A Comparative Study On Sentiment Lexicons For Automatic Labeling

Zohra Mehenaoui[1,†], Chayma Merabti[2,†], Houda Tadjer[1,†] and Yacine Lafifi[1,†]

[1]LabSTIC laboratory, University 8 Mai 45, Guelma, Algeria
[2]Computer Science Department, University 8 Mai 45, Guelma, Algeria

Abstract

Sentiment analysis is a natural language processing task that involves extracting meaningful information concerning people's opinions and sentiments towards products, services, and more, which can be utilized in several applications. This task requires using data presented on online platforms. With the increasing use of the World Wide Web, a huge amount of data can be exploited. To do so, this data should be present in a suitable format like datasets. A sentiment analysis dataset generally requires reviews or comments and their sentiment labels (positive, negative, or neutral). Experts can do the labeling task manually, but this requires a lot of time and energy, especially when dealing with a massive data size.In this paper, we performed automatic labeling of a dataset consisting of 5200 comments of students using different well-known sentiment analysis tools which are VADER, TextBlob, SpaCy, and SentiWordNet, making a comparison of these tools to find the most efficient one for automatic labeling of this dataset. The results showed that TextBlob outperforms the other tools with an accuracy of 92% and an F1-score of 89%.

Keywords

Sentiment analysis, labeling, VADER, TextBlob, SentiWordNet, SpaCy

1. Introduction

Online reviews reflect user sentiments and preferences. At least 32% of users rate products on shopping sites. 33% of them leave reviews, with 88% trusting them. [1]. Sentiment Analysis (SA), also known as Opinion Mining (OM) is a Natural Language Processing (NLP) activity that entails gathering relevant data to determine people's opinions and sentiments on products, services, etc [2]. The main process of sentiment analysis is sentiment classification, this process can be performed generally using two approaches:the lexicon-based approach and the machine learning-based approach, some of them went to use the hybridisation of both of them.

The lexicon-based approach relies on sentiment lexicons, that are dictionaries containing a collection of sentiment words labeled with their corresponding polarity (positive, negative, or neutral) and their sentiment scores. The sentiment of the entire statement is generally determined by either adding all scores of sentiment words or calculating their mean [3]. This approach is divided into two approaches: the dictionary-based approach and the corpus-based approach: The first approach requires using a set of sentiment words collected manually, then creating a dictionary by adding more words like their synonyms and antonyms, for example : WordNet [4]. The corpus-based approach adds sentiment words that are specific to the Study domain.[3].

The machine learning-based approach utilizes supervised learning for sentiment classification, employing labeled datasets. These datasets are typically partitioned into training and testing sets. The training set is used to train the classifiers of machine learning such as Support Vector Machine (SVM), Naïve Bayes (NB), Logistic Regression (LR), and others. Subsequently, the testing set is employed to assess machine learning model's performance.

13th International Conference on Research in ComputIng at Feminine, May 20–21, 2024, Constantine, Algeria
mehenaoui.zohra@univ-guelma.dz (Z. Mehenaoui); merabti.chayma@univ-guelma.dz (C. Merabti); tadjer.houda@univ-guelma.dz (H. Tadjer); lafifi.yacine@univ-guelma.dz (Y. Lafifi)
0000-0002-6732-7839 (Z. Mehenaoui); 0009-0006-7254-6215 (C. Merabti); 0000-0001-7624-1343 (H. Tadjer); 0000-0001-8232-4196 (Y. Lafifi)

To enhance the accuracy of the classification task and leverage the strengths of both approaches, many researchers went for the hybridization of these methodologies in sentiment classification, termed the hybrid approach.

Utilizing either lexicon-based or machine learning-based approaches necessitates labeled datasets, which entails assigning labels to individuals' reviews or comments regarding entities, such as products. This annotation task typically demands manual effort, resulting in a slow and costly process, particularly when dealing with large datasets. Consequently, some choose automatic labeling methods to reduce time and resources. This paper presents a comparative study of four well-known and used sentiment lexicons and tools VADER, TextBlob, SentiWordNet, and SpaCy to Auto-label dataset from Mark My Professor educational website.

The rest of the paper is organized as follows: Section 2 reviews existing literature pertinent to our study, linking ideas and providing context. Section 3 details the methodology followed by the comparative study presented in this paper. The obtained results are presented and discussed in Section 4, further elucidating our study's outcomes. Finally, Section 5 will serve as the study's conclusion, summarizing key findings and offering insights for future research.

2. Related works

With the increasing use of online platforms, users' reviews and comments have become exploitable data for researchers who apply sentiment analysis in various application domains. Some of these studies were done using available datasets such as SST [5], SemEval [6] and more. Therefore, many researchers collected their own datasets or used unlabeled ones, so they performed the labeling task; some of them labeled their datasets manually. For instance, Liu et al., [7] used manually labeled data to train a language model. On the other hand, several studies have used sentiment lexicons to label the dataset automatically. For example, Isnan et al., [8] applied sentiment analysis to the data collected from TikTok reviews on Google Play, where they used VADER for the initial labeling (positive, negative, and neutral) and performed sentiment classification using the SVM classifier. Borg and Anton [9] used VADER along with a Swedish sentiment lexicon to initially label 168010 e-mails to classify sentiments. There are a few studies where the authors used ratings to label their datasets, with each review having its corresponding rating. In the study of building a sentiment analysis model for an android Application named KlikIndomaret during COVID-19 pandemic using VADER lexicon and transformers NLTK Library, the labeling task was done based on stars rating of reviews on Indonesian shopping application in Google play store [10]. Tama et al., [11] applied sentiment analysis to the dataset of Grocery and Gourmet Food from Amazon after labeling it using star ratings of the reviews, where they compared two labeling methods named Average and Binary. The average method labels based on the average rating adjusted to the amount of data available, while the binary labeling divides labeling by using certain assumptions. The results showed that the average method performed better than the binary method.

Bonta et al., [12] compared NLTK, TextBlob, and VADER for movie review classification, finding VADER outperformed the others (77% accuracy vs. TextBlob 74% and NLTK 62%). TextBlob was compared with SpaCy in another study where the results showed that TextBlob was faster than SpaCy while SpaCy produced more accurate results and showed the results in visual forms using charts and graphs [13].

An evaluation of different well-known lexicons was conducted on two Twitter datasets. The results from the Stanford dataset demonstrated that VADER outperformed other lexicons with an accuracy of 72%. AFINN-111 and Liu-Hu achieved 65%, while SentiWordNet and SentiStrengh achieved 53% and 67%, respectively.[14].

Another comparative study was conducted by Biswas et al., [15] in which they compared three automatic lexicon-based sentiment labeling techniques: TextBlob, VADER, and AFINN, to assign sentiments to two tweet datasets, SemEval-2013 and SemEval-2016, without any human assistance. The AFINN labeling technique achieved the highest accuracy of 80.17% in the first dataset and 80.05% in the

second using a BiLSTM deep learning model.

Automatic labeling of datasets can be performed using machine learning classifiers too where in the study of Jazuli et al.,[16], the authors used K-Nearest Neighbors algorithm method to improve the accuracy of sentiment analysis, the results showed that using K-nearest neighbors gave the accuracy of 79.43% with a value of k=15 after using it with 1.409 data. Most of the works cited earlier have worked on datasets derived from social networks. In this study, we test labeling tools on a dataset from a different context, which is that of e-learning, since our objective is to leverage learner reviews in online learning environments to enhance the learning process.

3. Methodology

This section describes the methods used to achieve the study's objectives, which involve applying sentiment lexicons to label the dataset used in the research.

3.1. Dataset

We used the dataset collected from Mark My Professor website: a Hungarian website dedicated to evaluate higher education teachers and trainers by their students [17]. This dataset consists of 5200 reviews from learners regarding the courses presented by their professors: 3372 positive reviews, 982 negative reviews, and 846 neutral reviews. These reviews are collected in Hungarian language, we used the English translation of the comments to examine the chosen tools.

3.2. Sentiment lexicons

Sentiment lexicons are lexical resources used for sentiment analysis. They contain lexical units with their sentiment polarities or sentiment scores used to determine the overall sentiment of the written text[18], sentiment lexicons that are used in this study are:

3.2.1. VADER

It's a vocabulary and rule-based sentiment analysis tool that is especially adapted to the sentiments expressed on social media. VADER is an acronym for Valence Aware Vocabulary and Sentiment Reasoner, created by Hutto and Gilbert [19] to address the issue of interpreting the vocabulary, symbols, and writing style found in social media. With its ability to distinguish between the text's emotional strength and polarity (positive, neutral, or negative), the authors have made the lexicon's Python code public as open source. VADER is widely used in social media platforms such as Twitter due to its ability to recognize abbreviations and written emojis. [14]. Although the dataset used in our study was not taken from social media but due to its wide range of use, we aimed to assess its performance in e-learning reviews.

VADER doesn't require any preprocessing thanks to its handling of extra letters (like 'gooood'), emojis, capitalization, etc. It can be installed easily with this Python instruction:

```
pip install VADERSentiment
```

It analyzes every word in the sentence to see if that word is included in the VADER lexicon. By applying the ‘`polarity_scores()`‘ function, it finds polarity indices and returns the metric values of positive, negative, and neutral, as well as the compound score, which is the calculation of the sum of the normalized polarity indices. The scores range from -1 to +1, where a score of -1 indicates the most extreme negative sentiment, while a score of +1 indicates the most extreme positive sentiment. To determine the overall sentiment of a statement, standardized thresholds are set and used for the classification process. We used the typical threshold values, which are:
For text with positive Sentiment, the compound score is >= 0.05 , for text with neutral Sentiment, the compound score is > -0.05 and < 0.05 and for text with negative Sentiment, the compound score is <= -0.05. Table 1 shows some of VADER classification of three samples token from the dataset.

Table 1
Samples of VADER classification

Review	Neg	Neu	Pos	Compound	sentiment
Useful subject and interesting presentation.	0.0	0.349	0.651	0.6808	Positive
Totally correct, fulfilling requirement.	0.0	1.0	0.0	0.0	Neutral
I'm very disappointed with the quality of feedback on my last assignment.	0.176	0.824	0.0	-0.4576	Negative

3.2.2. TextBlob

It is a preferred open-source, easy-to-use Python NLP library used for text processing, encompassing tasks such as sentiment analysis through labeling, tokenization, etc. It features a sentiment property that yields a tuple in the form of Sentiment (polarity, subjectivity), where the polarity ranges from -1.0 to 1.0 (from highly negative to highly positive) and subjectivity from 0.0 to 1.0 (from highly objective to highly subjective). [13].Same as VADER, TextBlob classifies the sentiment of a given text using a threshold. The one that we are using is the same as VADER's. We used TextBlob by importing it in Python using the instruction:

```
from TextBlob import TextBlob
```

Table 2 shows three samples token from the dataset and their classification using TextBlob.

Table 2
Samples of TextBlob classification

Review	polarity	sentiment
I got max points in my exam :D	1.0	Positive
Totally correct, fulfilling requirement.	0.0	Neutral
I'm feel intruth anxious about the approaching examination .	-0.125	Negative

3.2.3. SentiWordNet

It is a tool that analyzes sentiments using WordNet, assigning scores based on evaluations by judges to word sets regarding positivity, negativity, or neutrality. Scores are assigned with a numerical range from 0 to 1, where higher values indicate positivity and vice versa. Used in NLP to gauge the tone of words and phrases in written content for sentiment analysis, opinion mining, and text categorization. [14].

The overall sentiment of a statement using SentiWordNet is positive when the positive score is greater than the negative score, negative when the negative score is greater than positive score and neutral otherwise. positive and negative scores are the degree of positive and negative assigned to each text, the degree of objectivity can be calculated as 1 - (positive score + negative score). We used the version of SentiWordNet available in NLTK corpus by importing it with the instuction:

```
from nltk.corpus import sentiwordnet as swn
```

Table 3 shows some dataset's relevant words classification using SentiWordNet.

3.2.4. SpaCy

Is a fast and efficient Python natural language processing (NLP) library that utilizes ML. It provides pre-trained models for various languages and works based on an array of features for text handling: tokenization, parts of speech (POS) tagging, entity recognition, and dependency parsing. It comes with a friendly interface and brief documentation, making it preferred by scholars, programmers for chatbots, sentiment analysis, etc. [13]. Using SpaCy requires the installation of the tool with the instuction:

```
pip install SpaCy
```

Table 4 shows six samples of relevant used words in the dataset.

Table 3
Samples of SentiWordNet classification

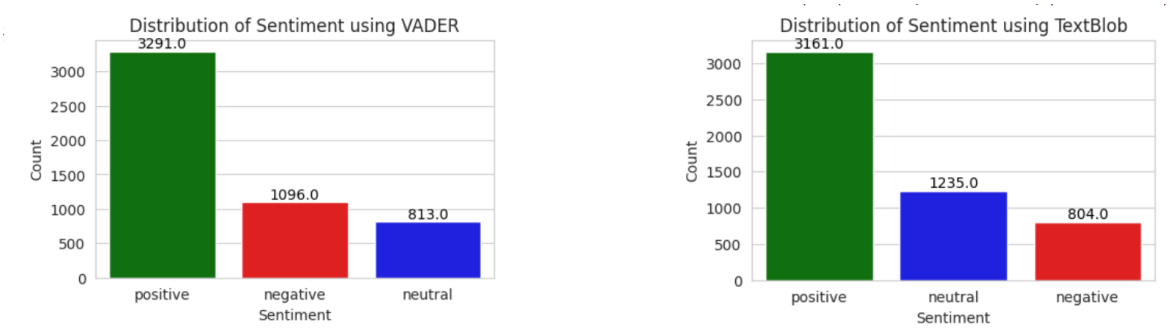
word	Pos score	Neg score	sentiment
Like	0.125	0.0	Positive
Teacher	0.0	0.0	Neutral
Disaster	0.0	0.375	Negative
Disappoint	0.0	0.25	Negative
Good	0.5	0.0	Positive
Exercise	0.0	0.0	Neutral

Table 4
Samples of SpaCy classification

word	Pos score	Neg score	sentiment
benefit	1.00	0.00	Positive
student	0.00	0.00	Neutral
correct	0.00	0.00	Neutral
scared	0.00	1.00	Negative
bad	0.00	1.00	Negative
honored	1.00	0.00	Positive

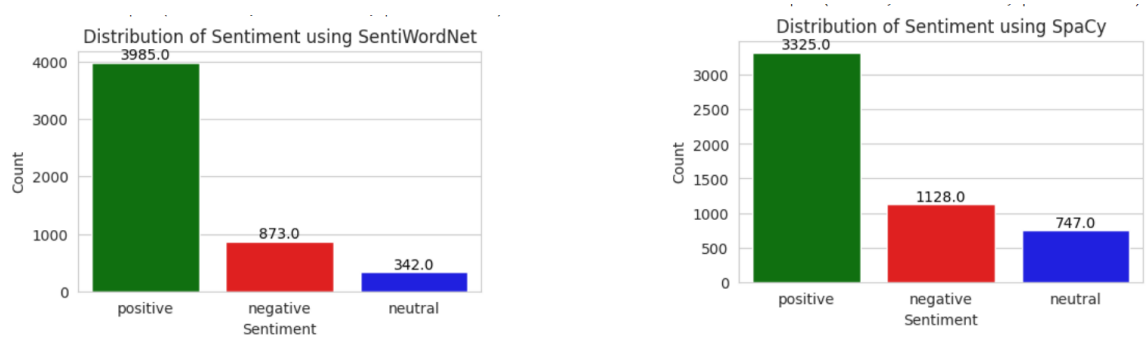
4. Results and discussions

After applying the labeling of the chosen dataset, the classification results using VADER and TextBlob are illustrated in figure 1, and SpaCy and SentiWordNet in figure 2 .



(a) VADER (b) TextBlob

Figure 1: Classification Results using VADER and TextBlob



(a) SentiWordNet (b) SpaCy

Figure 2: Classification Results using SentiWordNet and SpaCy

Evaluating tool performance in auto-labeling requires assessing metrics like:

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

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Where :
TP: True Positive, TN: True Negative, FP: False Positive, FN False Negative.

The results of measuring these performance metrics showed that TextBlob reached the highest accuracy 92.35%, SpaCy and VADER were close to each other with accuracy of 77.62% and 77.06% respectively, while SentiWordNet had the lowest accuracy of 71.40%.

TextBlob outperformed the other tools with F1-score too where it reached F1-score of 89.23% , VADER and SpaCy were close with F1-score of 70.78% and 71.68% respectively, and SentiWordNet fell to the value of 56.43%. Table 5 summarizes performance metrics results of the four tools.

Table 5
Evaluation metrics of the used tools

Tool	Accuracy	F1-score	Precision	Recall
VADER	0.7705	0.7078	0.7056	0.7114
TextBlob	**0.9234**	**0.8923**	**0.8937**	**0.9172**
SentiWordNet	0.7140	0.5643	0.6597	0.5381
SpaCy	0.7761	0.7168	0.7217	0.7156

Since VADER did not achieve good results, even though he proved his effectiveness in labeling, it primarily specialises in social media, while the dataset used was taken from an educational website. The same goes for SpaCy, as it used VADER to classify sentiments; therefore his results do not differ much from VADER. Rather, it is considered an improvement in its results. SentiWordNet performed not well, even though the dataset was not large, and it took the most time during the process of calculating the sentiment score, in contrast TextBlob performed the best with ease and speed of use, which makes it a good choice in data similar to this context.

One of the limitations of the lexicon-based tools such as the tools we worked with is that they can't detect sarcasm because they don't take into consideration the semantic meaning of the sentence, so that's a challenge for them.

5. Conclusion

Labeling a collected or unlabeled dataset presents a significant challenge within the research community due to its possible impact on the reliability of research findings, particularly those sensible to even minor errors. Manual labeling can be expensive process in terms of time and expert resources. To address this challenge, we examined well-known sentiment lexicons used researchers in their investigations, focusing on their application within e-learning domains using a dataset outlined in previous sections. Our analysis revealed that TextBlob outperformed other tools, achieving an accuracy of 92.34% and an F1-score of 89.23%. While SpaCy and VADER exhibited relatively close performance, with lower accuracy of 77.62% and 77.06%, respectively, SentiWordNet displayed the lowest accuracy at 71.40%.
Recognizing that a lexicon's efficacy and limitations may dependent on the dataset and context, it is plausible that SentiWordNet and VADER may excel under different circumstances. Therefore, TextBlob's supremacy in this study does not unequivocally consider it as the optimal tool for lexicon-based labeling sentiment analysis datasets. Furthermore, developing a lexicon tailored specifically for online reviews related to e-learning, coupled with its application on similar datasets, could yield enhanced performance. Additionally, the utilization of machine learning techniques like semi-supervised learning, deep learning

models and transformers (such as DistilBERT transformer) for labeling holds promise for delivering more precise results due to their effectiveness in capturing semantic relationships among words.

Declaration on Generative AI

During the preparation of this work, the author(s) used ChatGPT to correct errors and improve the clarity of certain paragraphs, as well as Grammarly for grammar and spelling checks. All content generated or suggested by these tools was critically reviewed and edited by the authors. The author(s) affirm full responsibility for the accuracy, originality, and integrity of the final manuscript.

References

[1] N. Vedavathi, A. K. KM, E-learning course recommendation based on sentiment analysis using hybrid elman similarity, Knowledge-Based Systems 259 (2023) 110086.

[2] M. Birjali, M. Kasri, A. Beni-Hssane, A comprehensive survey on sentiment analysis: Approaches, challenges and trends, Knowledge-Based Systems 226 (2021) 107134.

[3] P. Nandwani, R. Verma, A review on sentiment analysis and emotion detection from text, Social network analysis and mining 11 (2021) 81.

[4] B. Liu, Sentiment analysis and opinion mining, Springer Nature, 2022.

[5] T. Chen, R. Xu, Y. He, X. Wang, Improving sentiment analysis via sentence type classification using bilstm-crf and cnn, Expert Systems with Applications 72 (2017) 221–230.

[6] X. Ma, J. Zeng, L. Peng, G. Fortino, Y. Zhang, Modeling multi-aspects within one opinionated sentence simultaneously for aspect-level sentiment analysis, Future Generation Computer Systems 93 (2019) 304–311.

[7] K.-L. Liu, W.-J. Li, M. Guo, Emoticon smoothed language models for twitter sentiment analysis, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 26, 2012, pp. 1678–1684.

[8] M. Isnan, G. N. Elwirehardja, B. Pardamean, Sentiment analysis for tiktok review using vader sentiment and svm model, Procedia Computer Science 227 (2023) 168–175.

[9] A. Borg, M. Boldt, Using vader sentiment and svm for predicting customer response sentiment, Expert Systems with Applications 162 (2020) 113746.

[10] A. Budianto, B. Wirjodirdjo, I. Maflahah, D. Kurnianingtyas, Sentiment analysis model for klikindomaret android app during pandemic using vader and transformers nltk library, in: 2022 IEEE international conference on industrial engineering and engineering management (IEEM), IEEE, 2022, pp. 0423–0427.

[11] V. Tama, Y. Sibaroni, et al., Labeling analysis in the classification of product review sentiments by using multinomial naive bayes algorithm, in: Journal of Physics: Conference Series, volume 1192, IOP Publishing, 2019, p. 012036.

[12] V. Bonta, N. Kumaresh, N. Janardhan, A comprehensive study on lexicon based approaches for sentiment analysis, Asian Journal of Computer Science and Technology 8 (2019) 1–6.

[13] M. Pandey, R. Williams, N. Jindal, A. Batra, Sentiment analysis using lexicon based approach, IITM Journal of Management and IT 10 (2019) 68–76.

[14] M. Al-Shabi, Evaluating the performance of the most important lexicons used to sentiment analysis and opinions mining, IJCSNS 20 (2020) 1.

[15] S. Biswas, K. Young, J. Griffith, A comparison of automatic labelling approaches for sentiment analysis, arXiv preprint arXiv:2211.02976 (2022).

[16] A. Jazuli, W. Widowati, R. Kusumaningrum, Auto labeling to increase aspect-based sentiment analysis using k-nearest neighbors method, in: E3S Web of Conferences, volume 359, EDP Sciences, 2022, p. 05001.

[17] I. Bouacida, Sentiment Analysis and Opinion Mining Techniques for Learning Analytics, Master's thesis, Eötvös Loránd University, Budapest, Hungary, 2018.

[18] R. S. Jagdale, V. S. Shirsat, S. N. Deshmukh, Review on sentiment lexicons, in: 2018 3rd International Conference on Communication and Electronics Systems (ICCES), IEEE, 2018, pp. 1105–1110.
[19] C. Hutto, E. Gilbert, Vader: A parsimonious rule-based model for sentiment analysis of social media text, in: Proceedings of the international AAAI conference on web and social media, volume 8, 2014, pp. 216–225.