International Journal of Pure and Applied Mathematics
Volume 119 No. 14 2018, 1593-1601
ISSN: 1314-3395 (on-line version)
**url:** http://www.ijpam.eu
Special Issue
ijpam.eu

# DISCOVERING SEMANTIC USER PRODUCT REVIEWS USING HYPERGRAPH AND ONTOLOGY (SPGO)

Rashmee Dhamecha[1], Rahul Joshi[2], Preeti Mulay[3]
[1,2,3]CS, SIT, Symbiosis International University

**ABSTRACT --** Most of the product reviews and opinions are based on the textual feedback of the users which often yield wrong impression about the product. The inclusion of the product used in the form of image or video which eventually enriches the process of classification of the product ratings. Many systems are existed which are considering the users rating in the form of numerical rating or stars (*) to classify the sentiments or opinions of the users which yields the dramatized results. So as a tiny step towards this to provide proper classification of the users rating, proposed system introduces the using of NLP protocols on users comments and product usage visual pattern using Region of interest (ROI). And the whole process is catalyzed by the usage of semantic tree structure of Ontology and hyper graph traversal technique along with fuzzy classification to yield improved classification segments.

**Keywords** --- Fuzzy Classifier, NLP Protocol, Ontology, ROI, Semantic Tree.

## INTRODUCTION

In today's world for buying or going for any product, online reviews play most important part. User experiencing a product is the key to achieve better performance and accuracy. Analysis of product is done to identify the user reviews based on the product. This results in determining the usage of a product. User experience is all about what, why, where, when and who uses the product that determines the changing patterns in user's behavior based on which the overall usability of products is improved, by finding what the user needs. The concept of user experience includes two key concepts: usability and affect. Usability of a product is completely relevant on "how the product is being used?" .User's experience on the usage is used to achieve a specific goal. The second aspect focuses mainly on the user satisfaction with the product by identifying subjective feelings through interviews and various physiological measurements. Image schemas can be interpreted as Psychological patterns that form the human experiences in the world. The Fact-Finding of this paper is to link the user's attitudes and feelings to the image schemas involved in the interaction with a product. This is possible by using semantic method for exploring user's experience based on using image schema that leads to better understanding of the value associated with the particular design features.

Algorithm for analyzing the sentiments was first introduced on Chinese micro-blog content, which is actually accurate but not efficient. Due to this reason the algorithm that performs well in both accuracy and efficiency was introduced. The algorithm was optimized into three stages: Data structure optimization, query strategy optimization, and parallel optimization. Sentiment analysis has gained wide popularity in the fields of e-commerce and opinion mining. Later, the study was done on movie reviews to gain easier access to information about the movie; the users would like to watch. In some cases, comments might carry both positive and negative feelings simultaneously that becomes hard to detect. Some users make sarcastic comments with the purpose to make jokes about the characters, which is hard to detect, so the lexicons are updates by using lexical updating algorithm to make it best suit the movie reviews.

The social network has gathered a huge attention of the people and is considered one of the most disruptive innovations since last decade. The posts of the users of social media are used by the company to find the mentality of the users and many more things. Most of the research hypothesis is to find the relation between the users on the social network especially the negative links, to improve the recommendation system in social media using link prediction algorithm. This helps to find if there is some probability between two links of disliking each other than those users can be removed from each other's recommendation list or could assign lower weight to the users.

This paper is been classified for Literature survey under section 2.section 3 is dedicated for detailed narration of our proposed model. Whereas section 4 is dedicated for Comparative Analysis of our model and finally section 5 concludes the paper with its future enhancement scope.

## I. LITERATURE SURVEY

Jingbo Zhu and team demonstrate that opinion polling is a traditional approach to know about the quality of products and services from customers. It was done by asking questions to customers. The researchers proposed Aspect-based opinion polling which does not require any questions to be answered by customers. To learn the aspect related terms, a multi-aspect bootstrapping method is proposed which is used for identifying the aspects. The author proposed aspect based segmentation model and aspect-based polling algorithm. These algorithms are used to partition the sentence consisting of multiple aspects into single aspects. After segmenting the multiple aspect-based terms to single aspect, the accuracy of the algorithm is checked by using fivefold cross-validation technique [1].

The research work summarize that many users keep their different names on different social networking sites and it becomes necessary to identify the false identity of user's names. A lexical pattern-based approach is used to extract the fake identity of person name. This approach outperforms in three steps: extracting the set of aliases. The ranking is done in three ways: 1.Lexical patterns frequency.2.co-occurrences in anchor text and 3. Page counts based association measures. Finally, ranking scores are integrated using ranking support vector machine [2].

Similar users may express different sentiments in different domain. Sentiment analysis is done on users reviews. User reviews are classified in train domain and test domain to avoid the mismatch of sentiments. Hence a methodology is introduced to classify the sentiments from different domain that includes words with same semantics. The sentiments that express same sentiments are grouped together. For this the author used created thesaurus to expand feature vectors for binary classifier .The paper contributes to methods of finding relevant information from unstructured text with opinion and sentiment analysis [3,4].

The research work uses techniques that focus on five specific problems within the field of sentiment analysis: Document-level sentiment analysis, Sentence-level sentiment analysis, aspect-based sentiment analysis, sentiment lexicon acquisition. It presents a comprehensive technique to recognition to all sentiment expression within a given document and the aspects to which they refer [5]. In this paper, the work is restricted to only five subjective terms. But it is identified that there are other terms that are uncovered in this research and they are mood, sensation, attitude, and temperament that falls under the same category. So, a common framework should be developed to identify other subjective terms.

Jiaru Lin, and Xinyu Wang demonstrates that with the increasing data or information, it is necessary to maintain accuracy while analyzing the sentiments. The author investigated the algorithm on Chinese micro-blog first, but the algorithm does not perform well in efficiency but has greater accuracy. Due to this reason the algorithm that performs well in both accuracy and efficiency was introduced. The algorithm was optimized into three stages: Data structure optimization, query strategy optimization and, parallel optimization. The platform offers two main functions including text sentiment analysis and user sentiments and the results are then analyzed through the open group of API's [6]. [7] Proposes the idea (2011) that differentiates between the mentioned five subjective terms. The major drawback in automatically detecting the parameters like affect, emotions, feelings, opinions, and sentiments are - no proper differentiation between these subjective terms and the awareness of how these terms correlate to one another. This paper clarifies that not all the words have same synonyms.

The researcher modified the work by implementing a framework model for classifying the sentiments based on sentence-level sentiment analysis that predicts sentence level polarity based on the segmentation results and generates useful segmentation. A candidate generation model is developed, to generate the segmentation of the parameters of the sentence. This paper also proposes a model for segmentation ranking that result in sentiment classification by highlighting the usefulness of segmenting the candidate of a sentence. A classification model is proposed to predict the sentence level polarity based on the segmentation results [8].Opinion mining plays a very important role in summarizing the product by explaining the usage of the product by giving ratings or answering queries relevant to the product. People across the world are based on the reviews of the product before purchasing them. Khan and Jeong [13] presented a Lazy Learning supervised model and Lazy learning classification technique i.e. K Nearest Neighbor to classify the review sentences into subjective and objective classes.

The algorithm focuses on Aspect-level sentiment analysis that aims to find the sentiments and aggregate the sentiments that are listed in the document based on aspect-level. [9] In today's era Each and every second a new data is being generated and formed. So, an incremental clustering algorithm is proposed using Pearson's coefficient of correlation to easily cluster the data that becomes useful for professionals as well non-professionals to read and understand [10]. [14] Explained and presented the expectation maximization algorithm to extract features from product reviews along with the Latent Semantic Indexing (LSI) that analyzes the documents contains the text reviews and also the related concepts or meaning of those documents that result in determining the polarity and dependency of each word into another one.

## II. PROPOSED MODEL

This section describes the process of identifying user behavior with respect to a product based on the user's comments and usage activity patterns of a product. The details are depicted in the following steps

**Step 1:** As in the first step of SPGO Proposed model collects the review data from Amazon by using following URL: https://app.feedback.co/amazon-review-exporter. This data is collected for the electronic gadget like

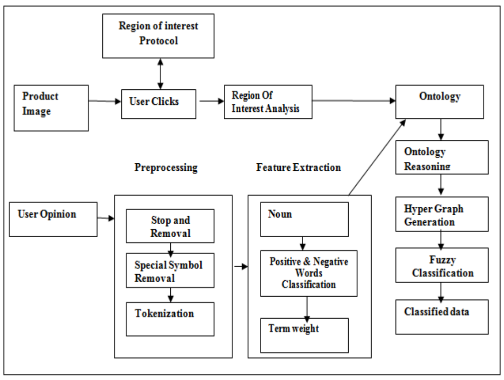



Figure 1.Overview of our Approach mouse, After this all the data is been stored in workbook with column names like serial no., title, comments, username and ratings Where the serial No. and ratings are in numerical format like integers ,comments and title is in the form of String.

**Step 2:** This step of our classification model evaluates the profiled sample activity of the users on the product. This is recorded based on the user clicks on the product image on their interesting segment of the product. These Segments includes clicks on the left, Clicks on the right, Clicks on the scroller of the mouse. The user clicks are being recorded for the number of rows of the dataset to the randomly generated Cartesian coordinate points on the image canvas. This can be represented by Algorithm 1.

---

ALGORITHM 1: ROI Evaluation

---

// Input: User Behavior Clicks

// Protocol Set $P_R = \{R_{CP}, L_{CP}, S_P\}$

//Output: Click weight List $C_{WSET}$

Function: getClickWeightList()

**1.** Start.
**2.** $R_{CC} = \emptyset$, $S_{CC} = \emptyset$, $L_{CC} = \emptyset$, $C_{WSET} = \emptyset$
**3.** [$R_{CC}$ = Right click count, $S_{CC}$ = Scroll click count, $L_{CC}$ = Left click count]
**4.** $R_P = P_{R[0]}$ , $L_P = P_{R[1]}$ , $S_P = P_{R[2]}$
**5.** x= MouseEvent_xAxis();
**6.** y= MouseEvent_yAxis();
**7.** **if**($x \in R_P$ and $y \in R_P$ )
**8.** $R_{CC}$++;
**9.** **if**($x \in L_P$ and $y \in L_P$ )
**10.** $L_{CC}$++;
**11.** **if**($x \in S_P$ and $y \in S_P$ )
**12.** $S_{CC}$ ++;
**13.** Add $R_{CC}$ to $C_{WSET}$;
**14.** Add $L_{CC}$ to $C_{WSET}$;
**15.** Add $S_{CC}$ to $C_{WSET}$;
**16.** **return** $C_{WSET}$

---

**Step 3:** SPGO takes the input of user comments that is in workbook format. This workbook data is been read in the form of double dimension list in Strings using an external API called jxl. Once the data is read in jxl then this data is subjected to preprocessing as mentioned in the below steps.

**1) Special Symbol Removal**: Here in this step all the words are being separated into a list. Then these words are being checked for the special symbols like,,.,:,;,?,{,},[,] etc…If a word contains any of these special symbols, then it is being replaced with the empty character to get the plane word.

**2) Stopword Removal:** All the conjunctive words of parts of Speech in English Language are loaded into the linear List. Then each word of the comments are shred off based on the presence of the word into conjunction list. This yields comments which are free of stopwords. This eventually decreases the redundancy of data.

3) **Stemming:** The stopword removed words are been subjective to identify Stemming Protocols with the words like ing, ed, ation, etc. If any of these Strings appeared in the word then it is been shred off to get the base form of the word.

This whole process of pre-processing eventually reduces the redundancy and makes the String light weight by returning the same direction.

**Step 4:** To identify the semantics of the user review SPGO uses some features which really help to identify the deeper and greater depth of semantics with the help of below mentioned steps.

1) **Term Weight:** The pre-processed string from the prior step is considered in this section to get the feature of term weight. Term weight can be described as the weight of each word in the given string. To get the term weight of each word in the string, all the words of the strings are enrolled in a list and then this list is being used get the unique word from it. Unique word is being obtained by using the set of hashes of the word. Then this unique list being weighted for each of the words to get the term weight.

2) **Proper Noun:** This step helps to identify the proper noun from the given comments. For this process proposed model uses the maximum English language words, that is about 100 thousand. Which is collected through the different web forums of oxford dictionary in workbook format. So these workbooks actually contains all the English language words excepts the nouns. And these workbooks are stored alphabetically which makes easy to check whether a word is starts with the given alphabet contains in the dictionary workbook or not. Based on this that is going to be decide as noun or not.

3) **Positive and negative word identification:** Set of positive and negative words are been kept in a list to identify each words of the comments as positive or negative and separate them in a list with their frequency.

**Step 5:** The positive and negative words which are extracted in the prior step are not clearly indicate the boundary between the positivity and negativity of a word. So that means extracted words can have poor semantics. For example, “Not good", this phrase contains a negative word behind a positive word that makes the positive word to behave as negative.

These kinds of words can be more efficiently treated with a semantic tool like protégé. Where protégé is the tool which builds the ontology tree by providing the valid hierarchy and stores in the xml file with extension of owl( Web ontology language). This owl file contains the info stated with RDF format. So our proposed model uses this xml to reason the positive and negative words with the steps as mentioned in algorithm 2.

---

ALGORITHM 2:Ontology Reasoner

---

// Input: .OWL File

// Output: Semantic Tree

1. Start
2. T = Ø // (Tree)
3. Str = FileString(.OWL File)
4. **If**($C_{Tag} \in$ Str)
5. [$C_{Tag}$ = class Tag]
6. Root = $C_{Tag}$ label
7. **While** (Str != EoF)
8. **If**($S_{CTag} \in C_{Tag}$)
9. Then create child node $C_N$
10. Add $S_{CTag}$ label to $C_{N.}$
11. **End if**
12. **End while**
13. **return T**

---

**Step 6:** Once the classification of positive and negative word is done. Then unique nodes are identified which are bounded by the respective edges which indicate the relationship between the two nodes in the form of closest semantics. This process is catalyzed by the other three features like term weight, proper nouns and user profile activity of the product on image clicks.

Then by using these nodes and edges a hyper graph is created where common features are clustered by traversing on inter edge formation in between the tiny clusters. This graph is stored in most advanced graph database like neo4j.which will be used further for comment classification process. The process of hyper graph generation algorithm is depicted in algorithm 3

---

ALGORITHM 3: Hyper graph creation

---

// Input: node list $N_L$
// Output: graph object G (U, W)

1. Start.
2. G ($U_i$, $W_i$) = Ø //graph object.
3. **for** i = 0 to size of $N_L$
4. $T_{SET} = N_L$
5. $F_{Nodei} = T_{SET[0]}$ ,
6. $T_{Node\ i} = T_{SET[1]}$ ,
7. $Edge_i = T_{SET[2]}$ ,
8. **for** j=0 to size of $N_L$
9. $T_{SET} = N_L$
10. $F_{Node\ j} = T_{SET[0]}$ ,
11. $T_{Node\ j} = T_{SET[1]}$ ,
12. $Edge_i = T_{SET[2]}$ ,
13. **If** ($F_{Node\ i} == F_{Node\ j}$)
14. $T_{G(u,w)}$ $F_{Node\ i}$ -> $T_{Node\ j}$ -> $T_{SET[2]}$
15. Add $T_{G(uij,wi)}$ to G(u,w)
16. **End for**
17. **End for**
18. **return** G(u,w)

---

**Step 7:** Here SPGO uses edge and node relations which represent user and comment features. Which are being numbered for the positive and negative comments respectively to form a vector. Where positive comments are having raising values and negative comments are having inverse of these raising values. Which are then feed to Pearson correlation equation to evaluate the correlation between the comments and classification labels. Which yields the output of correlation in between 0 and 1. where any value nearer to 1 indicates the positivity of the comments and nearer to 0 indicates the negativity of the comments.

$$r = \frac{\sum xy - \frac{\sum x \sum y}{n}}{\sqrt{\left(x^2 - \frac{\sum x^2}{n}\right)}\sqrt{\left(y^2 - \frac{\sum y^2}{n}\right)}} \quad (1)$$

Where

1. x is the positive comment raising numerical value
2. y is the negative comment inverse numerical value
3. n is the size array

**Step 8:** Here in this step all the correlations of the user comments are been recorded, which are in between 0 and 1. Now this value is used to generate the fuzzy crisp values like VERY LOW, LOW, MEDIUM, and HIGH AND VERY HIGH.

Comments are been scrutinized for the highest crisp value factors to identify the exact matched classification protocol to semantically classify the user comments more efficiently into positive, negative and neutral comments. This process of classification of the review can be shown in the algorithm 4.

---

ALGORITHM 4:Review Classification

---

//Input : Correlation vector $C_V$ ,$f_C$[fuzzy crisp set]

//Output: Classified list $C_L$

1. Start
2. **for** i=0 to size of $C_V$
3. $T_{Set} = C_{V[i]}$
4. $C_{LV} = T_{Set[1]}$
5. [$C_{LV}$=Correlation vlaue]
6. **if** $C_{LV} \in f_C$
7. add $T_{Set}$ to $C_L$
8. **end if**
9. **end for**
10. **return** $C_L$

---

## III. RESULT AND DISCUSSIONS

Proposed model of SPGO conducts some experiments to prove its ability to classify the comments semantically in better rates. For this experimental evaluation of the system, Proposed methodology uses a machine of standard configuration with 4GB primary memory and Core i5 processor. The System uses windows based java enabled machine with Netbeans 8.0 as IDE and MySQL 5.0.22 as Database Server. To construct the tree patterns for the positive and negative words system uses the web protégé 4.0 tool from online with URL https://webprotege.stanford.edu/ for ontology. And to store the dynamic graph model uses most advanced graph database like neo4j.

To measure the performance of the system proposed methodology uses some experiments as mentioned below

| No. of given coments | No of Actual Positive Comments | Relevant Positive comments identified(A) | Irrelevant Positive comments identified(B) | Relevant Positive comments not identified(C) | Precision =(A/(A+B))*100 | Recall= (A/(A+C))*100 |
|---|---|---|---|---|---|---|
| 50 | 24 | 21 | 3 | 4 | 87.5 | 84 |
| 100 | 61 | 54 | 7 | 8 | 88.52459016 | 87.09677419 |
| 150 | 134 | 120 | 14 | 16 | 89.55223881 | 88.23529412 |
| 200 | 160 | 137 | 21 | 24 | 86.70886076 | 85.0931677 |
| 250 | 225 | 209 | 24 | 26 | 89.69957082 | 88.93617021 |
| 300 | 267 | 222 | 28 | 25 | 88.8 | 89.87854251 |
| 350 | 321 | 278 | 48 | 50 | 85.27607362 | 84.75609756 |
| 400 | 378 | 335 | 55 | 61 | 85.8974359 | 84.5959596 |
| 450 | 422 | 382 | 62 | 66 | 86.03603604 | 85.26785714 |
| 500 | 467 | 401 | 68 | 71 | 85.5010661 | 84.95762712 |

**Performance Evaluation based on Precision and Recall**

Precision and recall are considered as the one of the best parameter to measure the performance of our system. Precision is known for the positive predictive values that indicate the amount of relevant information extracted through the system. Precision can be stated as the ratio of number of relevant patterns extracted for the given number of user comments to the sum of number of relevant and irrelevant patterns extracted for the given number of user comments. Relative effectiveness of the system can be evaluated thoroughly by using precision parameters.

Recall generally indicates the part of relevant results extracted over the extracted relevant results. Recall can be defined as the ratio of number relevant patterns extracted to the sum of relevant patterns not extracted. Absolute accuracy of the system can be properly denoted by this system.

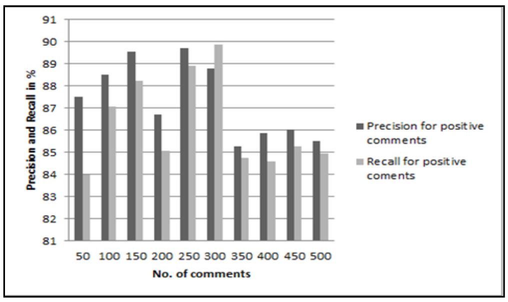
Precision can be more effectively explained as below

1. A = The number of relevant patterns extracted for the given number of comments
2. B= The number of irrelevant patterns extracted for the given number of comments
3. C = The number of relevant patterns are not extracted for the given number of comments

So precision can be given as

Precision = ( A / ( A+ B)) *100

Recall = ( A / ( A+ C)) *100




On conducting of experiments for precision and recall different reading are been recorded for positive, negative and Neutral words as depicted below with the tables and the graphs.

Table 1: Precision and Recall Estimation for Positive comments

Figure 2: Plot for Precision and Recall for Positive Comments

Table 2: Precision and Recall Estimation for Negative Comments

| No. of given coments | No of Actual Negative Comments | Relevant negative comments identified(A) | Irrelevant Negative comments identified(B | Relevant Negative comments not identified(C) | Precision =(A/(A+B))*100 | Recall =(A/(A+C))*100 |
|---|---|---|---|---|---|---|
| 50 | 17 | 11 | 4 | 2 | 73.33333333 | 84.61538462 |
| 100 | 26 | 22 | 4 | 3 | 84.61538462 | 88 |
| 150 | 7 | 4 | 1 | 1 | 80 | 80 |
| 200 | 21 | 19 | 4 | 3 | 82.60869565 | 86.36363636 |
| 250 | 11 | 7 | 1 | 2 | 87.5 | 77.77777778 |
| 300 | 20 | 14 | 2 | 4 | 87.5 | 77.77777778 |
| 350 | 15 | 9 | 1 | 3 | 90 | 75 |
| 400 | 9 | 6 | 1 | 2 | 85.71428571 | 75 |
| 450 | 12 | 8 | 4 | 1 | 66.66666667 | 88.88888889 |
| 500 | 10 | 9 | 0 | 1 | 100 | 90 |

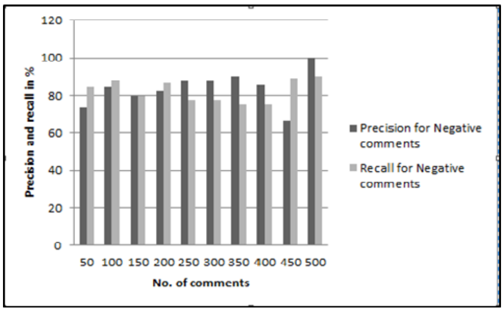



Figure 3: Plot for Precision and Recall Estimation for Negative Comments

Table 3: Precision and Recall Estimation for Neutral Comments

| No. of given coments | No of Actual Neutral Comments | Relevant neutral comments identified(A) | Irrelevant Neutral comments identified(B) | Relevant Neutral comments not identified(C) | Precision =(A/(A+B))*100 | Recall= (A/(A+C))*100 |
|---|---|---|---|---|---|---|
| 50 | 9 | 7 | 1 | 1 | 87.5 | 87.5 |
| 100 | 13 | 11 | 2 | 2 | 84.61538462 | 84.61538462 |
| 150 | 9 | 6 | 1 | 2 | 85.71428571 | 75 |
| 200 | 19 | 17 | 4 | 3 | 80.95238095 | 85 |
| 250 | 14 | 10 | 2 | 3 | 83.33333333 | 76.92307692 |
| 300 | 13 | 11 | 2 | 2 | 84.61538462 | 84.61538462 |
| 350 | 14 | 12 | 1 | 1 | 92.30769231 | 92.30769231 |
| 400 | 13 | 9 | 2 | 1 | 81.81818182 | 90 |
| 450 | 16 | 12 | 2 | 3 | 85.71428571 | 80 |
| 500 | 23 | 17 | 3 | 4 | 85 | 80.95238095 |

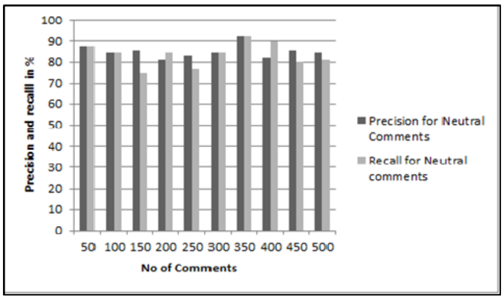



Figure 4: Precision and Recall for the neutral Comments

When SPGO model is compared with that of [14] which is Analyzing the sentiments based on the unsupervised and supervised Co- occurrence Data. Unsupervised method of this paper uses association rule mining to identify the sentiments through co-occurrence data. On the other hand supervised technique uses co-occurrence through weight matrix method. As we know the Association rule mining includes more number of candidate sets which eventually yields large unnecessary relations between the sentiments.

Whereas the proposed model SPGO uses the tree and graph approach which yields better results than this. This can be seen with the below mentioned table and plot.

Table 4: Comparison Table

| Methods | Average Precisio | Average Recall |
|---|---|---|
| SPGO | 85.42 | 84.0933 |
| Unsupervised & Supervised | 84.4 | 83.1 |

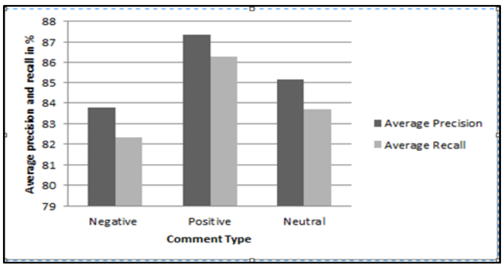



Figure 5: Comparison plot of SPGO

## IV. CONCLUSION AND FUTURE SCOPE

The Proposed model of SPGO efficiently classifies the comments for an electronic gadget like mouse collected from the amazon.com in the form of workbook format. This paper narrates all the techniques that are included to enhance the comment classification process which uses some modern tools like web protégé and Graph database like neo4j.

Using of graphs and trees efficiently identifies the user comments relations between users to product which increase the efficiency of the system when compared to that of [14] that is discussed in the prior section.

Classification of comments process can be implemented on live web pages of many online social networking sites by designing a high caliber web crawler to analyze mood of the users based on the current scenario.

## REFERENCES

[1] Zhu, J., Wang, H., Zhu, M., Tsou, B. K., & Ma, M. (2011). Aspect-based opinion polling from customer reviews. IEEE Transactions on Affective Computing, 2(1), 37-49.

[2] Ghose, A., & Ipeirotis, P. G. (2011). Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. IEEE Transactions on Knowledge and Data Engineering, 23(10), 1498-1512.

[3] Bollegala, D., Weir, D., & Carroll, J. (2013). Cross-domain sentiment classification using a sentiment sensitive thesaurus. IEEE transactions on knowledge and data engineering, 25(8), 1719-1731.

[4] Cambria, E., Schuller, B., Xia, Y., & Havasi, C. (2013). New avenues in opinion mining and sentiment analysis. IEEE Intelligent Systems, 28(2), 15-21.

[5] Feldman, R. (2013). Techniques and applications for sentiment analysis. Communications of the ACM, 56(4), 82-89.

[6] Zhao, Y., Niu, K., He, Z., Lin, J., & Wang, X. (2013, October). Text sentiment analysis algorithm optimization and platform development in social network. In Computational Intelligence and Design (ISCID), 2013 Sixth International Symposium on(Vol. 1, pp. 410-413). IEEE.

[7] Munezero, M. D., Montero, C. S., Sutinen, E., & Pajunen, J. (2014). Are they different? Affect, feeling, emotion, sentiment, and opinion detection in text. IEEE transactions on affective computing, 5(2), 101-111.

[8] Kennedy, A., & Szpakowicz, S. (2014). Evaluation of automatic updates of Roget's

Thesaurus. Journal of Language Modelling, 2(1), 1-49.

[9] Tang, D., Qin, B., Wei, F., Dong, L., Liu, T., & Zhou, M. (2015). A joint segmentation and classification framework for sentence level sentiment classification. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23(11), 1750-1761.

[10] Chen, T., Xu, R., He, Y., Xia, Y., & Wang, X. (2016). Learning user and product distributed representations using a sequence model for sentiment analysis. IEEE Computational Intelligence Magazine, 11(3), 34-44.

[11] Khan, J., Jeong, B. S., Lee, Y. K., & Alam, A. (2016, October). Sentiment analysis at sentence level for heterogeneous datasets. In Proceedings of the Sixth International Conference on Emerging Databases: Technologies, Applications, and Theory (pp. 159-163). ACM.

[12] Shinde, K., & Mulay, P. (2017, April). Cbica: Correlation based incremental clustering algorithm, a new approach. In Convergence in Technology (I2CT), 2017 2nd International Conference for (pp. 291-296). IEEE.

[13] Shelke, N. M., Deshpande, S., & Thakre, V. (2017, March). Exploiting expectation maximization algorithm for sentiment analysis of product reviews. In Inventive Communication and Computational Technologies (ICICCT), 2017 International Conference on (pp. 390-396). IEEE.

[14] Schouten, K., van der Weijde, O., Frasincar, F., & Dekker, R. (2018). Supervised and Unsupervised Aspect Category Detection for Sentiment Analysis with Co-occurrence Data. IEEE transactions on cybernetics, 48(4), 1263-1275.