Indian Journal of Science and Technology, Vol 8(13), DOI: 10.17485/ijst/2015/v8i13/61464, July 2015
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Building a Custom Sentiment Analysis Tool based on an Ontology for Twitter Posts

K. Vithiya Ruba* and D. Venkatesan

School of Computing, SASTRA University, Tamil Nadu - 613 40, India; vitkrish13@gmail.com, venkatgowri@cse.sastra.edu

Abstract

Twitter is a popular micro-blogging platform which allows the users to share their opinion on any domain. The thoughts of the people vary according to the domain and also the opinion may contain both positive and negative words which are called as opinion words and are given in the form of dictionary called lexicon dictionary. The sentiment analysis done without feature extraction fails to give the deep result about the users opinion but in our proposed approach , features of the domain are extracted by building ontology which helps in getting the refined sentiment analysis. Feature based sentiment analysis gives the best result. While analyzing the sentiment, scores are assigned to the tweets so that the sentiment score of our tweets are compared with the third party like American Customer Satisfaction Index score. This comparison shows that our score assignment gives the detailed analysis of the features than the third party.

Keywords: Opinion Words, Ontology, Protégé, Sentiment, Sentiment score, Tweets, Twitter

1. Introduction

Micro-blogging initially gained less attention, but with the popularity of social networking sites like Face book, Twitter it gained more attention from the people. Twitter has about 500 million registered users and about 400 million messages per day. Nowadays, Commercial competitors are gaining advantage by analyzing the user's posts about them, about their products and competitors for improving the product features in future. The users share their domain of interest on their profile which can be viewed and commented by anyone. Here in the case of Twitter micro-blogging we call the comment as a retweet.

Sentiment or opinion mining[14] is nothing but analyzing whether the given input is positive, negative or neutral[1] in other words, we can say it as determining the polarity of the input. The input here can be any data source like Blogs, Review sites like Amazon reviews, Restaurant reviews and micro-blogging like Twitter. One of the main reasons for choosing twitter is in the case of review sites and blogs they would have discussed about only particular domain. If the users are interested in some other domain they have to search for some other review sites and blogs, but in case of Twitter which is the micro-blogging any domain can be discussed here[16]. So, the people can search their interested domain which can be anything like product reviews or political reviews and perform their sentiment analysis on that domain.

In addition to the domain, the particular domain features can be extracted by building ontology for the interested domain. For ex., consider the sample tweet S: "The battery of Lenovo laptop was wonderful, although the screen size was bad". Scoring of the tweets can be done in two ways, either qualitative or quantitative. Qualitative is nothing but identifying whether positive, negative or neutral and quantitative is overall scoring of the tweets. In this paper, we score the tweets individually and also return how many positive, negative and neutral tweets are there. Here, in this example the opinion words are wonderful and bad, based on the opinion words score the tweets. Opinion words are contained in the opinion lexicon dictionary which is very important and the domain is Laptop which can be easily identified from the tweet and the features are: battery and screen size. From this sample tweet we can say that we are building ontology for

*Author for correspondence

the domain of Laptop. The features of a domain can be queried through onto CAT, a package in R.

More recent works have been done in ontology deployment in the micro-blogging. In [8] proposed an approach for populating an existing earthquake evacuation ontology with an instances based on tweets and this information is related to ontology like evacuation centre name, what are the products they are offering and also about the timestamp of the tweets are analyzed, additionally in real time Google maps are used for extracting the centre address even though it is not included in tweet and later can be appended to the ontology.

A multistage system proposed by in [15] is used for analyzing the sentiment in tweet was proposed which uses 2 class and 4 class classification. Twisent addressed the problems like spam's, structural, Entity specification and Pragmatics. Twisent shows the accuracy improvement in 2-class as 14% and in 4-class setting 8.94% when compared with C-Fell-It, a rule based system similar to Twisent. Although the accuracy is achieved Twisent uses generic lexicon and so failed to capture the sarcasm or implicit sentiment.

Another work was proposed by Pak A, Paroubek P[12] where twitter is considered as a corpus to collect the positive, negative and objective texts, the linguistic analysis is performed on the collected corpora and also a sentiment classification system is built where the features can be extracted based on the dataset and it is used to train the classifier whereas n-grams, bigrams and trigrams are examined as a feature and n-grams outperformed the other two. Naïve Bayes classifier is used as a classification system on N-gram and on part-of-speech, in future multilingual corpus data can be collected and the obtained data can be used for building multilingual sentiment classifier.

In[4] proposed an ontology based sentiment analysis and they use Ontogen 2.0 to create an ontology and the object-attribute pair is retrieved using the Jena API and ConExp is used for the visualization of ontology. The tweets are retrieved using the Twitter4J API but instead of performing the sentiment analysis on their own effort they depended on third party Open-Dover for analyzing the sentiment so the methodology used by them is not known by us and also the perfection of the sentiment score is not good.

2. Description of the Proposed Methodology

As we have discussed above, the domain ontology is created which is used for scoring the notions in the tweet. A set of tweets is given as the input regarding a specific domain and for each feature in the tweets the sentiment score is assigned. Three steps are involved in the proposed methodology

(i) Domain ontology creation for the particular problem under consideration.
(ii) The tweets relevant to the features of the domain are extracted from twitter.
(iii) The sentiment analysis is done for the retrieved tweets.

Algorithm: Ontology Creation
Ont_input: concept (c)
Ont_var: T- Free set of tweets
P- Free set of objects
Q- Free set of attributes
1. T<- ret_tweets(s)
2. foreach t ∈ T do
3. p<- ret_obj (t); if p ≠ Null then
4. p:= P U {p}
5. Q'<- ret_attr (t)
6. foreach q ∈ Q'; if (p,q) ≠ Φ do
7. Q: = Q U {q}
8. Table<- (p, q)
9. return Table

Figure 1 describes the architecture for proposed methodology.

2.1 Domain Ontology Creation

Various methods are available for developing the ontology[10]. In this step, the objects, attributes and the relationship between them are identified for the domain under consideration. Many tools like TODE, Onto Gen 2.0 are available for the creation of the domain ontology, but in this paper, we use one of the popular tools called protégé 4.1[22] for the creation of the ontology.

The following are advantages of using ontologies:

(i) Ontology size is appropriate.
(ii) Ontology can be reused.
(iii) The domain knowledge and the operational knowledge are separated.

Various ontology languages like FOAF, RDF are available, but we make use of the OWL language for the ontology creation. Detective classifiers are included in protégé which validates the models to consistent and gather new information by analyzing the ontology.

Figure 2 represents ontology created for laptop domain.

After the creation of the ontology[17] the querying of the object-attribute pair is done in DL query or SPARQL but as we are using the implementation language as R so the onto CAT package[23] of R is used for querying which is a Java library depending on OWL API and Java package of R. This package was created to support basic operations in an ontology which also includes the traversing and searching the terms. The reason used is a Hermi T reasoner when an OWL file is given as an input Hermi T reasoner can identify the relationship between the classes which is the advantage of using a reasoner in ontoCAT and the super class/subclass is supported here as a parent/child relationship.

Why R language is chosen?

One of the most important advantages of R is the packages[24]. Data manipulation can be easily done in R and also the graphical packages are more powerful for visual analysis of data.

Formal concept analysis is used for the easy representation of the ontology and from the objects and attributes collection ontology can be easily derived[9,11,18]. In ontology we have concepts sometimes called as classes,

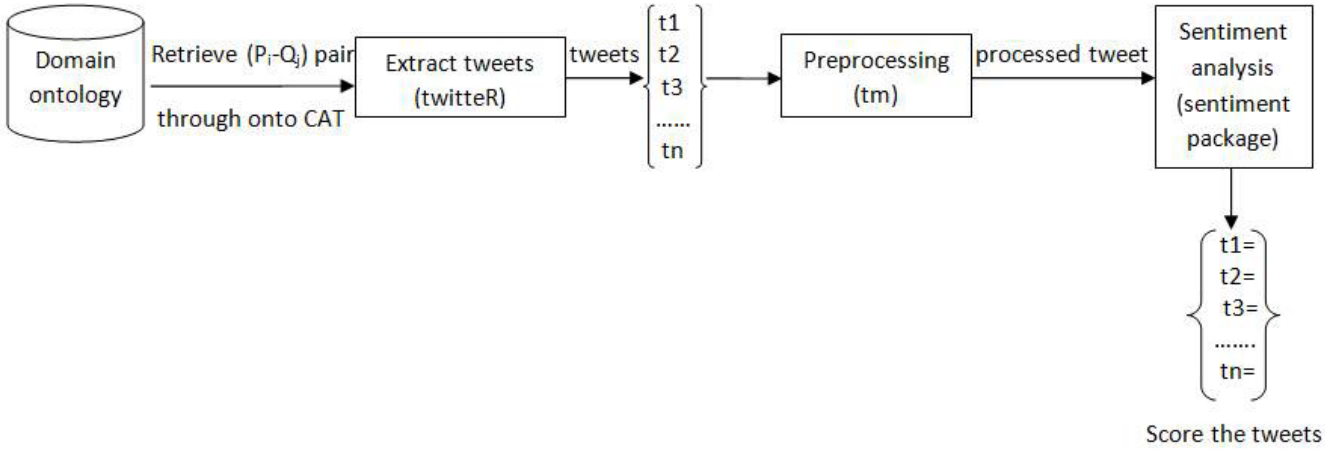

Figure 1. Architecture of the proposed approach.

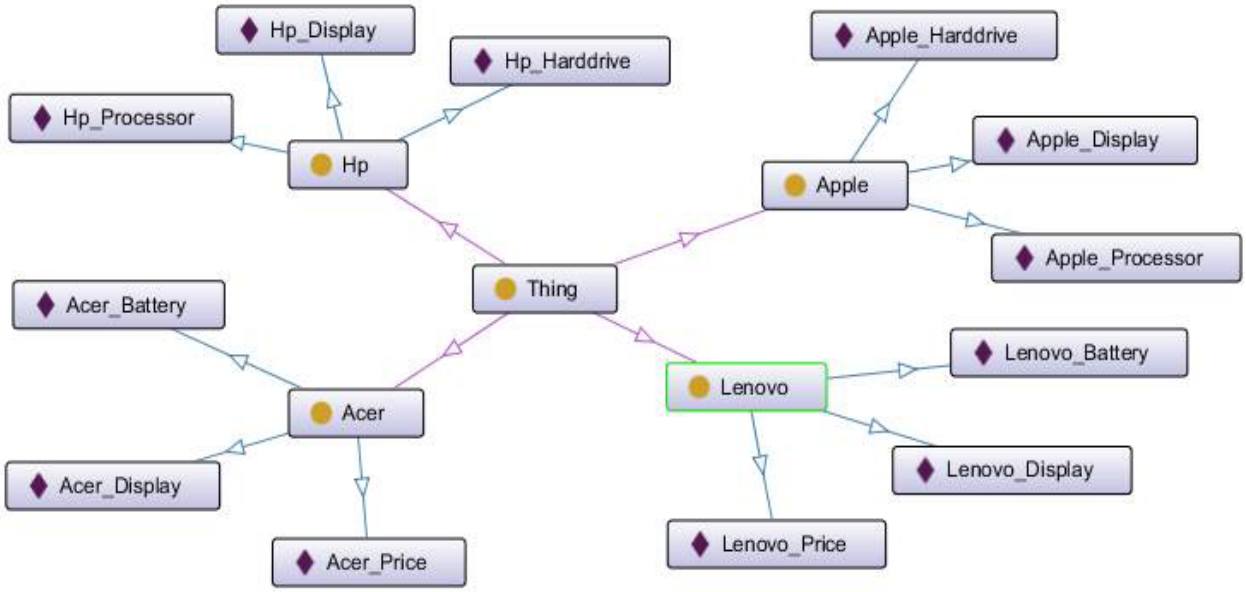

Figure 2. Ontology for laptop.

slots which are also known as roles or properties describe the various features and attributes of the concept.

Consider the concept as Laptop

C = {Laptop}
P = {set of objects}; {Lenovo, HP, Apple, Acer}
Q = {set of attributes}; {Display, Processor, Screen size, Hard drive}

Figure 3 represents the Object-Attribute relation for laptop.

In formal concept analysis the operator "'" is used as an object for derivation.

For a set of object P, P' can be defined as
P'= {all attributes in Q shared by the objects P}

For a set of attributes Q, Q' can be defined as
Q'= {all objects in P that have all the attributes of Q}

The pair sets (P, Q) of objects and attributes can be given as P'=Q and Q'=P which is called as formal concepts.

Generating the formal concept analysis

(i) A set of objects P is picked
(ii) Attributes of P, (i.e.) P' are derived
(iii) (P')' is derived
(iv) (P", P') is a formal concept analysis.

Figure 4 is the cross-table which represents the formal context. Rows represents the objects P and the columns represents the attributes Q, here we have a series of crosses called incidence which represents that each object P has attribute Q.

After the cross-table is represented, the graphical representation is done using concept lattices and visualized through Hasse diagram[3].

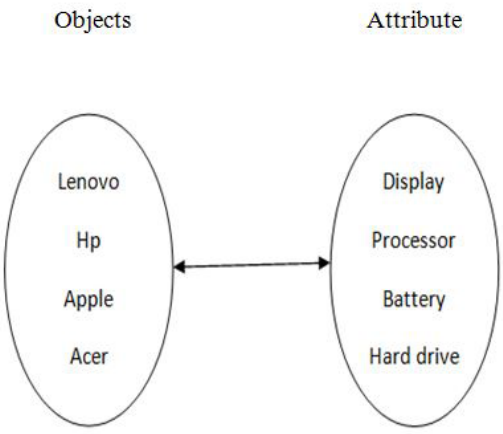

Figure 3. Object - Attribute relation.

Figure 5 is the lattice diagram; it can easily identify which object has the same attributes, which attributes shares the same object. Tracing up from a node is known as intension and tracing down from a node is called as an extension.

2.2 Extraction of Tweets based on the Features Identified

Before attempting to retrieve the tweets, the application has to be created in Twitter[19] which allows the users to search the tweets. After an application is created, it provides a user with consumer key and consumer secret key, access token, access secret token which can be used in the implementation for authentication of the application. twitteR package of R is a twitter REST API used for retrieving the tweets based on the given set of keywords. It is a R based twitter client which acts as an interface to the Twitter web API. The search term function of twitteR helps in retrieving the tweets. The package parameters are the search term, the longitude and latitude, and date for indicating the retrieval of tweets from that particular date. We can pass the search term as "lenovo+battery", the tweets that contain "Lenovo" and "battery" are extracted and also the tweets relevant to the Lenovo's battery are retrieved. Nearly 250 tweets are

A	B	C	D	E	F
	Display	Processor	Battery	Harddrive	Price
Lenovo	X		X		X
Hp	X	X		X	
Apple	X	X		X	
Acer	X		X		X

Figure 4. Laptop ontology.

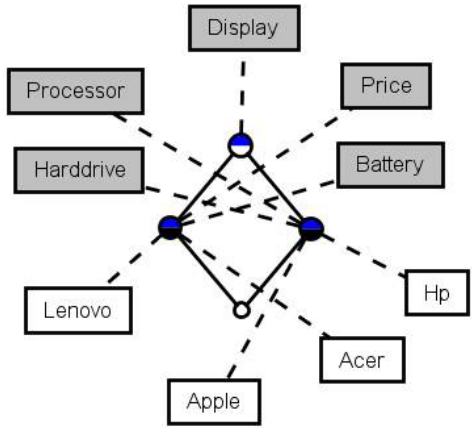

Figure 5. Hasse diagram visualization for laptop.

retrieved for every feature of the laptop. In addition to extraction it is mandatory to provide an interface for the OAuth 1.0 specification, which allows the users to authenticate via OAuth. It is supported in R through ROAuth package, setup_twitter_oauth () function of R helps in creating a connection to Twitter's search API. If successful a message is displayed. Now the tweets are saved in the data frame before we go for pre-processing step.

2.3 Sentiment Analysis

Many points have been discussed regarding the importance of sentiment analysis so, now we will discuss briefly about the techniques involved in the sentiment analysis, it can be done in two methods (i) Machine Learning and (ii) Semantic orientation. Usually natural language processing is used for text mining. Machine learning includes the supervised and text classification methods, The techniques used in machine learning are Naïve Bayes[2], Maximum entropy, SVM and Naïve Bayes is the most effective algorithm for the document classification, when machine language is used in the natural language processing techniques like K-nearest[5], Centroid classifier and N-gram model can be used.

When Semantic orientation is used[7], which is the unsupervised learning technique, determines how far a word is inclined towards positive and negative. The techniques used are the k-means clustering algorithm, hierarchical clustering and Association rule mining. Using semantic orientation (TF-IDF) Term frequency-Inverse document frequency weighting is found in the raw data. When semantic orientation is used in the real time application, it gives efficient result.

Steps Involved in Sentiment Analysis

(i) The retrieved data are cleaned and the document-term matrix is built which is also called as data preprocessing.
(ii) Lemmatization and the tokenization of tweets are done using packages.
(iii) Sentiment package of R is used for classifying the polarity of the tweets and it uses the inbuilt emotion dataset for approximately classifying the emotions into six categories like anger, joy, disgust, fear, sadness and surprise. The opinion lexicon dictionary contains the positive opinion words list and negative opinion words list for identifying the opinion words in a sentence and scores it accordingly.
(iv) After the sentiment is analyzed for each feature the histogram is generated for easy visualization of the sentiment score and also four csv files are created which holds the opinion, scores, plot and stack.
(v) The frequent words in the tweets are found.
(vi) And finally a word cloud is created for the visualization of the important words.

2.3.1 Data Cleaning

The data cleaning can be done easily using the text mining package of R. Data cleaning is nothing but the preprocessing of data which is done to reduce the noisy data, the following items should be removed from the retrieved tweets which will not be fit into the sentiment phase.

(i) Remove the punctuations, numbers, '@' symbol which is used while replying to the tweets of other users'.
(ii) Remove URLs (the tweets starting with http ://).
(iii) Remove only the '#' hash tag from the sentence, where the rest of the tweet is considered as a legible word and can be added to the collection of cleaned sentences for the better understanding of the tweet.

Term Document Matrix is created with the help of cleaned tweets which gives the frequent terms that occur in the collection of tweets. The matrix is created with the rows representing the documents and the columns showing the terms. In R language, text mining package is used for doing all the preprocessing techniques like removing numbers, punctuation, sparse terms, stop words and also the term document matrix is created with the help of this package only.

2.3.2 Tokenization, Lemmatization

Before tokenization and lemmatization the duplicate tweets are eliminated in order to avoid the confusion and to improve the efficiency, tokenization is done for the tweets, it is nothing but breaking down the text into meaningful elements called "tokens" which is given as the input for further processing, open NLP is used for processing tokenization. Lemmatization is reducing the inflected words to a base form or to a single meaningful item; here the lemmatization is done with the help of word net database. For e.g. the words like "walk" "walked" "walks" "walking" are reduced to a single form called "walk" which helps in easy identification of words.

2.3.3 Sentiment Analysis on Tweets

After preprocessing phase is over, the tweets are given to the sentiment analysis phase. For each feature the score is assigned based on the opinion words and the 'sentiment' package of R determines not only the polarity but also gives out the emotion results, it classifies the emotion into six different categories like: anger, disgust, fear, joy, sadness and surprise. In addition the BEST_FIT gives the most likely category that suits for the given sentence. The sentiment package includes the Naïve Bayes classifiers for identifying the classification. For emotion, a Naïve Bayes classifier is trained on Carlo Strapparava and Alessandro Valitutti's emotion lexicon[12] and for polarity it is trained on Janyce Wiebe's subjectivity lexicon[13,14] Classify_emotion () function takes the parameters like text columns which indicates the data frame, algorithm specifies the Naïve Bayes, prior field indicates the numeric probability for using the classifier. Classify_polarity () function takes the parameters similar to Classify_emotion () but additionally it specifies pweak field which is a numeric probability for specifying a weakly subjective term appears in the given text and pstrong field indicates a strongly subjective term. The sentiment package also depends on the NLP package and the RStem package of R for processing. The opinion lexicon dictionary contains the list of positive and negative words, now we can split the sentence array which contains the list of tweets into separate words and then comparison of the words in the dictionary is done with the list containing the separate words using the match () function which returns the position of the matched term and then scores the words.

2.3.4 Creating Histogram

After the sentiment analysis is done with the help of opinion lexicon dictionary and a sentiment package of R, the histogram is generated which indicates the overall positive, negative and neutral scores for the tweets, in addition to that the mean value is calculated from the detected score to indicate whether the overall score of the tweets is positive or negative. The table form of the scores is also indicated. The histogram compares the features of each product and shows which product feature has a highest positive score and which product feature has more negative score and also gives the clear analysis of which product feature is better than other products. After this four different csv file (comma separated files) are created which are prefixed with the search term, for e.g. the search term is "apple+battery" then the files are named as apple battery_opinion containing the opinion about the tweets, applebattery_scores contains the scores for each individual tweets, applebattery_plot shows the visual representation of scores and applebattery_stack gives the information about retweets, user id, longitude, latitude information, screen name and also about the status source.

Figure 6 shows the comparison of Apple laptop's one particular feature battery; similarly for all other features the histogram can be generated.

2.3.5 Frequency of the Words

Term frequency is the ratio of how many times a word occurred in the tweets to the total number of words in the tweets and it is calculated as a normalized frequency. The weighting of the terms is done in order to identify the most important term in the tweet which is also the term the users have discussed frequently on twitter, weighting scheme like tf-idf (term frequency-inverse document frequency) is often used in text mining as a weighting factor.

Considering a term frequency $tf_{a,b}$ it shows the number of times the term t_a occurs in the document d_b and idf_a is the ratio of total number of document to the number of documents where the term t_a occurs and finally by multiplying the values of tf and idf (tf.idf) tf-idf is calculated.

R supports the function WeightTfIdf (), where the document term matrix which contains the cleaned

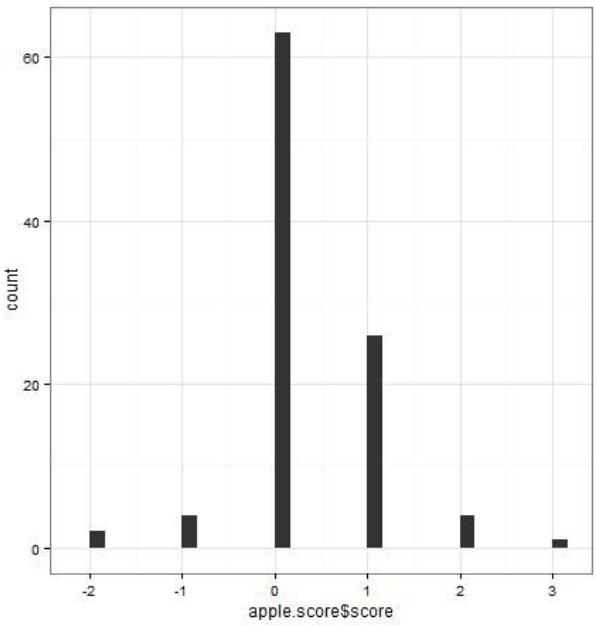

Figure 6. Histogram scores for apple+battery.

corpus is passed as a parameter and resultant is ranking of important terms in the tweet. After the frequency is calculated we will have to know the association between the terms in tweets, this lets us know how one term is related to the other term, and also helps in the identification of term relation and how the users have related the terms, findAssoc () method is supported in R to identify the association between the document term matrix, query term and a correlation limit ranging from 0 to 1 passed as parameters.

2.3.6 Word Cloud

A word cloud is used for the visually representing the text data which typically depicts the keyword or the tags used more frequently by the users. The maximum number of tagged words for a single item is identified (i.e.) the frequency of the word. Different types of word cloud are available, we use collocate word cloud gives the more focused view of the corpus. The word resulting after processing collocate word cloud is the word used most often in conjunction with the search word. For easy understanding of the visualization the frequency is denoted as size and the collocation strength as brightness.

Following terms are used in a word cloud

(i) Size- Large tags attracts the users than small tags.
(ii) Centering- The word in the middle of the cloud gets more attention than in the border.

When searching for the specific word, the tag word cloud provides very good support. The word cloud package in R allows the users to state the maximum number of words to be plotted and minimum frequency above which the words should be displayed are taken as parameters of the word cloud() function.

3. Performance Measure

3.1 Experimental Setup

There are many freely available tools for detecting the sentiment of the tweets, but what methodology they have used for analyzing the tweets is not known. In the work [6] ontology based sentiment analysis, the ontology was created based on the particular domain and the features related to the domain are extracted as tweets and pre-processing is done, but for analyzing the sentiment they used Open-Dover which is the third party sentiment analysis tool and gives the scores for the features.

In this paper, our proposed method gives the scores for each tweet and the methodology used in analyzing the tweets are known, using the open NLP and NLP package we can find the sentence tokenizer and POS tagger. Using the word net dictionary the lemmatization of the words is done. Sentiment package helps in sentiment analysis. After the sentiment of each features are done the comparison is done, which helps in identifying the comparative results of the search terms.

The figure for the comparative results for search term

Figure 7 shows the comparative results of the feature 'battery' for four laptops and the mean value is calculated for the above scores which help in analyzing the results easily. The mean value of Acer battery is 0.88, for HP mean is 0.01, for Lenovo 0.1 and for Apple the mean value is 0.29. From this we can come to a conclusion that Acer battery is better than other batteries'.

The result is compared with the scores of the American Customer Satisfaction Index for laptops. The score table is retrieved from the ACSI URL and compared with our score calculation, merge () function helps in merging the matching rows and shows the comparative results for our twitter score and ACSI score, from the score it is identified that twitter score gives the best result when compared to the ACSI score. The following figure gives the overall comparison result:

From Figure 8 score.acsi is ACSI score and score. Twitter is sentiment score done based on twitter. For

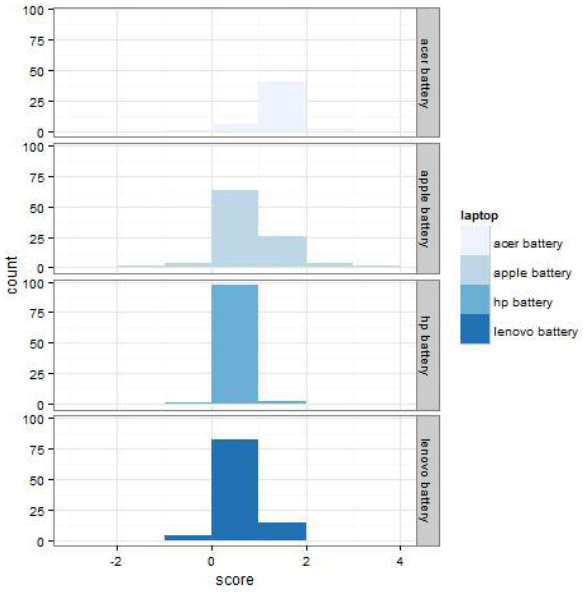

Figure 7. Comparable results of laptops.

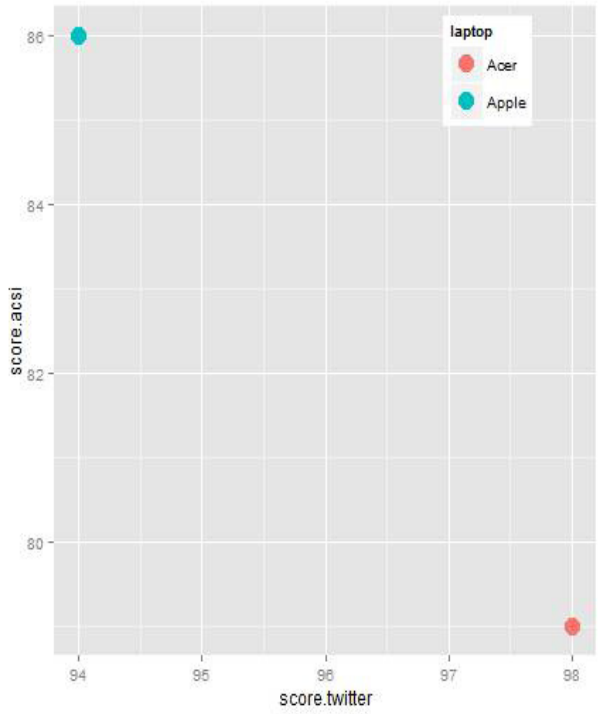

Figure 8. The score comparison of twitter and ACSI.

Acer, twitter score is 98 and ACSI score is 79, for Apple, twitter score is 94 and ACSI score is 86. An overall score of twitter is best compared to ACSI score.

4. Conclusion and Future Work

The results proved experimentally in Figure 7 and Figure 8 shows that the proposed custom sentiment analysis tool for twitter increases the performance by increasing the overall scoring of tweets compared to the third party result, with the deployment of the ontology the subjects discussed in tweets, are analyzed and the score is assigned to the feature of laptop and also the opinion words helps in identifying the mood of people and scores the tweets to the best. The future work of this proposed approach can be building a fully automatic ontology method.

5. References

1. Agarwal A, Xie B, Vovsha I, Rambow O, Passonneau R. Sentiment analysis of twitter data. Proceedings of the ACL 2011 workshop on languages in social media; 2011. p. 30–8.
2. Claster WB, Cooper M, Sallis P. Modeling sentiment from twitter tweets using Naïve Bayes and unsupervised artificial neural nets. Proceedings of the CIMSIM'10; 2010. p. 89–94.
3. Davey BA, Priestley HA. Introduction to lattices and order. Cambridge University Press; 2002. ISBN 978-0-521-78451-1.
4. Efstratios K, Christos B, Theologos D, Nick B. Ontology-based sentiment analysis of twitter posts. Expert systems with applications. 2013.
5. Soleimanian GF, Reza KS, Isa M. A new approach in bloggers classification with hybrid of K-nearest neighbor and artificial neural network algorithms. Indian Journal of Science and Technology. 2015 Feb; 8(3):237–46.
6. Ganter BB, Wille R. Formal concept analysis. Mathematical foundation; 1999.
7. He Y, Alani H. Semantic smoothing for twitter sentiment analysis. Proceedings of the10th international semantic web conference (ISWC); 2011.
8. Iwanaga I, Nguyen TM, Kawamura T, Nakagawa H, Tahara Y, Ohsuga A. Building earthquake evacuation ontology from twitter. Proceedings of the IEEE international conference on granular computing (GrC); 2011. p. 306–11.
9. Karthikeyan K, Karthikeyani V. Ontology based concept hierarchy extraction of web data. Indian Journal of Science and Technology. 2015 Mar; 8(6):536–47.
10. Ning L, Guanyu L, Li S. Using formal concept analysis for maritime ontology building. Proceedings of the 2010 international forum on information technology and applications (IFITA'10); 2010; 2:159–62.
11. Obitko M, Snasel V, Smid J, Snasel V. Ontology design with formal concept analysis. Concept lattices and their applications. 2004. p. 111–9.
12. Pak A, Paroubek P. Twitter as a corpus for sentiment analysis and opinion mining. Proceedings of the 7th international conference language resources and evaluation (LREC '10); 2010.
13. Pang B, Lee L. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval. 2008. p. 1–135.
14. Zol S, Mulay P. Analyzing Sentiments for Generating Opinions (ASGO) - A New Approach. Indian Journal of Science and Technology. 2015 Feb; 8(S4):206–11.
15. Mukherjeet S, Malu A, Balamurali AR, Bhattacharyya P. TwiSent: A multistage system for analyzing sentiment in twitter. 21st ACM Conference on Information and Knowledge Management; 2012.
16. Tumasjan A, Sprenger TO, Sandner PG, Welpe IM. Predicting elections with twitter: What 140 characters reveal about political sentiment. Proceedings 4th international AAAI conference on weblogs and social Media. 2010. p. 178–85.
17. Vigneshwari S, Aramudhan M. Social information retrieval based on semantic annotation and hashing upon the multiple ontologies. Indian Journal of Science and Technology. 2015 Jan; 8(2):103–7.

18. Zhang R, Xu H. Building the ontology system in semantic web based on formal concept analysis and rough set. Journal of Convergence Information Technology. 2011. p. 56–62.
19. Available from: https://dev.twitter.com/oauth/overview/application-owner-access-tokens
20. Available from: http://www.aclweb.org/anthology/P07-1123
21. Available from: http://people.cs.pitt.edu/~wiebe/
22. Available from: http://protege.stanford.edu/
23. Available from: http://www.ontocat.org/
24. Available from: http://www.r-project.org/