SENTIMENT-BASED MODEL FOR RECOMMENDER SYSTEMS

1st Nurul Aida Osman
Center for Artificial Intelligence Technology
Universiti Kebangsaan Malaysia
Selangor, Malaysia
p78944@siswa.ukm.edu.my

2nd Shahrul Azman Mohd Noah
Center for Artificial Intelligence Technology
Universiti Kebangsaan Malaysia
Selangor, Malaysia
shahrul@ukm.edu.my

Abstract— **Recommender systems have proven to be a valuable way for online users to cope with the issues of information overload. They have become one of the most powerful and popular tools in electronic commerce as illustrated by Amazon.com, YouTube, Netflix, Yahoo, and IMDb. While recommender systems have shown significant contribution, they still suffer from the long-standing problems related to cold-start users and data-sparsity. This is due to the fact that recommendation algorithms mostly rely on users' rating to make prediction of items. Such ratings are usually insufficient and very limited. On the other hand, sentiment ratings of items which can be derived from online news services, blogs, social media or even from the recommender systems themselves are seen capable of providing better recommendations to user as opposed to tags alone. Sentiment-based model has been exploited in recommender systems to overcome the data-sparsity problem that exists in conventional recommender systems. Hence, embedding sentiment in recommender systems may significantly enhance the recommendation quality of recommender systems. Among the aims of this research is to integrate sentiment analysis in recommender systems particular to those items with no associated rating that commonly contribute to the problem of data-sparsity.**

Keywords—recommender systems, collaborative filtering, sentiment analysis, opinion mining

I. INTRODUCTION

The issues contribute to the two long-standing problems of recommender systems are cold-start users and data-sparsity. Data sparseness can lead to cold-start problem that will reduce the performance of recommender systems. Users tend to express their preferences and opinions of items by writing comments and reviews. The existing recommendation methods only explore ratings and metadata but do not analyze what users have to say about particular content of the items. The comments are excellent indicators of users satisfaction. Sentiment analysis algorithms offer an analysis of the user's preferences in which the comments may not be associated with an explicit rating. This will have an impact on the popularity of the recommendation item. Thus, sentiment analysis are used to determine the words or sentences that have sentiment value [2].

Hence, this study aims to investigate the contribution of sentiment analysis towards recommender systems. This paper provides the solution to minimize the data sparseness in recommender systems by integrating the sentiment-based model in collaborative filtering recommendation method.

II. RELATED WORK

A. Recommender Systems

Recommender Systems are software tools and techniques providing suggestions for items to be of use to a user. The suggestions provided are aimed at supporting their users in various decision-making processes, such as what items to buy, what music and implementation consideration, setting guidelines for the selection to listen, or what news to read. Recommender systems have proven to be valuable means for online users to cope with the information overload and have become one of the most powerful and popular tools in electronic commerce. Correspondingly, various techniques for recommendation generation have been proposed and during the last decade, many of them have also been successfully deployed in commercial environments.

Recommender systems are usually classified into the following categories according to the approach of recommendations [3]:

a) Content-based recommendation: in which a user is recommended items that are similar to those that the user liked in the past

978-1-5386-3812-5/18/$31.00

b) Collaborative recommendation: using Collaborative Filtering (CF) algorithm where a user is recommended items that other user with similar tastes liked in the past.

c) Hybrid approaches: combination of content-based and collaborative methods.

In CF recommender systems, rating, which indicates how a particular user liked particular item has been the most popular representation to date [4]. Ratings however, have limitations particularly when the dimension of user-item matrix increases and evolves that may result to data sparsity problem. As a result, few research efforts have took place to overcome the aforementioned problem [9]. One of the effort is to elicit user preferences expressed in textual reviews, a problem known as sentiment analysis, and map such preferences onto some rating scales that can be understood by existing CF algorithms.

B. Sentiment Analysis and Opinion Mining

Sensing the mood and the preferences of user through text analysis techniques is a research area that has been quite active in recent years. The sentiment classification is commonly tackled with binary classifiers. However, within the context of recommender systems, sentiment must be correlated with ratings. In other words, it is not suffices to classify specific statements as positive or negative but to be able to correlate such statements in the form of ratings. One of the most important tasks in sentiment analysis is to identify which words express a sentiment [2]. Words which are associated with a sentiment intensity are referred as opinion words. Opinion words can be recognized either by manual, corpus-based or dictionary-based approaches. As manual approaches are very time- consuming, it is suffices to combine with other automated methods. SentiWordNet dictionary is a popular linguistic resource in sentiment analysis which provides an answer to the "how and which words people use to express preferences?" [5]. Previous studies using the lexicon of SentiWordNet have shown promising results as discussed in [6].

C. Sentiment Based Recommendation Approach

Among the goal of this research is to integrate sentiment analysis in recommender systems particular to those items with no associated rating that commonly contribute to the problem of data-sparsity. Based on the existing literature on sentiment-based recommender systems, most approaches depend on conventional sentiment-analysis approaches which lack of semantic and contextual information. Research in sentiment analysis have shown such information can produce better sentiment accuracy [7]. According to [1], a problem known as sentiment analysis in textual review expressed by users can be understood by integrating a rating inference approach into the existing CF algorithm. It infers numerical ratings from textual reviews, so that user preferences represented in the reviews can be fed into existing CF algorithms [8]. Previous studies also adopting sentiment analysis technique in recommender systems by categorizing users according to the average polarity of their comments. A research by [2] introduced a framework involve analyzing unrated users comments and reviews and generate them into sentiment ratings. The sentiment ratings will then merged with the user-item matrix to obtain a multi-dimension criteria for recommendation. Recommendations of ranked items will then be presented to users. However, most of the approaches are using conventional sentiment analysis techniques. Those techniques have limitation where there is no contextual information in different domains.

Recommender systems are having difficulties to provide precise recommendation in different domains. However, embedding such enhanced-sentiment approaches were yet to be investigated in sentiment-based recommender systems. Hypothetically, improved accuracy of sentiment results will lead to better performance of recommender systems.

III. THE PROPOSED APPROACH

The aim is to overcome the issue of data sparseness in recommender system by integrating textual review into CF recommendation. Based on existing literature on sentiment-based recommender systems, most approaches depend on conventional recommender techniques that lack of semantic which lead to major data sparseness problem. Research in sentiment analysis have shown that such information can produce better sentiment accuracy [7] to help in improving the data sparseness problem. Figure 1 illustrates the problem issued by the data sparseness in recommender systems and the proposed solution by integrating the rating items and textual reviews.

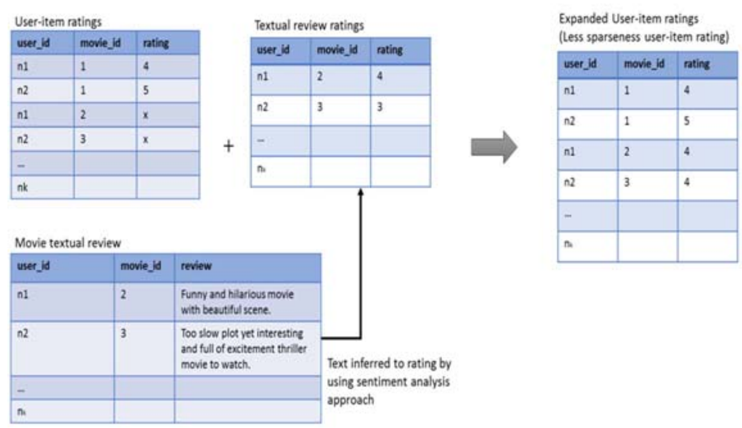

Fig 1. Data sparseness solution by integrating the rating items and textual reviews

Recommender systems are unable to give a good recommendation and accurate information in different domains. However, embedding such enhanced-sentiment approaches were yet to be investigated in sentiment-based recommender systems. Hypothetically, improved accuracy of sentiment results will lead to better performance of recommender systems.

The overall flow process of integrating sentiment analysis in collaborative recommendation is shown in Figure 2.

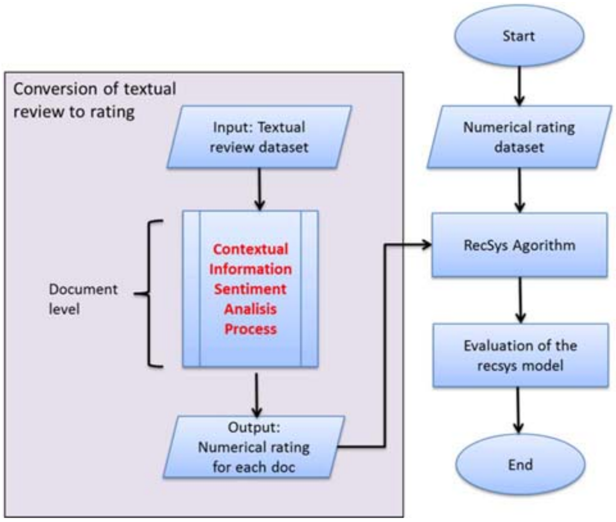

Fig 2. An overall flow process of sentiment analysis integration in collaborative recommendation

Figure 2 depicted an overall flow process of integrating sentiment analysis in collaborative recommendation. There are two major processes which are the conversion of textual review to rating process and integration of sentiment rating to collaborative filtering algorithm. In this experiment, the textual review dataset were taken from the Internet Movie Database (IMDb) which contains of 1000 reviews from Movie Lens.

The process of converting textual review to numerical rating first begin with text processing. Rule-based approach was implemented to extract all the opinion words. The process includes splitting text into sentences, sentence into token, POS tagging, enriching the POS-tagged text with our own tags using dictionaries. These tags are in a different "semantic level" than POS-tags: "positive", "negative", "negation", "incrementer" and "decrementer". Then, some basic extraction rules over the tagged text, in the form of Python functions will be applied. Each review will be converted into a numerical rating by using sentiment analysis technique. Contextual information sentiment analysis process will be applied in order to convert all the opinion words into a numerical rating.

Sentiments obtained from the previous process were integrated into the recommender system framework in order to establish a sentiment-based recommender model. The integration was done by inferring ratings from textual data and then feeding them into a CF algorithm. Evaluation of the sentiment ratings algorithm will then be measured by using the standard evaluation measures which are precision, recall, and accuracy. The result was ranked and presented to users. Evaluation of the sentiment-based recommender system will be measured by the statistical measure root mean square error (RMSE) which was applied during evaluation.

Algorithm 1 describes the process of converting textual review into sentiment rating. The resulting algorithm makes use of four word lists: NegationList, PosList, NegList and IntensifyList. PosList contains all the positive lexicons, NegList contains all the negative lexicons, NegationList contains the inverse lexicons such as “not”, “do not”, “cannot” and others while IntensifyList contains lexicons of increment such as “very” and “extremely” and decrement such as “little”and “decrease”. These opinion lexicons are taken from Bing Liu Lexicons.

Algorithm 1 Pseudo-code algorithm for converting textual review into sentiment rating

```
Input: MovieLens-100K textual review
Output: Sentiment rating review
FOR every review in the MovieLens dataset:
  FOR each word in the review:
     IF the word is in the NegationList
      NegNum = -1
     ELSE:
      Append word to SentimentWordList
      #Reset variables
      Negnum=1
   FOR word in IntensifyList:
       IF word is in review:
        IF review is Inc
          PosCount = PosCount * 2
        IF review is Dec
          NegCount = NegCount / 2
   # Add as Positive review
   if posCount+negCount > accuracy:
          positive ++
            Append Array with sentiment score
   # Add as Negative review
   if posCount+negCount < -accuracy:
        negative ++
            Append Array with sentiment score
```

The collaborative recommendation can be computed as follows:

$$\hat{r}_{ui} = b_{ui} + \sum_{j \in D_c^k(u;i)} d_{ij}\left(r'_{uj} - b_{uj}\right) \quad (1)$$

In equation 1, $\hat{r}_{ui}$ estimates the rating of a user u for an unseen item i. It relies on the bias estimate b_{ui} of the user u for various items j and on a score computed using the k most similar items to i that the user u has already rated. The neighbourhood $D_c^k\ (u;i)$ is the neighbourhood of the k most similar items that the user has already rated or commented and r'_{uj} is the result of the mapping function that accounts for both explicit ratings and those inferred from comments.

Figure 3 depicted a flow diagram of enhancing sentiment analysis approach in recommender system that is further to be implemented.

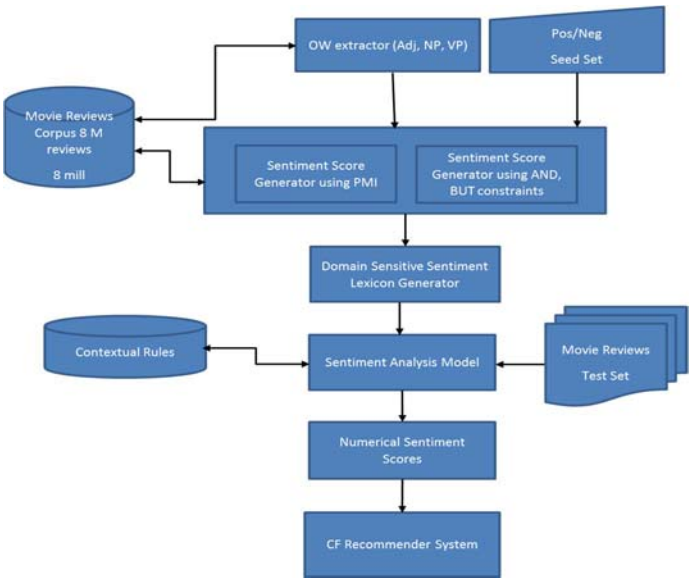

Fig 3. The enhanced sentiment analysis process consisting of the following major components.

As shown in Figure 3, first process is to extract all potential opinion words from the corpus. The focus is only on adjectives, verb phrases and noun phrases. The Stanford SNAP Movie Reviews corpus (https://snap.stanford.edu/data/web-Movies.html) which contains about 8 million movie reviews for this task is used.

In the second process, from the generated OW list, the PMI calculation between the OW and a set of predefined positive and negative seed term is used. Example of positive seed terms are "good", "excellent", "interesting", and "happy", whereas example of negative seed terms are "bad", "terrible", "horrible", and "poor". This would generate a sentiment score for each OW in the range of [+1, -1]. The closer the OW is to one end of the spectrum, the higher its sentiment strength. In the third process, the linguistic rules like "BUT" and "AND" will be applied. The "AND" rule will join two adjectives of the OWs with similar sentiment and the "BUT" rule will join two adjectives with different sentiment. All the occurrences of seed terms with the applied linguistic rules from the corpus were taken, and use these occurrences of the OWs and the SEED TERM to generate a sentiment score. For example OW "AND" POS_SEED_TERM rule will generate a positive sentiment score while OW "AND" NEG_SEED_TERM will generate negative sentiment score. For OW "BUT" POS_SEED_TERM rule will generate negative sentiment score while OW "BUT" NEG_SEED_TERM will generate positive sentiment score. These rules only work on adjectives, so scores for adjective OWs using this technique were only generated.

Next, the scores of adjectives from the second and third process will be combined, and the scores of noun and verb phrases generated in the second process will be used. The generated sentiment lexicon is sensitive to the domain of the corpus. The sentiment lexicon generator combines the score from the two sentiment scores above it. It just takes the average of each score for each word. The contextual rules would be useful to improve accuracy. There are also negations such as "not" and "no", intensifiers such as "very" and "extremely" and diminishers like "slightly", "hardly", "minimally". However these words are optional, but by adding them it would improve the accuracy in the final score. So if a movie reviews corpus was used, the lexicon generated would best perform on a movie reviews test set.

To measure the effectiveness of this model, the accuracy value of the domain sensitivity generated lexicons will be compared to the general lexicon. First, a general domain independent lexicon for example SentiWordNet and Bing Liu Lexicon were used for sentiment classification on the Movie Reviews dataset. Then the generated lexicon can be used to demonstrate it performs better than general lexicons because it was compiled using a corpus from the same domain. The accuracy value for both models will be measured. The Movie Reviews dataset can then be classified. For each review document, the lexicon to generate a final numerical sentiment score for it was used. The contextual information are considered to further improve accuracy including negation such as "not", "no", "never", "haven't", "hadn't", intensifiers and diminisher words which is mostly adverbs such as "very", "extremely", "slightly", "minimally" and others.

Finally, the numerical scores for the movie review dataset documents can be used as the input for CF recommender system, and finally it would predict and recommend movies to the user based on their preference, toward each movie. The algorithm for generating scores of sentiment in textual reviews are describe in algorithm 2 as follows:

Algorithm 2 Pseudocode algorithm for generating sentiment scores from the occurrences of the OWs and the SEED WORDS within the linguistic constraints

```
Input: S0, PRT, NRT, AMR dataset
Output: S1 expanded
Begin
PCS_counter <-0
NCS_counter <-0
For Syn in S0:
    For pos_term in PRTs:
       For each occurrence of "syn AND pos_term"
       OR "pos_term AND syn":
           PCS_counter <- PCS_counter + 1
       END For
     For each occurences of "syn BUT pos_term" OR
     "pos_term BUT syn":
           NCS_counter <- NCS_counter + 1
       End For
     For pos_term in NRTs:
       For each occurrence of "syn AND neg_term"
       OR "neg_term AND syn":
           NCS_counter <- NCS_counter + 1
       END For
     For each occurences of "syn BUT neg_term" OR
     "neg_term BUT syn":
           PCS_counter <- PCS_counter + 1
       End For
     End For
End For
PCS_counter <-0
NCS_counter <-0
Polarity (syn)<-((PCS-NCS)/(PCS+NCS+1*10^-5))
IF Polarity (syn) > 0:
     Polarity (syn) <- "positive"
ELSE IF Polarity (syn) < 0:
     polarity (syn) <- "negative"
ELSE Polarity (syn) < "objective"
END IF
S0 expand <- S1
End
```

In algorithm 2, the final polarity score of the opinion word will then be calculated by dividing the value of subtracting positive and negative score of word with 10^{-5}. The reason of using 10 power of -5 is because if the positive constraint score (pcs) and negative constraint score (ncs) in the denominator are both 0, the program will freeze because of "division by 0 error". So to avoid this, any small number will be added to them to avoid this error. The small number was chosen so it does not affect the result of the final value.

IV. Preliminary Experiments

The experiment was carried out using the movie dataset which contain rating and textual reviews. The baseline dataset used in this experiment was obtained from MovieLens 100k. This data set consists of 100,000 ratings (1-5) from 943 users on 1682 movies. Each user has rated at least 20 movies. This dataset then was expanded by adding the movies with the sentiment rating to them by crawling the IMDb to download user reviews for the remaining movies. Users who have provided fewer than 10 reviews and reviews without user-specified ratings were filtered out to avoid movie duplications that tend to increase data sparseness problem. In the preliminary experiment, the resulting dataset contains approximately 1000 reviews on 1731 movies were added to the baseline dataset, provided by same number of users. The expanded movie dataset now contains 1731 movies. As a result, there were two sets of test been prepared in this experiment which are:

1. Output from basic CF as a baseline
2. Output from basic CF compared to sentiment rating output

The standard measure RMSE is used to evaluate all the two sets of test. Figure 4 shows the preliminary test and evaluation that has been done.

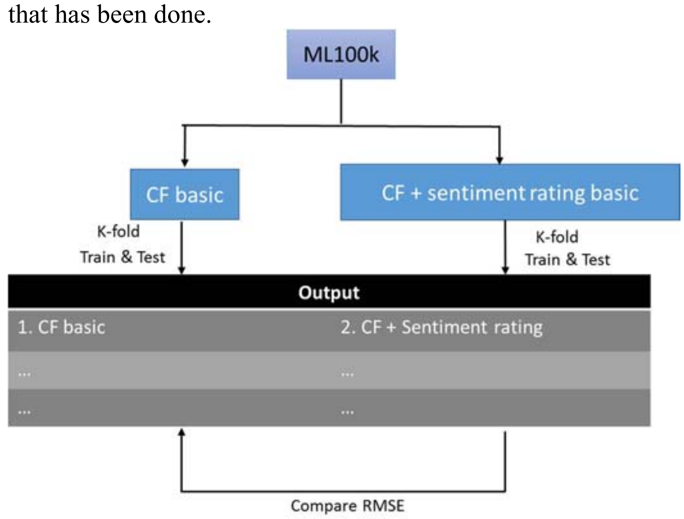

Fig 4. Preliminary test and evaluation

Using the expanded movie dataset of Movie Lens 100k that contains of 1731 movies, the result for the 1st test and 2nd test showed an improvement in RMSE. The results of the tests are depicted as follows:

TABLE I. PRELIMINARY RESULT

Test set	Expanded Movie Lens 100k	
	Sparsity level (%)	User-based RMSE
CF basic without sentiment rating	93.912	2.9802
CF + Sentiment rating	93.901	2.9714

Table 1 shows the result of the preliminary experiment for the first and second test that has been conducted using expanded Movie Lens dataset review. The results show that the value of RMSE for CF combine with sentiment rating give a slightly minimal error compared to CF without sentiment rating applied to them. The sparsity level also decreased from 93.912% of CF basic without sentiment rating to 93.901% of CF with sentiment rating. By analysing the obtained results shows that by applying sentiment based model to collaborative recommendation proven to have a good quality of recommendation and outperform the conventional collaborative recommendation. The experiment will be further carried out using the third test by applying CF with enhancing sentiment rating model to improve the RMSE value and to enhance the performance of recommender system by outperform the both tests.

V. FUTURE WORK

The limitations of the ambiguous word in the different domains can be solved by the help of the contextual information in each domain which will be implemented in our future work. For future implementation, we will overcome the issue of domain sensitivity for situation where there exists an ambiguity word in different domains such as word “unpredictable” in the movie domain gives a positive sentiment for example “unpredictable plot”, while the same word will give a negative sentiment in the car domain, for example “unpredictable steering”. The proposed model will then be combined with our current sentiment based model to gain best RMSE value of recommendation model and minimized the number of data sparsity.

VI. DISCUSSION AND CONCLUSION

The study is important to solve the data sparseness issue in recommender system by overcome the limitation of rating in recommender systems. Recommender systems cannot give a good recommendation in situation where there is no ratings available. By embedding sentiment in recommender systems significantly enhance the recommendation quality of recommender systems. Integration of sentiment in collaborative filtering recommendation able to enhance the problem associated with cold- start. Data-sparsity level also can be minimized. From the preliminary experiment results, we can say that the aim of developing sentiment-based recommendation algorithm that can enhance the performance of recommender systems has been achieved.

ACKNOWLEDGMENT

This research is partially supported by the Malaysia Ministry of Education Grant FRGS/1/2014/ICT02/UKM/01/1 awarded to the Center for Artificial Intelligence Technology at the Universiti Kebangsaan Malaysia

REFERENCES

[1] Leung C. W. et al.2006. Integrating Collaborative Filtering and Sentiment Analysis: A Rating Inference Approach. In Proceedings of of the European conference on artificial intelligence (ECAI ’06) workshop on recommender systems, pp. 62-66.

[2] Peleja, F., Dias, P. & Martins, F. 2013. A recommender system for the TV on the web: integrating unrated reviews and movie ratings. Multimedia Systems, 19(6): 543-558.

[3] Jannach, D., Zanker, M., Felfernig, A. & Friedrich, G. 2010. Recommender Systems: An introduction. New York: Cambridge University Press.

[4] Adomavicius, G. & Tuzhilin, A. 2005. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Trans. on Knowl. and Data Eng. 17(6): 734-794.

[5] Snyder B. and Barzilay R..2007. Multiple Aspect Ranking using the Good Grief Algorithm. Proceedings of NAACL HLT 2007. Association for Computational Linguistic, pp. 300307.

[6] Denecke, K. 2009. Are SentiWordNet scores suited for multi-domain sentiment classification? In: Proceedings of ICDIM’2009. pp. 33-38.

[7] Agarwal B. et al..2015. Sentiment Analysis Using Common-Sense and Context Information. Computational Intelligence and Neuroscience Volume 2015 (2015), Article ID 715730 Vol 15, pp. 1-9.

[8] Miguel Á. García-Cumbreras, Arturo Montejo-Ráez, Manuel C. Díaz-Galiano.2013.Pessimists and optimists: Improving collaborative filtering through sentiment analysis. Journal of Expert Systems with Applications, pp. 6758–6765.

[9] Ghaayen, A. S. & Noah, S. A. M. 2017. Using tags for measuring the semantic similarity of users to enhance collaborative filtering recommender systems. International Journal on Advanced Science, Engineering and Information Technology, 7(6): 2063-2070

[10] Burke, R., Ramezani, M. 2010. Matching recommendation technologies and domains. Recommender Systems Handbook, pp. 367–386. Springer, New York.

[11] Feng, S. et al. 2013. Is Twitter A Better Corpus for Measuring Sentiment Similarity? Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 897–902,

[12] Aggarwal, C. C. 2016. Context-Sensitive Recommender Systems. Recommender Systems Textbook. Springer, NY. pp. 255-281.

[13] B. Agarwal, N. Mittal, and V. K. Sharma. 2014. “Semantic orientation based approaches for sentiment analysis,” in Case Studies in Intelligent Computing—Achievements and Trends, pp. 62–75, CRC Press, Taylor & Francis.

[14] B. Agarwal and N. Mittal. 2014. “Machine learning approaches for sentiment analysis,” in Data Mining and Analysis in the Engineering Field, pp. 193–208, IGI Global.

[15] Alhamid, M. F. et al. 2015. Towards context-sensitive collaborative media recommender system. Multimedia Tools and Applications. Volume 74, Issue 24, pp 11399–11428.