayes, BFTREE algorithm, OneR algorithm, and J48 algorithm for sentiment prediction using three manually annotated datasets from Amazon and IMDB websites. Their dataset consisted of reviews and feedback. Naive Bayes exhibits a

learning rate to be faster and J48 shows adequacy in the true positive and false positive rates. All in all this study shows that sentiment analysis has scope to improve processing using deep learning. Fernando Ferri et al. [3] provide a thorough analysis of different studies and classification of sentiment classification approaches and tools their advantages, disadvantages, and limitations. Different approaches such as i) machine learning (ii) hybrid (iii) lexicon-based approaches are discussed thoroughly. P. Ancy Grana [4] represents the study of machine learning methods. It is a textbook understanding of machine learning methodologies such as collecting a data set, preprocessing, fitting the data in various pre-trained models (Naive Bayes, Support Vector, and Linear Regression), and implementing Rule Based NLP approach and Automatic sentiment analysis approach using ML.

Nirag T. Bhatt et al. [5] represent the literature survey. They discuss a procedure or a theoretical review of the machine learning approach towards sentiment analysis on Twitter data which includes data collection using Twitter API, data preprocessing, feature extraction, and models like (SVM, Naive Bayes, and Random Forest Classifier). They conclude that comparing features like Unigram, TF-IDF, and Bag of Words with machine learning classifiers will show which feature will get the best accuracy out of them.

Methodology

There are three ways to calculate the sentiments of a given text.

1. Vader Lexicon
2. Machine Learning

Vader Lexicon

The sentiment analysis based on the lexicon approach is a technique used to analyze the sentiment of a given text by using predefined lists of words with assigned polarities (positive, negative, or neutral). The approach involves creating a dictionary of words and assigning a score or weight to each word, indicating the sentiment associated with that word. For example, the word "love" would have a high positive score, while the word "hate" would have a high negative score. To analyze a given text, the approach involves breaking down the text into individual words and then calculating a sentiment score. The resulting sentiment score can be positive, negative, or neutral, depending on the sum of the individual scores. So in this case if the input was given a text about Barak Obama such as,

“Barack Hussein Obama II is an American former politician who served as the 44th president of the United States from 2009 to 2017. A member of the Democratic Party, he was the first African-American president of the United States”

The polarity score for the above text will be 19.200% positive, 80.800% neutral, and 0.000% negative. The overall score turns out to be neutral. Under challenging circumstances like movie review polarity score from the Lexicon, the approach faces difficulty in keeping the score near accuracy. Here are some reviews of the latest Movie Boston Strangler.

Review 1. “Start was good . . .

But as usual, knightly performance is empty. Leaving wanting for more. Might succeed in an erotic thriller focusing on any knightly character. Where her big eyes and Unfinished robotic delivery might work but here it doesn't. Katie McGrath is her real doppelganger. And a better actress. Able to emulate feelings and emotions and guilt betrayal very well. Katie McGrath should get picture deals."

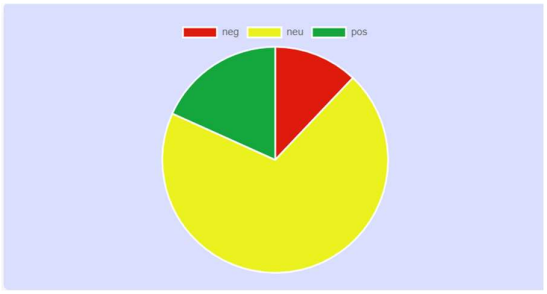


fig 1 graph of the polarity score

Review 2. "It was alright but Knightley was as ever Knightley. Bland. All big eyes stare into the camera. And so skinny. Is she anorexic? Her running mate is a much better actor. Always mix her up with her Doppelganger Keeley Hawes."

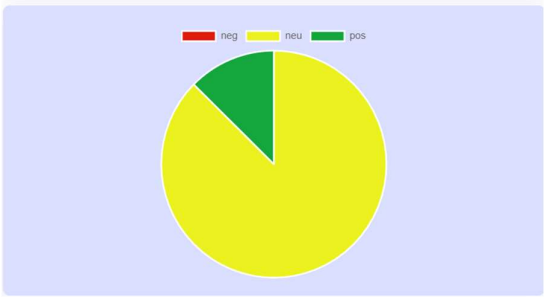
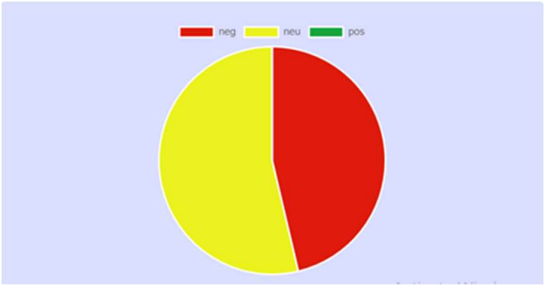


fig 2 graph of the polarity score

Review3. "Not that gripping and keiras fake American accent is horrendous"

fig 3 graph of the polarity score



As shown in the pictures no matter how positive or negative the review is, it will always show the text to be neutral. It is because this approach relies heavily on the quality and completeness of predefined dictionaries. These dictionaries may not always capture the nuances of language or the context in which the words are used. Therefore, the accuracy may be limited by the quality of the dictionary used. The overall score comes near 69%.

Machine Learning Approach

Machine learning approach includes data cleaning, data preprocessing, model selection, and evaluating the accuracy of those models

Data source

Machine learning approaches require a dataset to train and test a model. This is a short text sentiment analysis. Hence 2 class classification data such as the IMDB dataset was a perfect fit for this.

The dataset for this research, 'IMDB Dataset of 50k Movie Reivew' by Lakshmipathi N, is taken from Kaggle. The dataset contains 50 thousand rows and 2 columns named 'reviews' and 'sentiments'.

Data cleaning and data processing

Data cleaning involve removing stop words, removing any null values, lemmatization, and stemming from the text. The provided by Kaggle was pretty much clean. It did not have any null values.

The textual column labeled 'sentiment' is turned into a numerical vector. Machine learning approaches such as classifiers and learning algorithms do not process raw text data. They can only compute numerical data. This is done by the TfidfVectorizer class embedded in the sklearn library.

NaiveBayes

To perform sentiment analysis using Naive Bayes, you first need a labeled dataset of text examples with their corresponding sentiment labels (e.g., positive, negative, neutral). Then, you need to preprocess the text data by removing stop words, stemming or lemmatizing words, and converting the text into a numerical format (e.g., bag-of-words representation).

Next, you need to train a Naive Bayes classifier on the preprocessed data. Naive Bayes makes use of Bayes' theorem and assumes that the features (i.e., words in this case) are conditionally independent given the class (i.e., sentiment label). This assumption simplifies the calculation of probabilities and allows the classifier to be trained efficiently. During the process of training, the Naive Bayes estimates the probability of each word given a particular sentiment label, as well as the prior probability of each sentiment label in the dataset. These probabilities are used to calculate the posterior probability of a particular sentiment label given a new text example, using Bayes' theorem.

Finally, to classify a new text example, you simply calculate the prior probability of each sentiment label using the trained Naive Bayes classifier. Naive Bayes is an uncomplicated yet powerful technique for analyzing sentiment that has practical uses in a range of natural language processing applications.

Support Vector Machines(SVM)

SVMs are another popular machine-learning algorithm used for sentiment analysis. During training, SVM finds the optimal hyperplane that maximally separates the data points belonging to different sentiment classes. The hyperplane is determined by a decision boundary that separates the data points with the highest margin. Here the classes are positive and negative reviews. So the decision boundary will lie between these classes and calculate the minimum possible distance between them and the data points of those classes. This is how the model is trained for this dataset. After the training, it classifies new text data. If the new data points lie close to the positive-reviews decision boundary then it will be classified as reviews belonging to the positive-reviews class. The same will happen to the negative-reviews class. Overall, SVMs are effective for sentiment analysis because they can handle high-dimensional data and are robust to overfitting. However, SVMs require careful tuning of hyperparameters, such as the regularization parameter and the kernel function, to achieve optimal performance. Here the parameters passed were kernel='linear'.

Logistic Regression

Next the LogisticRegression class of the sklearn.linear_model package was used to see if a different or better outcome could be achieved. The class involves executing the logistic regression algorithm that leverages the output generated by the linear regression function and feeds it into a sigmoid function. And doing this will give output between 0 and 1 which is perfect to identify the outcome in the form of positive or negative feedback. Now this is achieved using the sigmoid function.

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

The sigmoid function converts any value between -∞ to +∞ into 0 and 1, and this can be achieved by this formula. Given below equation is the formula for the sigmoid function. The formula states that if x is any negative the value of the output will be close to 0. This can be represented by limits.

$$\lim_{x\to-\infty} \sigma(x) = \lim_{x\to-\infty} \frac{1}{1 + e^{-x}}$$

$$= \lim_{x\to-\infty} \frac{1}{1+e^{\infty}}$$

$$= 0$$

And if x is passed as any positive number the output value will be close to 1.

$$\lim_{x\to\infty} \sigma(x) = \lim_{x\to\infty} \frac{1}{1 + e^{-x}}$$

$$= \lim_{x\to\infty} \frac{1}{1+e^{-\infty}}$$

$$= 1$$

If x is passed as an exact 0 value then the output will be 0.5.

$$\sigma(0) = \frac{1}{1 + e^{-(0)}}$$

$$= 0.5$$

This is how the algorithm estimates predictions by taking linear regression output as an input and providing output in the form of 0 and 1. 0 can be considered as a negative review and 1 as a positive and 0.5 as neutral. Logistic regression provides a probabilistic output. Additionally, logistic regression is computationally efficient and works well with small and medium-sized datasets. However, logistic regression may not perform as well as other machine learning algorithms for more complex tasks or datasets with a large number of features. As with any machine learning algorithm, it is important to tune hyperparameters, such as the regularization strength and learning rate, to optimize performance.

Decision Trees

Reason behind picking a decision tree classification algorithm is, a) It imitates the way humans think, b) The logic is constructed in a tree-like structure and hence becomes very easy to understand.

During training, the algorithm builds a tree-like model. It does that by splitting the data recursively based on the 'sentiment' feature. The goal of the algorithm is to create a tree that is as small as possible while still accurately predicting the class of the data. The model then predicted the sentiment class of new text examples by traversing the tree based on the decision rules until it reaches the leaf node. The model structure of the decision tree can be visualized and the decision rules can be easily explained. However, decision trees can suffer from overfitting and instability if the tree becomes too complex or if there is noise in the data. To address these issues, Random Forest and Gradient Boosting can be used to combine multiple decision trees and improve accuracy.

Model evaluation

Model	**Accuracy**
SVM	0.84090909090909
Decision Tree	0.65909090909090
Naive Bayes	0.63484848484848
Logistic Regression	0.83030303030303

Confusion Matrix

Now to verify the effectiveness of these models confusion matrix must be used. The confusion matrix is used specifically for classification problems. For the sake of this study, a two-class classification problem is used. Two class classification problem is nothing but when a model predicts the probability between two possible outcomes. for example, predicting if a cancer is malignant or benign is a two-class classification problem. In this case, models predict whether a review is positive or negative. And to verify the accuracy of such a classification problem confusion matrix is the best method. This method will also calculate the Recall, Precision, Accuracy, and F-measure. All of them are calculated by adding the sum of four combinations and applying them to each individual formula. Below are the combinations that are used to verify effectivity.

Index	1	2	3	4	5
Actual Value	Positive	Positive	Negative	Negative	Positive
Predicted Value	Negative	Positive	Negative	Positive	Negative
Result	FN	TP	TN	FP	FN

Above are the given actual and predicted values of sentiments. The combinations stand for True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). count of each of them is summed up and that sum is used in the below formulas. Below the image is the outcome from the Jupyter Notebook editor which shows the counts TP, TN, FP, and FN. Since the SVM model turned out to be a better model than the rest confusion matrix of SVM is calculated. 290 values are True Positive, 45 False Positive, 60 False Negative, 265 True Negative

```
array([[290,  45],
       [ 60, 265]])
```

fig-4 confusion matrix of SVM

Classification Measure

A classification report plays a critical role in evaluating the performance of a machine learning model. It provides a concise summary of essential metrics like precision, recall, F1-score, and support for each class in a classification problem. The report assists in identifying areas that require improvement, especially in cases where the dataset is imbalanced or when the cost of misclassification of one class is higher than others. Generally, the classification report is utilized at the end of a machine learning project's training and evaluation phase. The report's insights guide further optimization and training of the model to enhance its performance.

	precision	recall	f1-score	support
positive	0.83	0.87	0.85	335
negative	0.85	0.82	0.83	325
accuracy			0.84	660
macro avg	0.84	0.84	0.84	660
weighted avg	0.84	0.84	0.84	660

fig-5 classification measure of SVM

Conclusion

Naive Bayes, SVM, Logistic Regression, and Decision Tree algorithms on a labeled dataset provided close-to-accurate results. However, it requires a large amount of labeled data for training and can be computationally expensive. Meanwhile, the lexicon-based approach relies on pre-defined dictionaries or lexicons of words with known sentiment scores. These lexicons can be created using various methods, such as crowd-sourcing or expert judgment. During sentiment analysis, the lexicons are used to assign sentiment scores to each word in a given text example, and the overall sentiment of the example is calculated by aggregating the scores of the individual words. This approach is simpler and faster than machine learning, but it can be less accurate, especially when dealing with words with multiple meanings or nuances. The lexicon-based approach can be easily adapted to different domains or languages, as new lexicons can be created for each domain or language. Additionally, it can be combined with other approaches, such as machine learning, to improve the accuracy of the predictions. The choice of approach depends on the specific task and the available resources. The machine learning approach is suitable for tasks that require high accuracy and have a large amount of labeled data, while the lexicon-based approach is more suitable for tasks that require quick and simple analysis or for domains that do not have a large amount of labeled data.

References

[1] Singh, J., Singh, G., & Singh, R. (2017). Optimization of sentiment analysis using machine learning classifiers. Human-centric Computing and Information Sciences, 7, 1-12.

[2] Bhatt, N. T., & Swarndeep, S. J. (2020). Sentiment analysis using machine learning technique: a literature survey. Int. Res. J. Eng. Technol.(IRJET), 7, 12.

[3] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams engineering journal, 5(4), 1093-1113.

[4] Kumari, S. P. S. G. S., & Yadav, P. Sentiment Analysis Techniques and Approaches.

[5] Alessia, D., Ferri, F., Grifoni, P., & Guzzo, T. (2015). Approaches, tools, and applications for sentiment analysis implementation. International Journal of Computer Applications, 125(3).

[6] Umarani, V., Julian, A., & Deepa, J. (2021). Sentiment analysis using various machine learning and deep learning Techniques. Journal of the Nigerian Society of Physical Sciences, 385-394.

[7] Chiny, M., Chihab, M., Bencharef, O., & Chihab, Y. (2021). Lstm, vader and tf-idf based hybrid sentiment analysis model. International Journal of Advanced Computer Science and Applications, 12(7).

[8] Singh, J., Singh, G., & Singh, R. (2017). Optimization of sentiment analysis using machine learning classifiers. Human-centric Computing and information Sciences, 7, 1-12.

[9] Yue, L., Chen, W., Li, X., Zuo, W., & Yin, M. (2019). A survey of sentiment analysis in social media. Knowledge and Information Systems, 60, 617-663.

[10] Nandwani, P., & Verma, R. (2021). A review on sentiment analysis and emotion detection from text. Social Network alysis and Mining, 11(1), 81.

[11] Cui, J., Wang, Z., Ho, S. B., & Cambria, E. (2023). Survey on sentiment analysis: evolution of research methods and topics. Artificial Intelligence Review, 1-42.

[12] Ligthart, A., Catal, C., & Tekinerdogan, B. (2021). Systematic reviews in sentiment analysis: a tertiary study. Artificial Intelligence Review, 1-57.

[13] Wankhade, M., Rao, A. C. S., & Kulkarni, C. (2022). A survey on sentiment analysis methods, applications, and challenges. Artificial Intelligence Review, 55(7), 5731-5780.

[14] Zhou, Z. G. (2022). Research on Sentiment Analysis Model of Short Text Based on Deep Learning. Scientific Programming, 2022.

[15] 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.

[16] Dabade, M. S. (2021). Sentiment analysis of Twitter data by using deep learning And machine learning. Turkish Journal of Computer and Mathematics Education (TURCOMAT), 12(6), 962-970.

[17] Swathi, T., Kasiviswanath, N., & Rao, A. A. (2022). An optimal deep learning-based LSTM for stock price prediction using twitter sentiment analysis. Applied Intelligence, 52(12), 13675-13688.

[18]Bose, R., Dey, R. K., Roy, S., & Sarddar, D. (2019). Analyzing political sentiment using Twitter data. In Information and Communication Technology for Intelligent Systems: Proceedings of ICTIS 2018, Volume 2 (pp. 427-436). Springer Singapore.

[19] Overview on Methods for Mining High Utility Item set from Transactional Database", International Journal of Scientific Engineering and Research (IJSER), ISSN (Online): 2347-3878,Volume 1 Issue 4, December 2013