ISSN 2007-9737

Understanding Blogs through the Lens of Readers' Comments

Abhilasha Sancheti[1], Natwar Modani[1], Gautam Choudhary[2], Chintha Priyadarshini[3], Sai Sandeep Moparthi[4]

[1] Adobe Research, Big data Experience Lab, Bangalore, India

[2] Indian Institute of Technology, Department of Computer Science, Roorkee, India

[3] Indian Institute of Technology, Department of Mathematics, Kanpur, India

[4] Indian Institute of Technology, Department of Computer Science, Bombay,India

{sancheti, nmodani}@adobe.com, {gc.iitr, priyadarshinichintha, moparthi.saisandeep}@gmail.com

Abstract. In order to keep their audience engaged, authors need to make sure that the blogs or articles they write cater to the taste of their audience and are understood by them. With the rapid proliferation of online blogging websites, the participation of readers by expressing their opinions and reviews has also increased in the form of comments on the blogs. These comments are valuable source for the authors to understand how their audience are perceiving their blogs. We believe that associating comments to the specific part of the blog they refer to will help author in getting insights about parts of the blog which are being discussed and the questions or concerns that readers have about those parts. Moreover, categorizing these comments will further aid the author in imbibing the comments. In this work, we describe a method to associate comments to the specific parts of the blog and introduce a hierarchical way to categorize the comments as Suggestion, Agreement, Disagreement or Question.

Keywords. Classification, comments association, support vector machine, feature selection.

1 Introduction

Social blogging platforms allow authors to share information, their personal experiences and opinions on a wide variety of topics. These platforms also allow readers to leave their comments on the blog. The readers might agree or disagree with the author, provide suggestions for improving the blog or might have some questions. The authors need to make sure that the blogs they write cater to the taste of their audience and are understood by them so that they remain engaged. The primary mechanism for an author to understand how her audience are reacting to the blogs is to understand the comments they write for the blog. For the popular blogs, the number of comments can be large. Hence, if the author has created many blogs, reading all the comments, understanding the commenter's reaction, and which parts of the blog (called blog segment) need attention is a tedious and time-consuming task. Understanding what parts of the blog the audience did not understand (and had questions about), what parts did they agree/disagree with, and where did they feel the need to make changes to the blog can enable the authors to write the blogs in a more engaging way in future or to improve the existing blogs. In absence of a tool support, the authors do not fully benefit from insightful comments made by the readers to improve the content of their future blogs (or make changes to existing ones).

In this work, we present a method for automatically associating the comments to the part of the blog they refer to and classifying the comments as Suggestion, Agreement, Disagreement or Question. Additionally, we provide a visual representation of this information, so that authors can quickly understand the type and strength of user reactions generated by the various parts of their blogs. This analysis can also help a reader in determining what blogs (and which parts of these blogs) should they read depending on their interests and roles. In today's world, where information overload is a major challenge, such insights can help them become more productive.

In Section 2 we detail the prior explorations in the field of comment association and classification. In Section 4 we describe the methodology in detail and present the results in Section 5 followed by the conclusion.

2 Related Work

To the best of our knowledge, there is no existing work which talks about comment association and classification simultaneously. Moreover, there is no work which classifies comments in the set of categories that we are considering, though there are different works which have looked at these categories separately.

2.1 Classification

Classification of user comments is a well-researched area. People have come up with various classification schemes for YouTube video comments [17], product reviews [3], tweets [5], etc. Most of these schemes talk about the type of sentiment, emotion [16] or mood expressed in the comment. There are a few works on identification of spam, off-topic, obscene, toxic and abusive [2] comments made on online blogs or YouTube videos [1].

[1] `https://pdfs.semanticscholar.org/65fb/992b712d75c6499d8649d53ad575bdef9e0e.pdf?_ga=2.181395107.541616974.1532947452-1106483369.1517137894`

There are also a few works which focus on the semantics or content aspects of the short texts. There are prior explorations [21, 22] on advice mining from the web forums which introduce various linguistic features which can be used to identify advices.

We leverage these features to classify sentences as suggestion. [22] proposed a hidden Markov model for labelling sequential sentences as advice revealing or not and use syntactic, semantic and contextual features for their task. Our task is different from theirs since their task involves independent comments and not sequential sentences.

With respect to the agreement and disagreement categories, there have been previous work on recognizing disagreement in informal political arguments [1] and identifying agreement and disagreement in the social media dialogues [11]. Among these work, [1] show that use of contextual and dialogue features improve accuracies as compared to unigrams. Topic independent features improve the performance of agreement-disagreement classification over unigrams as demonstrated by [11].

Apart from this, stance detection in tweets is also similar to identifying agreement and disagreement in text with respect to a part of the blog. A set of structural, contextual, sentiment and label-based features for predicting stance towards a mentioned target are defined in [9]. However, above mentioned approaches will not directly work in our case because of the differences in domain and also the dataset under consideration.

While there has been significant work on classifying short text, some of which also address comment classification, we are not aware of any work which classifies the comments in multiple classes, and in particular, the classes we are taking into consideration (agreement, disagreement, questions and suggestions).

We hypothesize that comments belonging to these classes will provide constructive and valuable insights about the blog to its author and other readers.

2.2 Association of Comments

There have been some research in the area of associating comments to a part of the news story and aligning comments to the news topics. An unsupervised technique is proposed in [18] which takes cosine similarity of LDA, SS-PLSA and BOW features of both comment and the segment of the news article to align comments with the segments. [19] propose a supervised technique for the task of alignment and show that the structured learning approach performs better than the other unsupervised and binary classification approaches. Frameworks for aligning comments to news topics by automatically extracting topics from a given news article and its associated comments are described in work like [7, 6]. However, in this work we position the comment association task as a 'question-answering' task where comment is considered as a query and the different parts of the blogs as the answer.

3 Problem Definition

The problem that we are trying to solve can be stated as: How to help authors and readers in extracting useful insights from the comments on a blog to:

— Help authors in understanding the audience reaction and improve upon her writing in future

— Help readers in understanding the blog through comments, possibly to prioritize the reading

In this work, we aim to solve the following sub-problems which define the useful insights that we will be presenting to the authors/readers.

1. Understanding the scope of the comment with respect to the blog. By this we mean associating the comment to the segment(defined at the level of a paragraph) of the blog it is referring to.

2. Understanding the type of the comment. We are considering the following types:

   — **Agreement:** comments that support (parts of) the content of the blog.

   — **Disagreement:** comments that contradict/disapprove of (parts of) content in the blog.

   — **Suggestion:** comments that advise or suggest changes to the content or suggests some alternatives to what's present in the blog.

   — **Question:** comments that capture queries or doubts about (parts of) the content in the blog.

3. Visual representation of the comments related information for better insights

We hypothesize that the classes under consideration are exhaustive (after the removal of irrelevant comments) because there could be two scenarios in which reader can comment (1) reader does not understand the blog, and (2) the reader understands the blog. In scenario (1) he/she will have doubts with respect to the content of the blog and thus his/her comments will belong to the 'Question' category. In case (2) he/she will either have some suggestions for improving the content or will agree/disagree with the content. There might also be some comments, which go off-topic or are general point of views like "I don't like traveling alone" but we are not considering them since they can always be preprocessed and filtered out.

4 Methodology

In this section, we describe the dataset, the features, and the techniques used for the classification and the comment association task.

4.1 Data Preparation

We are not aware of previous work which simultaneously tackles the tasks of comment classification and its association with the part of the blog. Therefore, we curated a dataset to suit our purpose with the help of human annotators.

We collected a total of 90 blogs from different online blogging websites and asked the annotator to write comments on the presented blog indicating which part of the blog it corresponds to and the category (suggestion, question, agreement, disagreement) it belongs to.

We had a total of 271 comments with 95 in Question, 70 in Disagreement, 674 in Agreement and 39 in Suggestion category. Since the size of our corpus was small we used auxiliary datasets for the training the classifiers individually. We used Open Domain Suggestion Mining dataset [12] for the suggestion classification task. This dataset consists of tagged sentences from various domains such as electronics reviews, hotel reviews, customer service reviews, and travel forums. We used a subset of Internet Argument Corpus (IAC) [20] for the agreement-disagreement classification task. This dataset consists of pairs of the kind (quote, response), where quote is the base sentence and the response either agrees or disagrees with the quote. There was a huge class imbalance in this data and thus we downsample sentences with disagreement label to get a balanced dataset.

4.2 Classification

We build separate classifiers for Suggestion, Question and Agreement/Disagreement for the training purpose and propose a hierarchical approach to classify the given test comment into one of the possible categories. The categories under consideration are mutually exclusive in our case and thus each comment can belong to only one category.

4.2.1 Suggestion Classifier

We train a binary SVM classifier for classifying the comment as suggestion or non-suggestion. Our main contribution is in feature engineering. We consider the following set of features for this classification task and show the performance improvement results in the next section.

1. Clue Words (Clue): We curated our own list of clue words (such as suggest, recommend, advice, urge, request, etc.) which were selected by investigating the training data. A binary feature vector of dimension equal to the number of clue words is created where the value corresponding to each dimension denotes the presence or absence of a clue word in the comment.

2. Modal Verbs (MV): [21] has shown that advice revealing sentences often expresses modality which are expressed using the modal verbs (such as can, could, might, should, would, etc.). We define a set of modal verbs and a binary feature corresponding to each of the modal verb, indicating the presence or absence of the modal verb in the comment.

3. Imperative Mood Expressions (IME): [21] found that sentences containing imperative mood expressions (such as ‘do not bring mobile phones’, ‘it is a good idea to add more experimental results’) result in the actions in certain ways. We also used this feature in order to characterize the suggestions. We used the same heuristic method as defined by [21] for finding value of this feature. The heuristic says that if the verb present in the comment is not preceded by a subject, then most likely the comment contains an imperative mood expression.

4. Typed Dependencies (TypDep): We leveraged this feature as defined by [22]. We considered only conjunct, clausal subject, and nominal subject relations, which are denoted by “conj”,“csubj”, and “nsubj”, respectively in the comment’s parse tree obtained using Stanford Dependency Parser [4].

5. Informativeness Score (InfScore): It is the summation of the tf-idf score of all the words in the comment:

$$\sum_{w_i \in C} TfIdf(w_i),$$

where C is the comment and w_i is the word in the comment.

The training was done using the auxiliary dataset mentioned in above subsection. We used radial basis function (rbf) kernel while training the SVM.

4.2.2 Question Classifier

We consider identification of question as a binary classification task. We use Stanford parser [10] for obtaining the parse tree of the given sentence and check for the presence of either of the two tags, namely SBARQ and SQ in the tree.

1. SBARQ: Presence of this tag indicates the presence of a direct question introduced by a wh-word or a wh-phrase.

2. SQ: Presence of this tag indicates the presence of an inverted yes/no question, or main clause of a wh-question, following the wh-phrase in SBARQ.

We mark the given sentence as belonging to a Question category if either of these tags is present in the parse tree of the comment.

4.2.3 Agree-Disagree Classifier

We train a binary SVM classifier in order to classify the sentences into agreement and disagreement category. Presence of agreement or disagreement depends upon the context and thus for this classification task we also consider the segment of the blog that the comment is associated with. Each of the features we experimented with is explained below.

1. LIWC [14]: These features are used by [1] for recognizing disagreement in political arguments. We hypothesize that use of linguistic features will help in identifying the agreement and disagreement. We calculate the LIWC features for both the comment and the part of the blog it belongs to.

2. Glove Embedding (Glove) [15]: This feature represents the semantics of the comment. The feature value is a 50-dimensional vector obtained by the summation of the 50 dimensional embeddings of all the words present in the comment.

3. N-grams: We curated our own list of n-grams after investigating the data. These n-grams characterizes the presence or absence of agreement in the comment. For each n-gram, a binary value is assigned depending upon the presence or absence of that n-gram in the comment.

4. Positive and Negative sentiment words (Pos-Neg): We leverage the positive and negative sentiment words curated by [8] for identifying the polarity of product reviews to classify the comment as agreement or disagreement. We consider the difference in the number of positive and negative sentiment words present in the comment as the feature.

5. Positive Sentiment words (Pos): This feature denotes the number of positive sentiment words present in the comment.

6. Negative Sentiment words (Neg): This feature denotes the number of negative sentiment words present in the comment.

7. Afinn score (Affin) [13]: It gives a word polarity score between -5 to +5. The feature value is the summation of the afinn score of all the words in the comment. The same feature is also calculated for the part of the blog that the comment belongs to.

The training was done using the auxiliary dataset mentioned in the previous subsection. We used radial basis function (rbf) kernel while training the SVM. The model performed best with just the N-gram features.

4.2.4 Hierarchical Classifier

Given a comment, we propose a hierarchical approach to classify it into one of the classes we are taking into consideration. We use the individually trained classifier models as described in the previous section with the features as per the best performing models.

Figure 1 shows the workflow of this classifier. The comments first go through the Suggestion classifier which if predicted as a suggestion, are classified as suggestion and not passed through

ISSN 2007-9737

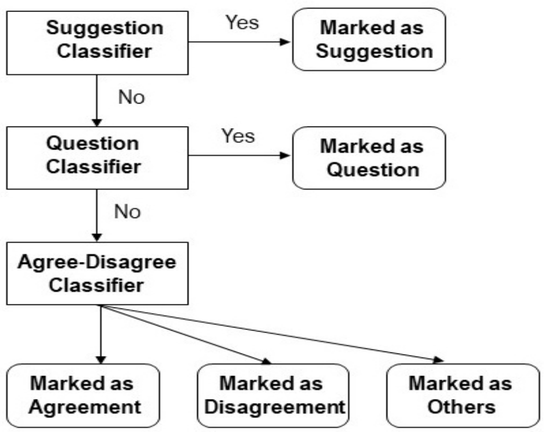


Fig. 1. Hierarchical Classification Workflow

any of the other classifiers. All the comments which are not classified as suggestions are fed to the Question classifier. If the comment is predicted as a question, then we label it as a question and pass the rest of the comments through the Agree-Disagree classifier.

This classifier finally classifies all the remaining comments into agreement or disagreement.

We believe that some suggestions like "Did you try deep Learning approaches?" are written in question form and will get mis-classified if they are first passed through the question classifier. Thus, we chose this hierarchy.

4.3 Association of Comments with Segment of the Blog

We model the association task as a 'question-answering' problem and use Learning to Rank models to rank different segments given a comment. We consider comment as the query and the segments of the blog as the answers.

Given a query the model ranks the answers. We used RankLib library's ListNet[2]model for this purpose.

We use the following lexical and semantic features for our purpose:

[2]https://sourceforge.net/p/lemur/wiki/RankLib/

Lexical Features:

1. Segment Length (SegLen): Number of terms in the segment
2. Segment Position (SegPos): Relative position of the segment with respect to the blog
3. Exact Match (EMatch): It is a binary feature indicating whether the comment is a substring of the segment
4. Term Match (TMatch): Number of terms that are common in the comment and the segment
5. Synonym Match (SMatch): It is the fraction of comment's terms whose synonym is present in the segment
6. Language Model(LM): It is a score which is computed as the log likelihood of the comment being generated from the segment

Semantic Features:

7. Word2Vec Similarity (W2V): It is the cosine similarity score between the summation of the word2vec embeddings of the words in the comment and the segment
8. Universal Sentence Embedding (USE) Similarity: It is the cosine similarity score between the USE embeddings of the comment and the segment

5 Evaluation and Results

We evaluated comment association task on the following two metrics.

1. Mean Reciprocal Rank (MRR): It is given by:

$$\frac{1}{|C|}\sum_{i=1}^{|C|}\frac{1}{rank_i},$$

where C is the set of the comments that are queried for association and $rank_i$ refers to the rank position of the correctly associated segment.

2. Percentage accuracy: It is the ratio of comments correctly associated to the total number of comments queried for the association.

Table 1 shows the results of the task of associating the comments with the segment of the blog. It can be inferred from the results that use of Word2Vec embeddings (W2V) is bringing down both MRR and accuracy values as compared to the case when it is not used. However, use of Universal Sentence Encoding (USE) feature along with all the other lexical features improves both MRR and accuracy values. There is an appreciable improvement in the metric values when only Term match (TMatch) feature is used.

We evaluated our individual classifier models and the hierarchical classifier model on Precision, Recall and F1 score metrics.

Table 2 presents the results from the Suggestion classifier. We can clearly see that there is a significant improvement in recall and f1-score when clue words (Clue) are used along with modal verbs (MV) and imperative mood expressions (IME). When clue words (Clue) and modal verbs (MV) features are used along with typed dependency (TypDep) and informativeness score (InfScore) features the precision score increases at the expense of recall. Finally using all the features together shows significant increase in the recall and f1 score values, indicating that all these features together help in identifying the suggestive characteristics of the comments.

Our model was able to identify the question with a precision of 0.86 and recall of 0.73 with F1-score being 0.79.

Table 3 presents the results of the agree and disagree classifier. We can see that using the N-grams provides good performance trade-off.

Table 4 presents the results of the hierarchical classifier. It is evident from the metric values that with the hierarchical classification the precision, recall and f1 score improves since once classified by one classifier as positive, the comment is not passed to the next classifier.

As one can see, the results of classifier are satisfactory to provide useful insights, even though there is room for improving these classifiers.

5.1 Visualization

We built a mobile app to present these insights about the comments and the blog to the author and the readers.

Figure 2a shows the landing page of the app where the author/reader can see the list of the blogs he/she has written or can read. The doughnut chart on the right side of each blog shows the category-wise distribution of the comments made on that blog and the number inside the chart is the total number of comments made on that blog. The legends in the top bar shows the class represented by each colour in the chart.

Figure 2b presents the view when a particular blog is chosen. The blog is partitioned into the segments demarcated by the blocks. The number in parenthesis besides each category is the count of comments on the blog that belongs to the category. The scroll bar is segmented according to the segments in the blog and the colour represents the dominant comment category for that segment.

Figure 2c shows the view of blog when the author/reader click on the scroll bar which takes him/her to the corresponding segment where the list of the comments and the category they belong to can be seen.

The author(or reader) can also look at the comments on the blog belonging to only a particular category by clicking on that category from the top legend bar. It can be seen from Figure 2d that all the segments coloured yellow have comments from question category.

As one can see, our tool can help the authors and readers by providing insights about what type of reactions/comments a given blog is attracting, and which parts of the blogs are responsible for those reactions. One can consider enabling several features using such information, for example, sorting the blogs based on most or least number of comments of a particular type (agreement/disagreement/question/suggestion). Also, one may directly find the segments of blogs which attract specific type of comments (and need not open the blogs individually to see them). One can also create summaries of comments of a given type for a particular segment of blog.

Table 1. Results of the Comment Association Model

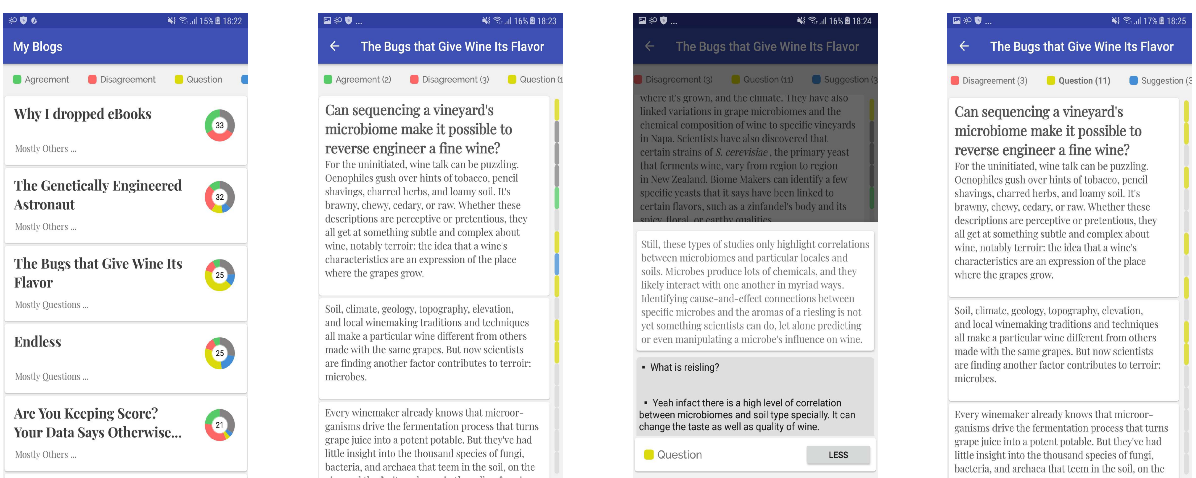
Features	MRR	Accuracy
SegLen+SegPos+EMatch+TMatch+SMatch+LM+W2V	0.745	0.631
SegLen+SegPos+EMatch+TMatch+SMatch+LM	0.763	0.675
SegLen+SegPos+EMatch+TMatch+SMatch+LM+USE	0.769	0.692
TMatch	0.849	0.798

Table 2. Results for Suggestion Classifier

Features	Precision	Recall	F1 score
Clue	0.44	0.18	0.25
Clue+MV+IME	0.46	0.64	0.54
Clue+MV+TypDep+InfScore	0.48	0.59	0.53
Clue+MV+IME+TypDep+InfScore	0.47	0.62	0.53

Table 3. Results for Agree-Disagree Classifier

Features	Precision		Recall		F1 score	
	Agree	Disagree	Agree	Disagree	Agree	Disagree
LIWC	0.54	0.55	0.43	0.65	0.48	0.60
Glove+N-grams+PosNeg	0.63	0.65	0.63	0.65	0.63	0.65
Glove+N-grams+Pos+Neg	0.63	0.66	0.64	0.65	0.64	0.66
Glove+N-grams+Affin	0.67	0.66	0.60	0.72	0.63	0.69
N-grams	0.65	0.79	0.84	0.58	0.73	0.67



(a) (b) (c) (d)

Fig. 2. Different Visualization of the Mobile App

Table 4. Results for Hierarchical Classifier

Class	Precision	Recall	F1 score
Suggestion	0.47	0.62	0.53
Question	0.95	0.75	0.84
Agreement	0.63	0.85	0.72
Disagreement	0.76	0.50	0.60

Such summaries are more useful as they would be talking about the same thing in same manner, and hence the possibility of creating a coherent summary is higher.

6 Conclusions

This paper presents a way to leverage the information present in the comments and deliver insights to the authors about the audience's reaction to the content at a granular level. These insights will help the author in understanding the audience and improving upon the future content. Readers can also benefit from the comments as they help in understanding the blog and possibly help in prioritizing which blog (and which parts of it) they should read.

References

1. **Abbott, R., Walker, M., Anand, P., Fox Tree, J. E., Bowmani, R., & King, J. (2011).** How can you say such things?!?: Recognizing disagreement in informal political argument. *Proceedings of the Workshop on Languages in Social Media*, Association for Computational Linguistics, pp. 2–11.
2. **Chu, T., Jue, K., & Wang, M. (2016).** Comment abuse classification with deep learning. *Von https://web. stanford. edu/class/cs224n/reports/2762092. pdf abgerufen*.
3. **Dang, Y., Zhang, Y., & Chen, H. (2010).** A lexicon-enhanced method for sentiment classification: An experiment on online product reviews. *IEEE Intelligent Systems*, Vol. 25, No. 4, pp. 46–53.
4. **De Marneffe, M.-C., MacCartney, B., Manning, C. D., et al. (2006).** Generating typed dependency parses from phrase structure parses. *Proceedings of LREC*, volume 6, Genoa Italy, pp. 449–454.
5. **Go, A., Bhayani, R., & Huang, L. (2009).** Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford*, Vol. 1, No. 12.
6. **Hou, L., Li, J., Li, X., Qu, J., Guo, X., Hui, O., & Tang, J. (2013).** What users care about: A framework for social content alignment. *IJCAI*, pp. 1401–1407.
7. **Hou, L., Li, J., Li, X.-L., Tang, J., & Guo, X. (2017).** Learning to align comments to news topics. *ACM Transactions on Information Systems (TOIS)*, Vol. 36, No. 1, pp. 9.
8. **Hu, M. & Liu, B. (2004).** Mining and summarizing customer reviews. *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, pp. 168–177.
9. **Lai, M., Farías, D. I. H., Patti, V., & Rosso, P. (2016).** Friends and enemies of clinton and trump: using context for detecting stance in political tweets. *Mexican International Conference on Artificial Intelligence*, Springer, pp. 155–168.
10. **Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., & McClosky, D. (2014).** The Stanford CoreNLP natural language processing toolkit. *Association for Computational Linguistics (ACL) System Demonstrations*, pp. 55–60.
11. **Misra, A. & Walker, M. (2017).** Topic independent identification of agreement and disagreement in social media dialogue. *arXiv preprint arXiv:1709.00661*.
12. **Negi, S., de Rijke, M., & Buitelaar, P. (2018).** Open domain suggestion mining: Problem definition and datasets. *arXiv preprint arXiv:1806.02179*.
13. **Nielsen, F. Å. (2011).** A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903*.
14. **Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001).** Linguistic inquiry and word count: Liwc 2001. *Mahway: Lawrence Erlbaum Associates*, Vol. 71, No. 2001, pp. 2001.
15. **Pennington, J., Socher, R., & Manning, C. (2014).** Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543.
16. **Savigny, J. & Purwarianti, A. (2017).** Emotion classification on youtube comments using word embedding. *Advanced Informatics, Concepts, Theory, and Applications (ICAICTA), 2017 International Conference on*, IEEE, pp. 1–5.

17. **Severyn, A., Moschitti, A., Uryupina, O., Plank, B., & Filippova, K. (2014).** Opinion mining on youtube. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pp. 1252–1261.

18. **Sil, D. K., Sengamedu, S. H., & Bhattacharyya, C. (2011).** Readalong: reading articles and comments together. *Proceedings of the 20th international conference companion on World wide web*, ACM, pp. 125–126.

19. **Sil, D. K., Sengamedu, S. H., & Bhattacharyya, C. (2011).** Supervised matching of comments with news article segments. *Proceedings of the 20th ACM international conference on Information and knowledge management*, ACM, pp. 2125–2128.

20. **Walker, M. A., Tree, J. E. F., Anand, P., Abbott, R., & King, J. (2012).** A corpus for research on deliberation and debate. *LREC*, pp. 812–817.

21. **Wicaksono, A. F. & Myaeng, S.-H. (2012).** Mining advices from weblogs. *Proceedings of the 21st ACM international conference on Information and knowledge management*, ACM, pp. 2347–2350.

22. **Wicaksono, A. F. & Myaeng, S.-H. (2013).** Automatic extraction of advice-revealing sentences foradvice mining from online forums. *Proceedings of the seventh international conference on Knowledge capture*, ACM, pp. 97–104.

Article received on 06/02/2019; accepted on 04/03/2019.
Corresponding author is Abhilasha Sancheti.