Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2016) 24: 4481 – 4496

doi:10.3906/elk-1502-173

Open source software adoption evaluation through feature level sentiment analysis using Twitter data

Muhammad Touseef IKRAM[1,*], **Naveed Anwer BUTT**[2], **Muhammad Tanvir AFZAL**[3]
[1]Department of Computer Science, Capital University of Science and Technology, Islamabad, Pakistan
[2]Department of Computer Science, University of Gujrat, Gujrat, Pakistan
[3]Department of Computer Science, Capital University of Science and Technology, Islamabad, Pakistan

Received: 20.02.2015 • **Accepted/Published Online:** 11.08.2015 • **Final Version:** 20.06.2016

Abstract: Adopting open source software from the Internet, developers often encounter the problem of accessing the quality of candidate software. To efficiently adopt the system they need a sort of quality guarantee regarding software resources. To assist the developer in software adoption evaluation we have proposed a software adoption assessment approach based on user comments. In our proposed approach, we first collected the textual reviews regarding the software resource, assigned the sentiment polarity (positive or negative) to each comment, extracted the adoption aspect which the comment talks about, and then based on the adoption aspects of the software generated an aggregated sentiment profile of the software. Twitter micro-blogging data about OSS products were crawled, preprocessed, tagged, and then summarized. To evaluate the proposed model, a set of experiments was designed and conducted using different classifiers, i.e. Apriori, GSP, and AdaBoost. For the feature level sentiment summarization we have used Bayesian statistics and frequency distribution techniques. The results show that the proposed approach achieved satisfying precision and recall, i.e. above 80% along with an average accuracy of 70.98%.

Key words: Classification algorithms, open source software reviews, sentiment classification, software adoption, supervised machine algorithms, Twitter

1. Introduction

The advent of Web 2.0 and social media, and the rise of e-commerce as a new shopping and marketing channel have dramatically changed the way people express their opinions about different products and services [1]. Twitter has become a rapidly growing user base to express opinions about product preferences, services, marketing campaigns, social events, political movements, firms, general people, and almost anything collectively called user generated content [2,3].

The focus of our work is on open source software product reviews. To adopt an OSS product developers have to judge the quality of these software resources. Weaknesses in a particular OSS product may lower customer satisfaction, resulting in OSS adoption failure. To adopt an OSS product, users have an option to read others' comments on OSS products from OSS forums and social media like Twitter. However, manually reading thousands of comments, opinions, and sentiments is not feasible. Therefore, this research aims at analyzing OSS product reviews and comments using sentiment analysis and text mining approaches. The analysis was conducted to indicate the salient emotions in OSS reviews.

*Correspondence: touseefgrw@hotmail.com

In this paper we try to implement the above-mentioned process using a semiautomatic way to extract software quality from user reviews to present the ranked OSS software in different adoption aspects. Secondly, it focuses on feature level sentiment summarization of different OSS adoption aspects in a way that will help adopters to select and implement OSS solutions. First, we extract a large number of tweets about several open source software products and their adoption aspects. Then we train the system by these labeled tweets as positive, negative, and neutral. Further, we use different techniques like Apriori, GSP, AdaBoost, Bayesian statistics, and frequency distribution for sentiment classification and summarization of OSS product reviews about nine OSS adoption factors.

We used this model to extract, analyze, and summarize the developer's sentiment regarding various facets of the OSS such as functionality, support availability, configurability, compatibility, and reliability. The proposed approach has the following main phases: 1) user review crawling, 2) data preprocessing, 3) feature and sentiment extraction, 4) sentiment strength analysis, 5) sentiment classification and summarization, 6) comparative feature analysis. A comparison with the state of the art depicts that: (a) existing techniques are based on one or more quality aspects of OSS resources [4]; (b) they are mainly focused on code analysis and architectural analysis [5].

The rest of the paper is organized as follows: in section 2, state-of-the-art work done is discussed. Section 3 summarizes the proposed methodology for sentiment analysis of OSS products. Results of experiments are presented in section 4. In section 5 we give the conclusions of the research work.

2. Related work

In this section we present the exploration of existing literature with reference to three different topics: sentiment analysis, OSS adoption factors, and software adoption sentiment analysis.

2.1. Sentiment analysis

Sentiment analysis refers to the extraction of subjective and factual material from users' opinions and attitudes to get an insight for decision making [6]. Sentiments are usually expressed about a particular topic either at topic level or aspect level [7,8]. Singh et al. [9] have used aspect level sentiment analysis for movie reviews by assignment of a sentiment label to each aspect and then generating an aggregated and net sentiment profile for the movie combining all the features. In addition, a comparison of the document level sentiment with the aspect level sentiment depicts an accurate and focused sentiment profile generation with the later one [10]. Feature level sentiment constitutes the domain specific knowledge. Most of the well-known techniques for feature level sentiment analysis require manual intervention by the domain expert and, in particular, the analysis of a single opinion aspect [11]. Wang et al. [12] have focused on aspect level review summarization by elucidating the most representative review sentences for each mined feature of the product. Furthermore, a case study has been carried out to validate the sumView system followed by a user survey for satisfaction evaluation [13].

2.2. OSS adoption factors

In this study we have restricted ourselves to open source software. OSS adoption is a well-defined domain, and a number of adoption factors have been extracted from the existing literature and grouped in technological, organizational, and environmental factors [14]. However, domain-specific aspects of OSS adoption are not used by contemporary systems, for example, cost saving, functionality, trialability, and quality characteristics (i.e. reliability, usability, compatibility, and scalability). These are common in the literature for being technological factors of OSS adoption [15]. Factors contributing to organizational category are capital investments, innovativeness, and staff IT capacity [16]. The environmental perspective of factors brings external effects to OSS adoption. Government policies, availability of support, market conditions, and success stories of adoption are among the factors contributing to this category [17].

2.3. Sentiment analysis of software aspects

Different approaches have been proposed for the opinion mining of software usability evaluation. Some of these are intended for software usability improvement by extracting knowledge from opinions [18]. Visualization of users' opinions for usability of the software makes its representation and comparison easier [19]. Galster and Tofan [20] investigated the effect of variability in functionality and quality characteristics of OSS products with Twitter data. The study aimed to understand and explore the feasibility of using micro-blogging in the context of the reliability of OSS products. Dehkharghani and Yilmaz [21] studied the extraction of the quality attributes from software based reviews by applying IDF to determine the document frequency of words in large numbers of tweets. Leopairote et al. [22] proposed a software quality assessment approach based on searching the user comments regarding a software resource automatically. Then the comments about the software quality aspects are extracted and classified as positive and negative.

Summing up our literature review it can be argued that there are only a few studies determining and predicting the adoption of new software on the basis of user sentiments and to the best of our knowledge most of them are focused on topic or product level.

3. Proposed system

The proposed system is designed to collect the comments about OSS resources and then determine the positive/negative sentiments about OSS products on the basis of different aspects of adoption. It then evaluates the subjective adoption threshold with sentiment analysis techniques to determine the adoptability of the candidate OSS resource. The system contains the following modules, which are presented in Figure 1.

3.1. Online review crawling

Different people write opinions about different features of OSS on product review websites. Reviews on these features of the OSS products are gathered using web crawling, which is one of the basic tasks of the proposed model. The crawling phase takes into account OSS adoption features like functionality, cost saving, documentation, support availability, reliability, security, and compatibility. Free format types of reviews are collected from the Internet and stored in a review database using a two-stage methodology. At the first stage information about different OSS products is collected and then reviews are crawled according to the product list as the second stage. The reviews and OSS products are related by the product ID. Six files of OSS product reviews in free format are crawled as training and testing datasets through the Twitter streaming API.

3.2. Data preprocessing

The next phase in the proposed system is the feature preprocessing. The reviews and tweets collected and downloaded from the crawling phase are free text with relevant and irrelevant information. In this phase irrelevant information, for example, special characters ("@"), punctuation ("!", "?"), URLs, symbols, HTML tags, re-tweets, spelling mistakes, and data about the reviewer is filtered out. At this step this extraneous information is removed. The next step is the removal of stop words from tweets. Stop words are the words

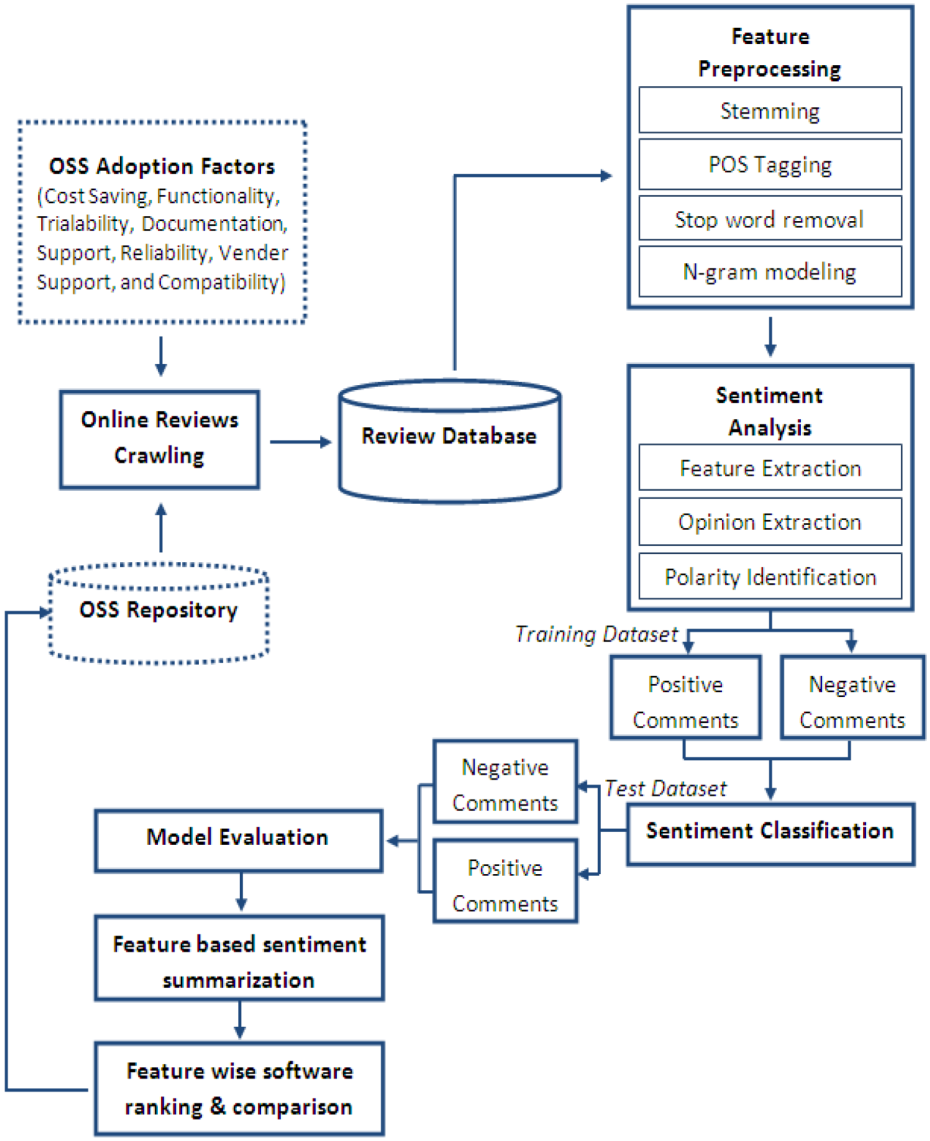

Figure 1. Proposed model for sentiment analysis.

which often appear in the reviews but have little lexical meaning, for example, “as well as”, “which”, “where”, “what”, “that”, “the”, “at”, “is”, and “on”.

The next function that is performed on the corpus is stemming in which the words are reduced to their term/root form by reducing suffixes. One of the most important advantages of stemming is to increase the hit rate of similar and identical words. Words and their associated stems are shown in Table 1.

As this research concentrates on feature level sentiment orientation, it is necessary to learn (or check) every word as a noun, adjective, or adverb. For this purpose part of speech (POS) tagging is used to identify features as well as their respective adjectives. Different POS taggers are freely available like Stanford POS tagger, maximum entropy POS tagger, TreeTagger, and Go tagger. We have used GO tagger, an online available linguistic parser, to parse the reviews. Afterwards the tagged reviews are stored in the review database containing the POS tag information relating to each word. Table 2 presents the description of how the reviews are POS tagged.

Table 1. Words and corresponding stems.

• Reliability → reliable
• Security → secure
• Functionality, Functionally → function
• Configuration, configurability → configure

Table 2. Reviews along POS tags.

"Your U.S. government uses open source software, and loves it; offers good functionality"
"Your_PRP$ U.S._NNP government_NN uses_VBZ open_JJ source_NN software_NN ,_, and_CC loves_VBZ it_PRP ;_: offers_VBZ good_JJ functionality_NN."

In this study we focus on the nouns, being features of software, and adjectives as opinions about that aspect. For this purpose _NN and _NNS represent singular and plural nouns, with _JJ and _JJS labels for the adjectives in comparative and superlative degree, respectively.

3.3. Feature and sentiment extraction

For the feature based sentiment classification a database file is created containing the opinionated text, along with tagged phrase, potential feature, and sentiment of the opinion for the particular review sentence. Only those reviews are selected from the dataset in which any of the adoption features have been mentioned, to make an opinion. Features are usually expressed as nouns and so each noun is extracted from the sentence. Similarly, opinions are extracted on the basis of the nearest opinion words, i.e. adjective or adverb for nouns. This task is performed on the basis of a set of opinion words normally used to express opinions. The extracted adjective or adverb is compared with the bags of positive and negative words from the sentiwordnet 3.0 polarity database and considered to be positive or negative depending on its match. Similarly, for the sentiment strength score calculation we utilized the sentistrength opinion lexicon to assign a score to each tweet.

Consider the following sentences in which feature and appropriate sentiment are expressed in Table 3:

- *"Open Source Software Survey Results depict there are some awful issues with its functionality."*
- *"Trialability of open source software makes my life easier."*
- *"Open source software is maintained by a big community who provide outstanding community support usually."*

Table 3. POS tagged reviews.

awful_JJ issues_NNS with_IN its_PRP$ functionality_NN
Trialability_NNS of_IN OSS_NN makes_VBZ my_PRP$ life_NN easier_JJR
OSS_NN is_VBZ maintained_VBN by_IN a_DT big_JJ community_NN who_WP provide_VBP outstanding_JJ community_NN support_NN usually_RB

In the first sentence, the word "functionality" is just after the opinion word "awful". In the second example, the word "trialability" is the feature for which the word "easier" is used to express the sentiment. In the third example, the opinion word "big" is used to express sentiment regarding the "OSS community"; furthermore, the word "outstanding" is used to opinionate the word "community support". This approach is suitable for the explicitly mentioned features whereas in this paper implicit features are extracted manually. A sample transaction file containing feature and sentiment regarding different OSS products is presented in Table 4.

Table 4. Product, feature, and sentiment file.

Product	Feature	Sentiment
OSS Adoption	cost saving	positive
OSS Adoption	Functionality	negative
OSS Adoption	scalability	positive
OSS Adoption	documentation	negative
OSS Adoption	trialability	Positive
OSS Adoption	compatibility	positive

Only those phrases are extracted that contain an adjective, adverb, verb, and noun. N-gram modeling (n number of consecutive words) is applied to the tagged data to extract context around individual words that stay in a text sequentially. Therefore, from the tagged review only the two or three consecutive words that adhere to one of the patterns generated are extracted. Valid rules extracted by tri-gram modeling are stored in a file with .arff extension, which is used by WEKA for further processing. Some of the extracted phrase patterns along with some examples are presented in Table 5 for further processing.

Table 5. Extracted opinionated phrase patterns and examples.

Pattern	1st Word	2nd Word	3rd Word	Examples
Pattern I	JJ	NN/NNS	--	(free trialability), (good functionality), (valuable feature), (great configurability) and so forth
Pattern II	VBZ	JJ	NN/NNS	(offers good functionality)
Pattern III	JJ	NN/NNS	NN/NNS	(outstanding community support), (open source software) & so forth
Pattern IV	VBZ	JJ	--	(looks interesting), (hates security), (is great) and so forth
Patter V	RB/ RBR/RBS	JJ/RB/ RBR/RBS	NN/NNS	(Severely dysfunctional software), (hugely awesome piece), (hardly relevant OSS) and so forth

3.4. Sentiment classification

The next phase in the proposed model is the sentiment classification, which is based on a supervised machine learning approach. For the aspect level sentiment classification and model evaluation we selected Apriori, GSP, and AdaBoost to apply to the review dataset. The reviews classified from the previous step are used as a training dataset for these classifiers using the machine learning WEKA tool. Then the best extracted noun and adjective rule combinations from these techniques are applied to the testing data to check whether the rules are applicable or not. The repeated rules as well as those rules whose support count is less than the lower bound min support, which is 0.04 in our case, are eliminated. Then we applied AdaBoost. On our training file, we performed experiments with different classifiers of Decision Trees with AdaBoost, which were ID3, ADTree and NBTree, REP Tree, Random Tree, J48 etc. Our final model is made with ID3.

3.5. Sentiment summarization

The summarization process is carried out at both the product and feature level. A feature and product level summary of OSS products is created, after determining the sentiment of every sentence with frequency count and Bayesian probability. To determine the Bayesian probability, the system uses a statistical opinion analyzer (SOA). After the calculation of the frequency count, Bayesian probability is calculated, which intends to give accurate and reliable predictions. The summarized information is presented in both tabular and visual form, with which customers get a quick insight into the product quality, and other features. A sum of positive and negative opinions and an average of precision and recall results are calculated.

4. Dataset and evaluation measures

This section describes the experimental setup and results through the application of the proposed methodology including the corpus, processing, and the results.

4.1. Corpus

To evaluate our system we collected user tweets and reviews on various OSS products and generic comments about OSS adoption. The reviews are collected from a public timeline straddling from February 2014 to July 2014 with a total of 25,485 reviews. We collected them from Twitter and different OSS forums and blogs. The dataset was initially filtered according to the following criteria: 1) relevant key words associated with OSS aspect must be present in the text, 2) duplicate text is also ignored 3), non-English reviews are also sieved out to avoid complications in analyzing multilingual tweets. The unique characteristics of tweet messages are they are short in length and have a lot of junk words, which make the sentiment analysis more difficult. The reviews are gathered from 22,723 developers with an average of 1.12 reviews per developer. This dataset contains reviews regarding five different OSS products and generic comments about the open source paradigm. From this data set we annotated, selected, and extracted 1992 reviews referring to different OSS adoption factors. On average, there are 332 reviews for each open source resource. Further the polarity of the tweets is detected, to assign them a positive or negative score manually with respect to a particular adoption aspect. As depicted in Table 6, we used a total of 1992 reviews containing 1381 positive and 674 negative reviews.

Table 6. Number and percentage of comments.

OSS Adoption	25,485	664	0.33%
Software I		352	0.18%
Software II		331	0.17%
Software III		293	0.15%
Software IV		186	0.09%
Software V		166	0.08%
Total		1992	

The main motivation behind using this dataset is to identify how representative the social media is about the OSS technology adoption. The experimental results in Table 7 show that the people in general discuss various aspects of OSS adoption as a whole. By investigating the adoption of specific software, OpenOffice and PostgreSQL are the software that people are talking about the most, followed by MySQL, GNU Linux, and Apache. The contents of posts are represented by a set of key words to express the positive and negative points about different products. These sentiment words are shown in Table 7 together with a tag cloud diagram shown in Figure 2 summarizing sentiment words for OSS product reviews. The size of the revealed term in the tag cloud diagram represents the number of its mentions in the tweets to depict its importance.

Table 7. Sentiment words in reviews.

Positive sentiment	Negative sentiment
Free, open, great, good, simple, awesome, interesting, worth, emotional, professional, fine, wonderful, favorite, valuable, secure, reliable, competitive, creative, compatible, impressive, easy	Ridiculous, weird, terrible, undeserved, bad, wrong, unreliable, dumb, rabid, confusing, overbearing, odd, marginalized, bloody, discriminatory
Neutral sentiment	**Undefined sentiment**
Other, cant, overall, own, official, else, many	Nigerian, statistical, recursive, causal, medical, marginalized, key, African

Figure 2. Tag cloud diagram for sentiment words.

4.2. Evaluation metrics

In order to evaluate the accuracy and performance of our proposed model we computed the standard performance metrics of accuracy, precision, and recall, which are commonly used in sentiment analysis. The trade-off between the precision, recall, and accuracy depends on their systematic application. For the OSS adoption corpus, these evaluation metrics are computed by using the formulas given below.

Accuracy is the percentage of the test set correctly classified to the total number of instances and is measured with the following formula:

$$\text{Accuracy} = \frac{\text{Number of Instances Correctly Classified}}{\text{Total Number of Instances}} \tag{1}$$

The percentage of predicted instances that are correctly classified is referred to as precision and is also called the positive predicated value. This scenario can be defined for a binary classification problem the judgment within a contingency table, as shown in Table 8.

Table 8. Confusion metrics.

		Human judgment	
		Yes	No
System judgment	Yes	TP (A)	FN (B)
	No	FP (C)	TN (D)

For example, if the number of instances correctly classified in a class, also called true positive, is labeled as (A) and number of instances incorrectly classified in a class, also called false negative, is labeled as (B) then precision can be measured with the following formula:

$$Precision = \frac{A}{(A + B)} * 100 \tag{2}$$

Recall, also known as sensitivity, is calculated as the number of instances correctly classified in a class, also called true positive, labeled as (A) divided by the sum of the number of instances correctly classified in a class and the number of instance that belong to this class but incorrectly classified to the other class, also called false positive, labeled as (C) in this scenario. Recall can be measured with the following formula:

$$Recall = \frac{A}{(A + C)} * 100 \tag{3}$$

5. Results and discussion

Our results are divided into the following main parts: 1) the percentage of positive and negative reviews of each OSS product, 2) OSS product-wise sentiment summarization, 3) the classification accuracy of the reviews. Tables 9 through 16 show the results obtained for the mentioned parts separately. We have analyzed 1992 tweets, of which 1381 are positive, whereas 672 are negative. It is evident from Table 9 that the frequency of positive sentiment outstrips the frequency of negative sentiments at general OSS adoption level and for each individual software used in this study. The grouped reviews are shown in Table 10.

Table 9. Software-wise opinion frequency distribution.

	Frequency of positive opinions	Frequency of negative opinions
OSS adoption	477	164
Software I	245	152
Software II	228	114
Software III	122	147
Software IV	176	42
Software V	133	55
Total	1381	674

Table 10. Software-wise Bayesian probability distribution.

	Probability of positive opinions	Probability of negative opinions
OSS adoption	0.7441	0.2559
Software I	0.6172	0.3828
Software II	0.6667	0.3333
Software III	0.4535	0.5465
Software IV	0.8073	0.1927
Software V	0.7074	0.2926
Total	0.6720	0.3280

We have calculated the feature-wise as well as product-wise opinion frequency and Bayesian probability distribution. Tables 9 and 10 show the frequency and Bayesian probability of positive and negative clauses of various OSS products, whereas Table 11 depicts the frequency and Bayesian probability to represent the sentiment polarity of each feature.

Analysis of the results illustrates that functionality, cost saving, reliability, and community support are the most commonly discussed features. There are few instances in connection with the government adoption and configurability of OSS. However, the analysis of the results shows that most of the instances are positive in relation to government adoption. The probability or likelihood to submit positive opinions is greater as compared to negative comments, either at the product or at the feature level.

Visualization is great way of exploiting information in the form of opinions or sentiment expressed in relation to various aspects by a sizably voluminous group of users in any domain. Figure 3 shows the opinion frequency graph of different adoption factors. The feature level sentiment profile for the positively and negatively opinionated software is visualized in Figures 4 and 5. The Bayesian probability distribution of different adoption factors is described in Figure 6.

Table 11. Feature-wise opinion frequency and Bayesian probability distribution.

Aspects	Polarity	OSS adoption	P1	P2	P3	P4	P5	Total
FUNC	POS	57	49	47	29	43	25	250
	NEG	19	37	25	32	3	9	125
	P. probability	0.1637	0.240	0.2540	0.308	0.2905	0.2380	
	N. probability	0.1743	0.0345	0.3012	0.2831	0.1111	0.225	
CS_SAV	POS	43	37	10	5	12	–	107
	NEG	13	3	2	1	2	–	21
	P. probability	0.1235	0.181	0.0540	0.0531	0.0818	–	
	N. probability	0.1992	0.0280	0.0240	0.0088	0.074	–	
REL	POS	39	18	21	10	16	15	119
	NEG	13	14	1	9	3	5	45
	P. probability	0.1120	0.882	0.1135	0.1063	0.108	0.142	
	N. probability	0.119	0.1308	0.0120	0.0796	0.111	0.125	
GOV_ADP	POS	25	1	1	–	2	–	29
	NEG	1	0	0	–	0	–	1
	P. probability	0.0718	0.0049	0.0054	–	0.0135	–	
	N. probability	0.009	0.0000	0.0000	–	0	–	
COM_SUP	POS	41	15	14	13	13	9	105
	NEG	8	9	13	14	6	11	61
	P. probability	0.117	0.0735	0.0756	0.1382	0.0878	0.0857	
	N. probability	0.733	0.0841	0.1566	0.1238	0.2222	0.275	
TRIAL	POS	27	24	10	3	7	5	76
	NEG	4	14	5	7	2	0	32
	P. probability	0.77	0.1176	0.0540	0.0319	0.0472	0.047	
	N. probability	0.366	0.1308	0.0602	0.0619	0.0740	0	
COMP	POS	29	25	10	7	6	10	87
	NEG	7	14	6	9	3	4	43
	P. probability	0.080	0.1225	0.0540	0.0744	0.0405	0.095	
	N. probability	0.642	0.1308	0.0722	0.0796	0.1111	0.1	
SCAL	POS	15	5	11	8	16	6	61
	NEG	10	5	5	13	1	2	36
	P. probability	0.431	0.0245	0.0594	0.851	0.1081	0.057	
	N. probability	0.091	0.0467	0.0602	0.1150	0.0370	0.05	
USAB	POS	24	12	21	6	9	6	78
	NEG	4	5	3	3	4	0	19
	P. probability	0.689	0.0588	0.1135	0.063	0.0608	0.0571	
	N. probability	0.366	0.0467	0.0361	0.0265	0.1481	0.0000	
SECU	POS	19	3	16	3	11	13	65
	NEG	20	0	6	14	1	5	46
	P. probability	0.0545	0.0147	0.0864	0.0319	0.0740	0.1238	
	N. probability	0.1834	0	0.0722	0.1238	0.0370	0.125	
CONF	POS	8	1	13	4	6	5	37
	NEG	4	3	6	3	0	2	18
	P. probability	0.0229	0.0490	0.0702	0.042	0.040	0.0476	
	N. probability	0.03669	0.0280	0.0722	0.0265	0	0.05	
DOC	POS	22	14	11	6	7	11	71
	NEG	6	3	11	8	2	2	32
	P. probability	0.0632	0.0686	0.0594	0.0638	0.0472	0.1047	
	N. probability	0.0556	0.0280	0.1325	0.0707	0.0740	0.05	

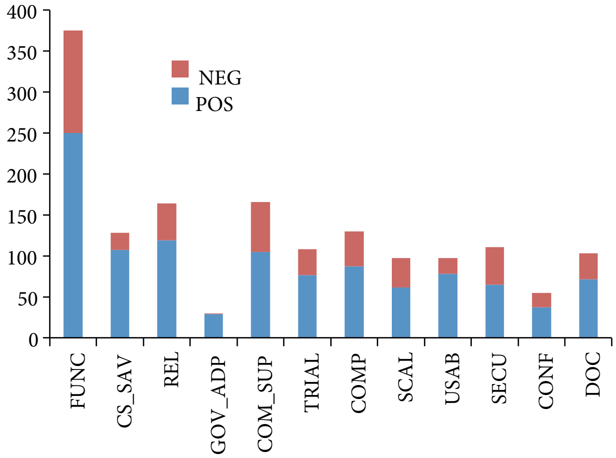

Figure 3. Opinion frequency graph of adoption factors.

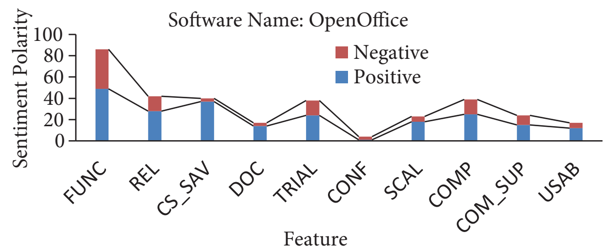

Figure 4. Sentiment profile of positively opinionated software.

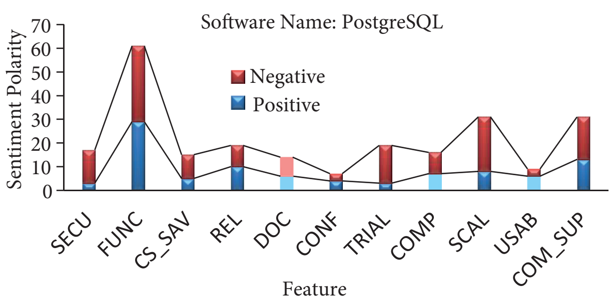

Figure 5. Sentiment profile of negatively opinionated software.

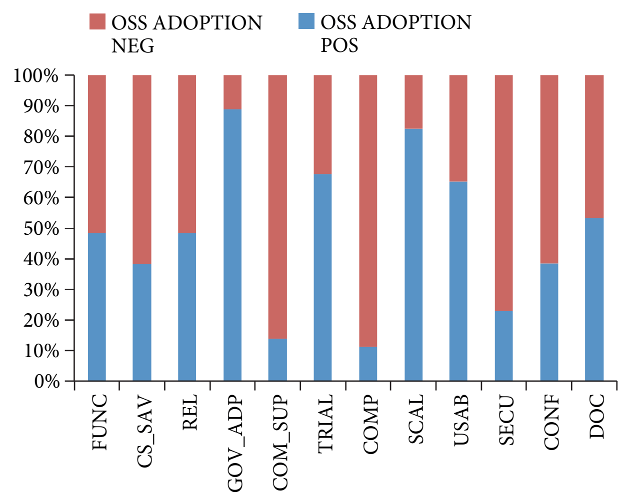

Figure 6. Bayesian probability distribution graph of adoption factors.

After the extraction of features and sentiments from tweets, we calculated the sentiment strength of the tweets. The positive sentiment strength ranges from 1 (not positive) to 5 (extremely positive) and negative sentiment strength ranges from –1 (not negative) to –5 (extremely negative). The results of sentiment strength analysis of OSS adoption and for each OSS product are represented in Table 12 along with the frequency of tweets and sentiment strength scores. It can be seen from the results that there are more positive tweets as compared to negative tweets, with an approximately average positive sentiment score of 41.17% and negative sentiment score of 28.05%, the remaining being neutral. The sentiment score plots are presented in Figure 7.

We then evaluated our results based on different classifiers. Table 13 shows the values of the evaluation metrics calculated by applying the classifiers to the test data set in predicting the new instances at implementation level.

Precision, recall, and weighted average of the dataset for each OSS product are calculated using training datasets; later we tested the results on the basis of test datasets. In this section, the results for each OSS product are presented individually in Tables 13 and 14 and Figures 8 and 9. We applied four different classifiers, namely

GSP, Apriori, AdaBoost, and Bayesian, for the extraction of opinion words from the OSS adoption dataset. The result shows that AdaBoost is more efficient by scoring (precision 90.3% and recall 91.8%) as compared to the other techniques such as Bayesian (precision 89.3% and recall of 91%), Apriori (precision 84.1% and recall of 85.6%), and GSP (precision 79.5% and recall 81.1%).

Table 12. Sentiment score, polarity, and frequency distribution.

		Negative					Neutral	Positive				
	Sentiment score	–5	–4	–3	–2	–1	0	1	2	3	4	5
OSS Adoption	Frequency	1	11	55	75	373	61	261	135	91	15	1
Software I	Frequency	1	8	35	73	173	62	147	79	49	13	2
Software II	Frequency	1	10	28	67	182	43	151	75	43	19	0
Software III	Frequency	1	11	21	65	145	50	142	52	39	9	1
Software IV	Frequency	0	4	11	24	117	30	58	57	34	6	1
Software V	Frequency	1	1	13	24	103	24	74	33	31	3	1

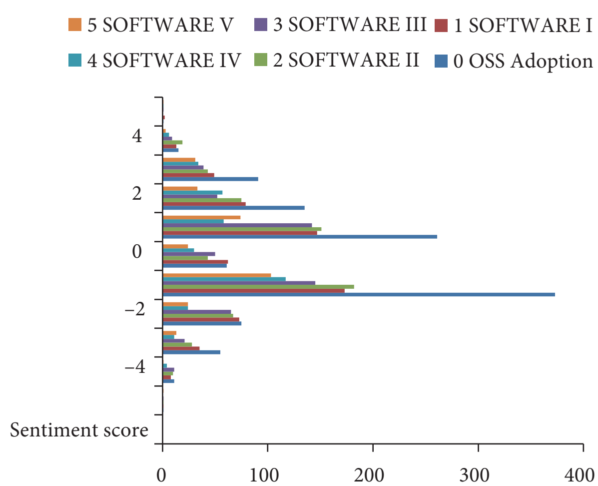

Figure 7. Sentiment score distribution of OSS adoption.

Table 13. Precision and recall for frequent feature extraction using AdaBoost and Apriori.

Dataset	AdaBoost		Apriori	
	Precision	Recall	Precision	Recall
OSS adoption	0.86	0.88	0.80	0.81
Software I	0.94	0.95	0.89	0.90
Software II	0.90	0.93	0.83	0.85
Software III	0.95	0.95	0.88	0.88
Software IV	0.88	0.90	0.80	0.83
Software V	0.89	0.90	0.85	0.87
Average	0.903	0.918	0.841	0.856

Table 14. Precision and recall for frequent feature extraction using GSP and Bayesian.

Dataset	GSP		Bayesian	
	Precision	Recall	Precision	Recall
OSS Adoption	0.80	0.83	0.94	0.96
Software I	0.78	0.79	0.90	0.92
Software II	0.84	0.86	0.89	0.90
Software III	0.75	0.78	0.92	0.90
Software IV	0.78	0.81	0.87	0.90
Software V	0.82	0.80	0.84	0.88
Average	0.795	0.811	0.893	0.91

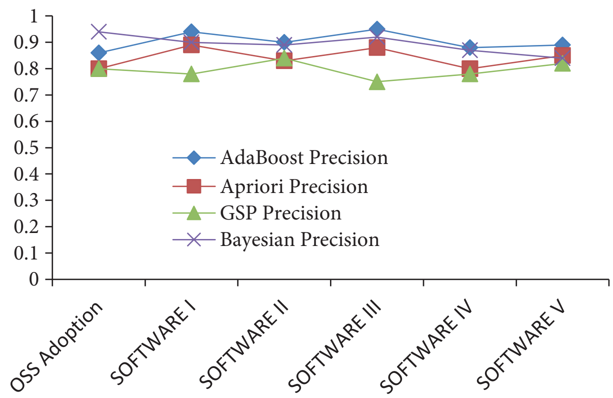

Figure 8. Comparison of precision of different techniques.

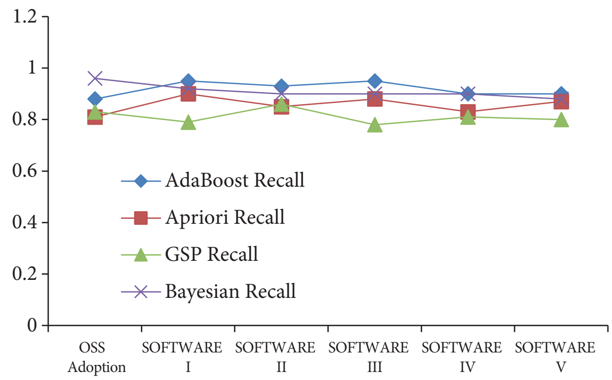

Figure 9. Comparison of recall of different techniques.

Furthermore, we calculated the accuracy for the sentiment analysis. From 25,485 reviews, we randomly selected 1992 reviews in which any of the adoption factors have been discussed, and calculated the correctness by using the accuracy metric given above. The experimental result of the OSS product review dataset shows that the accuracy of the classifiers applied is within the acceptable range. An evaluation of four different classifiers: AdaBoost, Bayesian, GSP, and Apriori, is conducted. A comparative analysis of different classification techniques on the OSS product review dataset is shown in Tables 15 and 16 using WEKA. The average accuracy of AdaBoost and Bayesian (73.45% and 72.67%) is higher as compared to Apriori and GSP (69.67% and 68.13%, respectively); the representation is shown in Figure 10.

Table 15. Accuracy comparison of classifiers used.

Classifiers	Total reviews	Positive reviews	Negative reviews	Neutral/undefined reviews	Accuracy
AdaBoost	1992	1463	398	131	73.45
Bayesian	1992	1448	401	143	72.67
Apriori	1992	1388	396	208	69.67
GSP	1992	1357	396	239	68.13

6. Conclusion

Developers submit their feedback and reviews regarding different software products through blogs, forums, and open source communities. These comments serve as a quality reference for new developers to adopt a new

Table 16. Accuracy of different techniques

Dataset	AdaBoost accuracy%	Apriori accuracy%	GSP accuracy%	Bayesian accuracy%
OSS Adoption	76.9%	80%	74.46%	81%
Software I	60%	58%	57%	72%
Software II	78.3%	65%	66.2%	67%
Software III	75%	74%	75.4%	78%
Software IV	73.5%	71%	68.11%	70%
Software V	77%	70%	67.63%	68%
Average	73.45%	69.67%	68.13%	72.67%

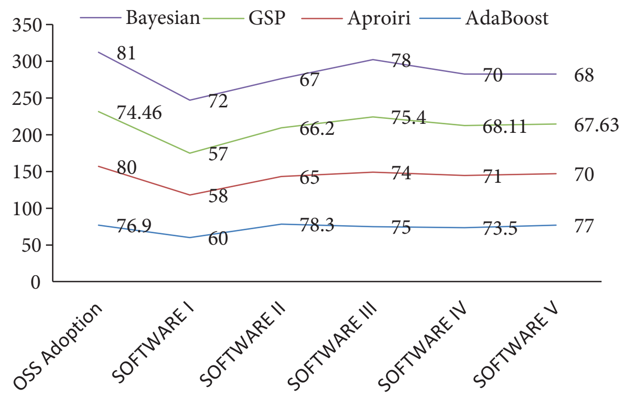

Figure 10. Comparison of accuracy of different techniques.

technology. In this paper, to evaluate OSS adoption we have proposed a model applying different opinion mining techniques. This model is constructed on different OSS adoption factors like functionality, cost saving, quality characteristics, and easy adoption. First the reviews are classified by using the lexicon based technique, i.e. sentiwordnet, to compute the sentiment for each software product. Then the sentiment strength of tweets is calculated using the setistrength opinion lexicon. Three different machine learning techniques, Apriori, GSP, and AdaBoost, are applied to the prepared data to find the most significant algorithm in extracting the nouns and adjectives. In experiments, we applied the model for OSS adoption reviews containing 1381 positive and 674 negative reviews. Our system achieved an accuracy of about 70.98%. Further, the experimental results depict that our model is effective in performing the task, having an acceptable score for precision and recall, above 80%. It can rightly be concluded that the application of ML techniques from sentiment classification is quite successful.

7. Future work

The proposed approach can easily be extended to comments and opinions available on various digital social forums. Future work can be continued in the following directions:

- To conduct automatic analysis and evaluation of OSS opinionated feature evaluation; the methods investigated in this research can be implemented and integrated with a search.
- An extensive study could be carried out by applying different visualization and summarization techniques.
- In addition, a comparison of the results produced by sentiment analysis can be made with a survey of targeted users to assess user satisfaction regarding the different features of OSS products.
- Conducting intrusive analysis of the psyche of a person writing a blog.

Lastly, it is worthwhile extending the current research to other related areas, i.e. travel blogs, product reviews, and hotel reviews; thus it has a tremendous scope for practical applications.

References

[1] Glynn E, Fitzgerald B, Exton C. Commercial adoption of open source software: an empirical study. IEEE Proc of ISESE; 17–18 November 2005; Noosa Heads, Australia; 225-234.

[2] Dehinbo K, Pretorius P, Dehinbo J. Strategic analysis towards deriving competitive advantage with the use of FOSS: the case of a South African university. IEEE Proc of ITNG 2012; 335-342.

[3] Mindel JL, Mui L. Verma S. Open source software adoption in ASEAN member countries. IEEE Proc of HICSS-40; 3–6 January 2007; Waikoloa, Big Island, Hawaii, USA; 226-235.

[4] Liu C, Zou Y, Cai S, Xie B, Mei H. Finding the merits and drawbacks of software resources from comments. IEEE Proc of ASE 2011; 432-435.

[5] El-Halees AM. Software usability evaluation using opinion mining. J Softw 2014; 9: 343-349.

[6] Kaur A, Gupta V. A survey on sentiment analysis and opinion mining techniques. J Emerging Technol Web Intell 2013; 5: 367-371.

[7] Anwer N, Rashid A, Hassan S. Feature based opinion mining of online free format customer reviews using frequency distribution and Bayesian statistics. IEEE Proc of NCM; August 2010; 57-62.

[8] Wu Y, Wei F, Liu S, Au N, Cui W, Zhou H, Qu H. OpinionSeer: interactive visualization of hotel customer feedback. IEEE T Vis Comput Graph 2010; 16: 1109-1118.

[9] Singh VK, Piryani R, Uddin A, Waila P. Sentiment analysis of movie reviews: a new feature-based heuristic for aspect-level sentiment classification. IEEE iMac4s 2013; 712-717.

[10] Rashid A, Asif S, Butt NA, Ashraf I (2013). Feature level opinion mining of educational student feedback data using sequential pattern mining and association rule mining. Int J Computer Appl 2013; 81: 31-38.

[11] Ma B, Zhang D, Yan Z, Kim T. An lda and synonym lexicon based approach to product feature extraction from online consumer product reviews. J Electron Comm Res 2013; 14: 304-314.

[12] Wang C, Xiao Z, Liu Y, Xu Y, Zhou A, Zhang K. SentiView: sentiment analysis and visualization for internet popular topics. IEEE T Hum Mach Syst 2013; 43: 620-630.

[13] Wang D, Zhu S, Li T. SumView: A web-based engine for summarizing product reviews and customer opinions. Expert Syst Appl 2013; 40: 27-33.

[14] Hauge Ø, Ayala C, Conradi R. (2010). Adoption of open source software in software-intensive organizations – a systematic literature review. Inf Softw Technol 2010; 52: 1133-1154.

[15] Karume SM. Mbugua S. Trends in adoption of open source software in Africa. Journal of Emerging Trends in Computing and Information Sciences 2012; 311: 1509-1515.

[16] Atoum I, Bong CH. A framework to predict software quality in use from Software Reviews. Proc of DaEng-2013; January 2014; Springer Singapore; 429-436.

[17] Jiang W, Ruan H, Zhang L, Lew P, Jiang J. For user-driven software evolution: requirements elicitation derived from mining online reviews. Adv Knowl Disc Data Min; Springer International Publishing 2014; 584-595.

[18] Zou Y, Liu C, Jin Y, Xie B. Assessing software quality through web comment search and analysis. Safe and Secure Software Reuse; Springer 2013; 208-223.

[19] Jamroonsilp S, Prompoon N. Analyzing software reviews for software quality-based ranking. IEEE Proc of ECTI-CON'13; May 2013; 1-6.

[20] Galster M, Tofan D. Exploring possibilities to analyse microblogs for dependability information in variability-intensive open source software systems. IEEE Proc of ISSREW'3; 4–7 November 2013; 321-325.

[21] Dehkharghani R, Yilmaz C. Automatically identifying a software product's quality attributes through sentiment analysis of tweets. IEEE Proc of NaturaLiSE'13; 25 May 2013; San Francisco, CA, USA; 25-30.

[22] Leopairote W, Surarerks A, Prompoon N. Evaluating software quality in use using user reviews mining. IEEE Proc of JCSSE'13; 30–31 May 2013; 257-262.