IADIS International Journal on Computer Science and Information Systems
Vol. 8, No. 2, pp. 147-158
ISSN: 1646-3692

PERSONALIZED SUMMARIZATION OF CUSTOMER REVIEWS BASED ON USER'S BROWSING HISTORY

Zehra Kavasoğlu. *Istanbul Technical University Istanbul/Turkey*

Şule Gündüz Öğüdücü. *Istanbul Technical University Istanbul/Turkey*

ABSTRACT

Every e-commerce web site today has the product review feature which allows customers to express their opinions and comments about the product they have purchased. These comments are important for potential customers when deciding which product to buy. However, reading large amounts of customer reviews available for each product is a time consuming process. For this reason, customers usually tend to read small pieces of topmost comments and skip the rest of them. Also, depending on personal preferences and needs, customers might be interested in different features of various products. Therefore, a feature based summarization of the products is very helpful for potential customers in selecting the best product option. Existing feature based review summarization methods create a product summary for a common user profile ignoring the individual preferences. In this paper, we propose a novel feature based approach for personalized review summarization by giving importance to potential individual customer preferences. In order to evaluate our method, a dataset has been collected from a popular Turkish e-commerce web site. The experimental results show that our method is successful in finding and summarizing the most relevant reviews for the active user.

KEYWORDS

Review Mining, Personalization, Feature Based Summarization

1. INTRODUCTION

With the growing popularity of internet, e-commerce web sites are taking more and more places in our lives. Nowadays, a growing number of people are shopping online. Using the product review feature of e-commerce web sites, these customers are submitting comments and declaring their opinions about the products as well as indicating satisfaction with the products. These product reviews are the primary reason for the increasing numbers of online shopping customers, since they have an important impact on the decision of the customers which product to choose. According to eMarketer (http://www.bazaarvoice.com/social-commerce-statistics), customer reviews are 12 times more reliable for the potential customers than original product descriptions provided by the manufacturers.

The product reviews assist consumers to decide the best product that meets their needs. Using the product review feature, each customer is able to post different comments about different specifications of a product. However, the reviews reflect the personal judgments of the customers because their requirements and the expectations might differ in many ways. Therefore, a review is totally subjective and it provides important personal feedback about the product. Also, usually customers do not have time and desire to read every single product review that is available. At this point, summarization techniques become very useful in showing the general idea of the reviews. To get the characteristics of customer behavior, the reviews should be sentimentally analyzed in order to determine the positive/negative sides of the product. Several previous studies on feature based summarization overcome this issue by summing up the reviews for a common user profile. Nevertheless, interests and needs are different for each customer and a potential customer is eager to make use of the reviews that are addressing his/her personal interests and needs when selecting the most suitable product option. Thus, reviews should be filtered according to the personal preferences of the potential customers and feature based summarization should be directed by personal preferences.

There are many studies on feature based summarization, but in these studies the personal preferences are ignored and the main goal is to summarize the reviews for an average user. Personalization is taken into account generally on text mining approaches. News filtering according to personal preferences is investigated in some works (Wu et al. 2011, Katakis et al. 2009). Personalization on review mining is also researched in a recent study on personalized recommendation of user comments (Agarwal et al. 2011). However, these common text mining approaches do not meet the requirements on product review mining as in the feature based investigations. To the best of our knowledge, there are no studies available related to personalization on feature based investigations. Feature based systems usually take specific model comments and explore its feature-sentiment relations. Yet, this approach is not sufficient for a potential customer. If a customer is looking for a phone, he/she wants to examine all the models available and compare the reviews based on his/her personal preferences. Reading only the reviews about the features that are related to customers' personal preferences can help the potential customer make the right decision and save time for finding the valuable information from vast amount of reviews.

In this paper, we propose a novel method for personalized review summarization on an e-commerce web site. Our method can be summarized as follows:

1. Extracting common features of the products from click-through pages of the current user
2. Finding desired products of a current user based on extracted common product features
3. Finding reviews that are more related to the products in search and user needs
4. Identifying product features on the reviews
5. Identifying positive/negative opinions on the reviews
6. Generating feature-opinion pairs to understand the related sentiment of a feature
7. Producing a summary of these feature-opinion pairs

Our method is different from traditional feature based summarization in a number of ways. First of all, we use the search log history of users in order to extract user preferences for personalization purposes. Second, our method has a shorter runtime since the summarization is performed on filtered relevant reviews. And lastly, while identifying product features, we use the product features taken from the web site and this can be evaluated as supervised method on feature extraction.

We evaluate our method using a real dataset obtained from a popular Turkish e-commerce web site (hepsiburada.com) using ROUGE toolkit. The experimental results show that our method outperforms feature based review summarization methods in finding reviews related to the personal preferences of the user.

The rest of the paper is organized as follows: Related Works are summarized briefly in Section 2. Our proposed method "Personalized Review Summarization" and system architecture are presented in Section 3. We describe the experimental dataset and the results in Section 4. Finally, we conclude in Section 5.

2. RELATED WORK

To the best of our knowledge, there exists no personalization study on product review summarization. Our proposed approach is a new method making use of feature based review summarization together with the application of personalization techniques. For this reason, this section will review the key studies about personalization and existing works on feature based review summarization.

Personalization methods are commonly used by search engines to get the correct search results with minimum trials. The search engines try to find related web pages based on the query words entered by the user and rank these web pages according to their relevance. By clicking through the result pages, people try to find information about the topic expressed in query words. At this point, click-through pages could be used to establish a relationship between the page contents and the query words. The information from click-through data is used in query expansion to get more accurate results (Xue et al. 2004). Sun et al. (2005) also use click-through data on web page summarization. They aim to find co-occurred words from matched search queries and click-through pages, and then direct their page summarization according to these common words.

Feature based review mining is different from traditional review mining since it is based on capturing the opinions from the reviews related to the product features. Traditional review mining is able to produce just a brief insight from all of the reviews. When we think about the different needs for different customers, brief insight would be inefficient; we need detailed information based on the product features. Existing methods on feature based review mining could be grouped as Latent Dirichlet Allocation(LDA) based methods and bootstrapped lexicon methods.

LDA based methods are applied to the feature based summarization. Yang and Datta make rated aspect summarization in two steps: discovering the aspects by semi-supervised LDA, predicting and aggregating the ratings for each aspect and then summarizing the results. For each aspect, displaying its overall rating is generally useful. But, they do not show the related sentences. Multi-grain LDA could be also used as stated in this work (Titov and McDonald 2008). Topics would be investigated as aspects. They use star ratings to specify the sentiment of each aspect. This work assumes that there is a rating for each aspect, this is not generally valid, and also our system does not include these ratings.

Bootstrapped lexicon methods are commonly used in feature based summarization. One of these works belongs to Zhuang et al. (2006) on movie domain. They start with specific opinion words and take their synonyms/antonyms from WordNet. Then, they find the feature-opinion pairs using grammatical dependency and summarize the results. This work is specified just for one domain "movie" and does not address all of the domains. They use grammatical dependency in the sentence. Yet, for some languages, no such sentimental relations are defined for the sentence. The advantage of this work is its well defined output.

Another study is done on local service reviews (Blair et al. 2008). They take the reviews from a service of interest, use bootstrapped lexicon with WordNet for opinion detection, update the polarity scores for each word to find out how positive or negative this opinion and make summarization related to the features. This work is also close to our work and aims to produce a generic summarization method compatible with all of the services. Defining different polarity scores for the words is also an advantage of this work. But, this method is computationally complex and therefore the implementation of this method is much more difficult than other bootstrapped lexicon methods.

Some works consider personal preferences in summarization. Agarwal et al. (2011) work on finding similarities between users according to their previous entries and their ratings to the comments. They try to rank the comments according to their preferences. In our case, it is possible that the user is not logged in which means that the user has no entries and comment ratings available. So, their work is not applicable for such a case. Other work tries to filter the news related to the user preferences (Wu et al. 2011). The system has the personalized interest keywords of the users and general category of keywords like sports, politics, economics etc and tries to get news that have overlapped keywords in both personal data and news page.

Hu et al. (2004) find frequent product features on which the customers have expressed their opinions and identify whether each opinion sentence is positive or negative. The advantage of this work is that the method is easily applicable to all of the domains.

In this paper we propose a novel personalized review summarization method. Our approach utilizes the web log history of a user to find out which reviews are more suitable for his/her needs. We are filtering the reviews according to the needs of a potential customer and we are not summarizing all of the available reviews. This makes our system faster and show more related reviews for a customer. Both our product feature identification and review summarization methods are different from and more effective than the earlier works when generating a summary of related reviews.

3. PERSONALIZED REVIEW SUMMARIZATION

The personalized review summarization system is designed as three main components: the preprocessing component, the personalization component and the summarization component. The architectural overview of our system is given in Figure1.

Two data sources are used in this study: user web log history and the reviews obtained from an e-commerce web site. The preprocessing component is responsible to conduct preprocessing operations on both data sets. The personalization component's role is to identify users' needs in order to retrieve related reviews. The summarization component extracts summaries from related reviews by first applying POS tagging operations. Then, the opinions of users related to the product features are identified by this component. The details of the system are given by the following subsections.

3.1 Preprocessing Component

In this work, it is assumed that web log data is available. The web log data has search query words and a set of click-through pages. Since the reviews and search query terms are entered by users, they might be misspelled or they might include stop words. So, for both data sources, stop words removal and correction of misspelled words are necessary as text processing operations and firstly done in this component. Then, we continue operations on user web log data. The web log data is represented as "Session" (Xue et al. 2004) entities. Because a page represents a product model, we can specify the session entity (*S*) as search query (q) – model (*m*) pairs, and the number of sessions is specified by (*n*):

$$S = \{(q_1, m_1), \dots, (q_n, m_n)\} \qquad [1]$$

The terms existed on the search queries could be the key points to the customer needs. Each query has a set of terms (*t*). We keep terms in query words with their frequencies to understand their importance:

$$q = \{\ \big(t_1, freq(t_1)\big), \big(t_2, freq(t_2)\big), \dots\ \}$$

And the click-through pages, user has sequentially clicked on, could be used to identify the user interests.

Each page defines a specific model and each model has features (*fList*) and user reviews (*R*), *f* and *r* specify a feature and a review respectively, and then we can define the model as:

$$m = \{fList, R\},\ \ R = \{r_1, r_2, \dots\},\ \ fList = \{f_1, f_2, \dots\}$$

Product features, i.e., attribute name-value pairs, are also extracted from Web pages: $f = \{(key, value)\}$ where *key* defines the feature description and *value* defines its value.

In our method, we have used this model information to understand which product features are important to the customer. In this work, it is assumed that if some product features exist in most of the click-through pages of a user than these features may be important for that user when selecting the product. We regard the features existing on more than half of the click-through pages as common features (*cfList*) and specify the number of visited models with *n* as stated in [1]:

$$cfList = \{f_1, f_2, \dots\}\ ,\ \forall f \in cfList : \alpha(f)$$

$$\alpha(f) = \left(\sum_{i=1}^{i=n} \gamma(i,f)\right) > \left({}^{n}\!/_{2}\right)\quad ,\quad \gamma(i,f) = \left\{ \begin{array}{l} f \in fList_i \ \rightarrow\ 1 \\ otherwise\ \rightarrow 0 \end{array} \right\}$$

We also keep features with their frequency values:

$$f = \{\ \big(f\ , freq(f)\big)\}$$

The output of this component is preprocessed user reviews, weighted query words and weighted common features.

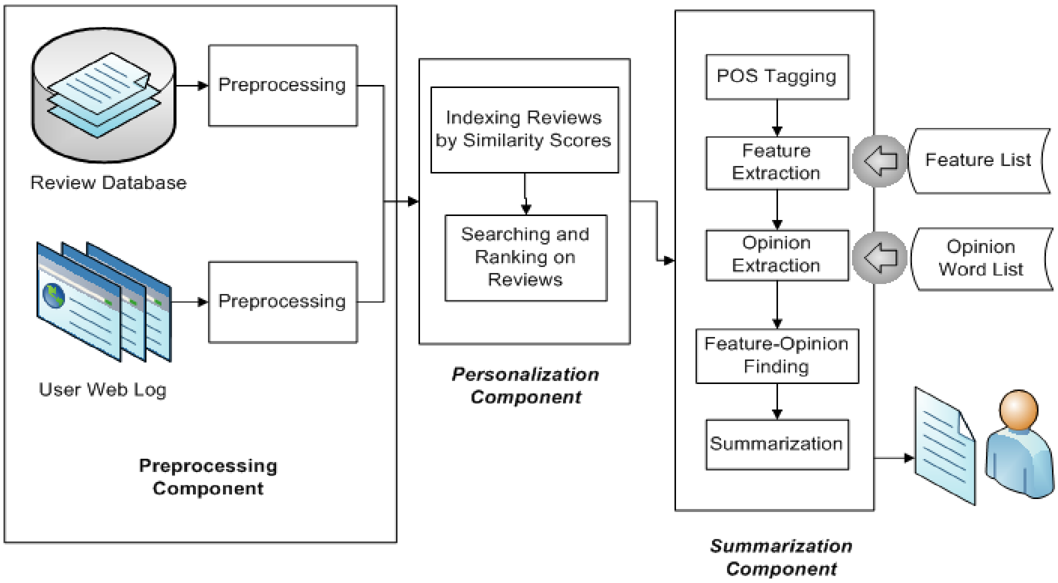

Figure 1. Personalized Review Summarization

3.2 Personalization Component

Searching on reviews is done in this component by using "Lucene" library. We differentiate boost values according to the user information to get more relevant reviews to the user. Boosting with user related boost values on indexing and searching steps is the key point for personalization.

Common features are used to identify relevant products to the user needs. We calculate cosine similarity for each model existing in the database based on the common features and boost reviews existing in the related models proportional to their similarity scores at indexing step. The similarity value for a model (m_i) and boost value for a review (r) related with this model are calculated as:

$$sim_i = \cos(cfList, fList_i)\ , \qquad boost(r) = sim_i + boost(r)$$

So, we index the reviews that exist on more similar products with higher boost values proportional to their similarity scores. In searching, we use weighted search query words and weighted common features. We highly boost them proportional to their frequencies. Each review consists of words and each word in the reviews is boosted by additional frequency scores if this word exist in common features or query words:

$$r = \{w_1, w_2, \dots\}$$

$$boost(w_i) = \{(freq(w_i) + boost(w_i)) \mid \exists w_i \in (f \cup t)\}$$

Different features and opinions could be mentioned in each sentence existing in the reviews. So, these operations are done on the sentence level. Fuzzy matching is used not to lose misspelled relevant words. Because of getting ordered results from Lucene, we could order related customer sentences existed in the reviews. The output of this component is ordered relevant reviews to the user needs.

3.3 Summarization Component

Related customer sentences are taken from personalization component. Product features that exist on sentences are nouns or noun phrases and opinions are mostly adjectives or verbs in Turkish language. So, POS (Part of Speech) tagging is used to define each word type. "Zemberek" library is used for language operations.

That we gather product features from the web site and store to the specific file could be thought as a supervised method. We want to define the existing product features on the reviews. If the POS tag of a word is noun or noun phrase, we look for this word to the file that includes product features. Because of an agglutinative feature of Turkish, the words could take some suffixes. Inflectional suffixes do not change the orientation of the word, so the root word carries also the same meaning. However, derivational suffixes change the orientation of the words, if we take their root, we could lose their meaning. So, the root types of the words are not enough to search for features. Taking the derivational suffixes into consideration, we use stem words, which are words cleaned from inflectional suffixes keeping derivational suffixes, to find mentioned features. Each noun word as stemmed is explored in the feature file. If it exists related sentence is signed with this feature.

Turkish word data set construction is currently part of BalkaNet project. But, they mainly focus on finding the hypernym-hyponym relations not synonym-antonym relations. In order to be sure, we store the synset records to the database and search for antonyms and synonyms bootstrapped by our opinion text files. We only add 20 words totally as new words. So, existing Turkish WordNet is not efficient to explore semantic orientation of the words. We have manually generated opinion words and make the opinion list file (Zhuang et al. 2006). Opinions could be on verbs or on adjectives in Turkish language. The opinions could be directly carried with the root word or could be carried with the suffixes. Searching opinions with root words are not effective for Turkish, stemming should be done, which requires taking off the inflectional suffixes and keeping the derivational suffixes. Thus, regarding to the language properties, we construct our effective opinion files as negative and positive. For each word in the sentence, firstly original versions are used to check if it has an opinion, and if nothing matches stemmed versions of the words is used.

After finding the features and opinions in the reviews, it is necessary to define which opinion is related to which feature. Generally the most adjacent opinion defines the value of a feature. Thus, we find feature-opinion pairs with this method. We take a sentence which is signed by some feature words, look for its opinion, if nothing matches continue with the other sentence.

We take the feature-opinion paired sentences and summarize according to the features to give overall opinions to the related features. The output structure of the system:

<model_name> <feature> positive (<sentence_count>)
<related_sentences>
negative (<sentence_count>)
<related_sentences>
<other_features>
<other_models>

4. EXPERIMENTAL RESULTS

4.1 Data Set

To the best of our knowledge, there is no study made in Turkish on review summarization and there is no available data set. We built our data set by crawling user reviews and product properties sections of the e-commerce web site “hepsiburada.com”. We get 4877 user reviews. Our categories are: SLR cameras (12 models), mobile phones (7 models), book (5 models), movie (6 models), sport clothes (4 models), washing machine (3 models), iron (8 models), mouse (12 models), hair straighteners (8 models) and watch (6 models). We also store 1464 product properties related to the models.

4.2 Results

We have proposed a personalized feature based summarization (PFBS) method and implemented in Java. The computer has been used in this work has Intel Core 2 Duo CPU – 2.53GHz, 4GB RAM, 32 byte Win7 operating system.

The first 100 feature-opinion paired sentences are taken to the summarization part from the systems considering the reading capability of a potential customer. We select top 5 categories from the category list that is ordered with review and feature counts: mobile phone, SLR camera, mouse, hair straighteners and wrist watch, seen in Table1.

Table 1. Number of models and reviews

Product Categories	Model Count	Review Count
Mobile Phone	7	900
Photo Camera	12	300
Mouse	12	743
Hair Straighteners	8	650
Watch	6	691

First, we evaluate our method by observation. It is assumed that user web log data comes to our system at each step. We generate 5 sample user scenarios for selected categories based on this assumption. For example, we generate a user scenario for mobile phone category. The user, looking for a mobile phone, enters “android operating system” to the search area then related mobile phone models are listed. He/she clicks “Galaxy SIII”, “Galaxy Note II” and looks over these models; then he/she enters “touch screen” to the search area and clicks “iPhone 4”, “iPhone 5” and “Galaxy SIII” to investigate them. While the process of this user, the user web log incrementally comes to our system. At the first step after searching for “android operating system”, the review panel shows the mobile phone reviews that consists these keywords and the reviews of the mobile phones that has android operating system at the top of the reviews. The models mentioned at top in the reviews are “Galaxy SIII”, “Motorola Atrix” etc. When he/she clicks “Galaxy SIII”, the reviews of the “Galaxy SIII, Galaxy Note II, Galaxy Y Pro” display in higher order while “Blackberry, HTC One” in lower order based on the feature similarity. After second “touch screen” search, the panel lists the reviews of the touch screened mobile phones that have android operating system in higher order, then the reviews of the mobile phones that have android operating system or touch screen feature are listed according to the frequencies. The observational results of our system are satisfying but, it is necessary to state the results statistically.

Then, we evaluate our method comparing to the Hu and Liu's existing feature based summarization (FBS) method using the statistical comparison tool "ROUGE", Rouge-N method (N=1). Because FBS is commonly used valid method in review mining and easily applicable, we have chosen this work for comparison. We calculate the coherence score between the summarized results and the search query words of the users to understand that how much relevant results are getting from the systems. We have implemented their method for Turkish first, and then we compared their method with our method.

We compare two systems in two perspectives:

1. F-scores for coherence
2. Running times.

First, we only take the reviews existed in their related category. For example, the reviews in the mobile phone category are taken to the systems for a mobile phone user's scenario. We want to understand that how much relevant reviews are taken although the related category is given to the systems. The coherence results are shown in Figure 2 and PFBS has really higher coherence scores than FBS.

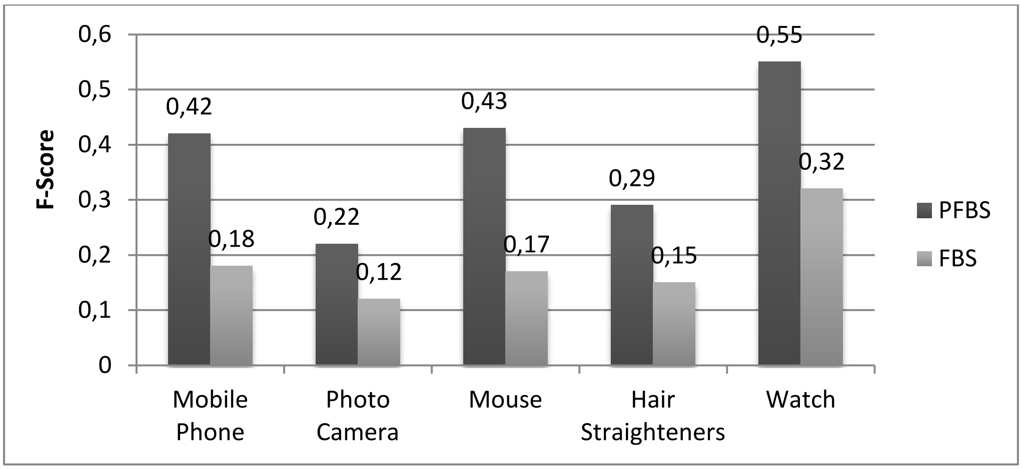

Figure 2. F-Scores with the reviews taken from the related categories

Second, we take all the reviews to the systems ignoring their related categories. As seen in Figure 3, our PFBS has the same values as in the results with the related categories. This shows that PFBS is stable with respect to getting related reviews in both conditions. The coherence scores for FBS are less than or equal to the results in the first condition. This states that the most relevant reviews are taken when the reviews in the related categories are given to the FBS.

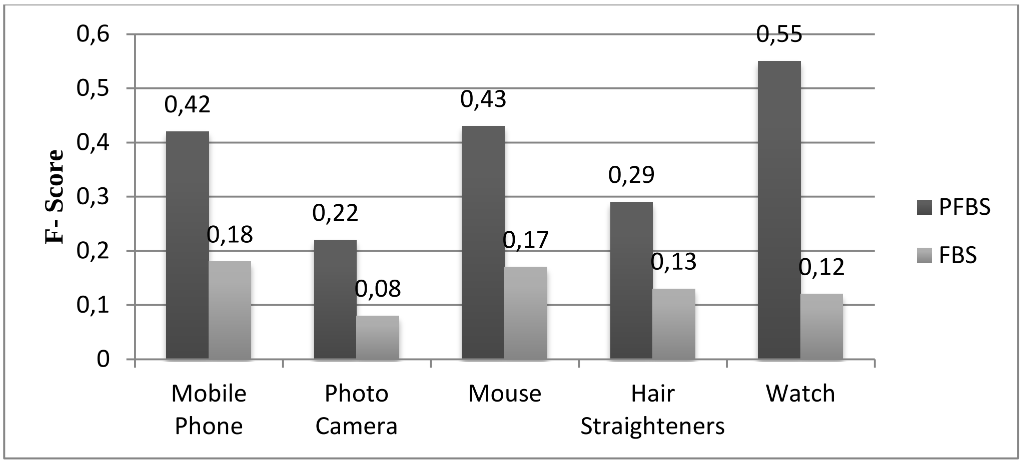

Figure 3. F-Scores with all reviews

Also, we compare two systems according to their runtimes. Our method has shorter runtimes than FBS because of dealing with just the related reviews that are taken from the search process. But, FBS deals with all of the reviews and this makes it slower than our system. As seen from Table2, our system is approximately 36 times faster than FBS when the reviews are taken in the related categories, and also approximately 55 times faster than FBS when all the reviews are taken. This shows that our system is more successfull on operating high volume data than FBS.

Table2. Runtimes for both systems (milliseconds). PFBS1, FBS1: The reviews taken from the related categories for PFBS and FBS; PFBS2, FBS2: All of the reviews taken for PFBS and FBS

Product Categories	PFBS1	FBS1	PFBS2	FBS2
Mobile Phone	41050	851911	117505	3729585
Photo Camera	2899	212763	34929	2885062
Mouse	10073	405566	41113	2803032
Hair Straighteners	23419	418385	104436	3672988
Watch	12464	364486	56822	3297990

We observe that when users search with general words like “sports watch”, “cheap smart phone”, “wireless optic mouse” etc., much more relevant scores are taken than feature based searches like “Android operating system”, “ionic hair straighteners” etc. Also, we observe that if the product has numerous features, the diversity in the reviews is getting higher and this causes less coherence scores.

When all of the reviews are given to the systems in the sample scenarios, we see the irrelevant featured reviews are listed in the results in FBS. For example, the “steam boiler” featured sentences are listed in the results for a mobile phone customer.

When the reviews in the related categories are given to the systems, the results seem relevant. But, the results could be irrelevant for the potential customer in detail. For a mobile phone user’s scenario, when the reviews in the mobile phone category are given to the systems, only mobile phone reviews are listed in the results in both systems. But, if a customer searches for Android operating system not for Bada system, the reviews on both operating systems occurred in the results. However, Android comments tend to exist in higher orders in the PFBS results. Thus, PFBS has higher coherence scores than FBS in all of the situations.

PFBS deals with just filtered reviews not all of the reviews. This makes the system faster and memory efficient. Behind the coherence scores, PFBS has shorter runtimes and less memory allocations than FBS.

5. CONCLUSION AND FUTURE WORK

In our work, we try to order customer reviews for a potential customer regarding his/her interests. While surfing on the web site, we get his/her web log data at each page and filter related customer reviews based on his/her search queries and click-through pages. We display feature based summarization for each model obtained from filtered reviews. The results are very promising in our work and we believe that personalization in feature based review summarization systems will become important in the near future.

We have implemented personalization methods on feature based review summarization techniques. This contributes to the existing works by faster running times and displaying relevant summarization results to the user expectations. Although this study is carried out on Turkish language, it is also possible to apply the same procedure for other languages by changing the "preprocessing" component responsible for language operations and keeping the other two components the same.

In this study, we plan to improve and refine our techniques further. We want to expand our opinion words and preferably find a new method to make it automatically. We want to give weights to the opinion words; some opinion words have more strength emphasis. It would be good if we identified their strengths. We also want to find effective grammatical semantic relations for Turkish language. We have worked on Turkish language for this work and we also want to implement necessary operations for English language to provide a common usage. Finally, we want to make a sample web interface to the system to be evaluated by the human evaluators.

REFERENCES

Agarwal, D., Chen, B. C., & Pang, B. (2011, July). Personalized recommendation of user comments via factor models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing* (pp. 571-582). Association for Computational Linguistics.

Baccianella, S., Esuli, A., & Sebastiani, F. (2009). Multi-facet rating of product reviews. *Advances in Information Retrieval*, 461-472.

BalkaNet - Design and Development of a Multilingual Balkan WordNet. http://www.dblab.upatras.gr/balkanet/.

Benammar, A. (2004). ENHANCING QUERY REFORMULATION PERFOMANCE BY COMBINING CONTENT AND HYPERTEXT ANALYSES.

Blair-Goldensohn, S., Hannan, K., McDonald, R., Neylon, T., Reis, G. A., & Reynar, J. (2008, April). Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era*.

Cui, H., Wen, J. R., Nie, J. Y., & Ma, W. Y. (2003). Query expansion by mining user logs. *Knowledge and Data Engineering, IEEE Transactions on*, *15*(4), 829-839.

Eirinaki, M., & Vazirgiannis, M. (2003). Web mining for web personalization.*ACM Transactions on Internet Technology (TOIT)*, *3*(1), 1-27.

Hu, M., & Liu, B. (2004, August). Mining and summarizing customer reviews. In*Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 168-177). ACM.

Titov, I., & McDonald, R. (2008). A joint model of text and aspect ratings for sentiment summarization. *Urbana*, *51*, 61801.

Katakis, I., Tsoumakas, G., Banos, E., Bassiliades, N., & Vlahavas, I. (2009). An adaptive personalized news dissemination system. *Journal of Intelligent Information Systems*, *32*(2), 191-212.

Joachims, T. (2002, July). Optimizing search engines using clickthrough data. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 133-142). ACM.

Lu, Y., Zhai, C., & Sundaresan, N. (2009, April). Rated aspect summarization of short comments. In *Proceedings of the 18th international conference on World wide web* (pp. 131-140). ACM.

Lucene. http://lucene.apache.org/core/.

ROUGE:Recall-Oriented Understudy of Gisting Evaluation. http://www.berouge.com/Pages/default.aspx

Sun, J. T., Shen, D., Zeng, H. J., Yang, Q., Lu, Y., & Chen, Z. (2005, August). Web-page summarization using clickthrough data. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 194-201). ACM.

Tadano, R., Shimada, K., & Endo, T. (2010). Multi-aspects review summarization based on identification of important opinions and their similarity. In *Proceedings of the 24nd Pacific Asia Conference on Language, Information and Computation (PACLIC24)* (pp. 685-692).

Willemsen, L. M., Neijens, P. C., Bronner, F., & de Ridder, J. A. (2011). "Highly Recommended!" The Content Characteristics and Perceived Usefulness of Online Consumer Reviews. *Journal of Computer-Mediated Communication*,*17*(1), 19-38.

Wu, X., Xie, F., Wu, G., & Ding, W. (2011, November). Personalized news filtering and summarization on the web. In *Tools with Artificial Intelligence (ICTAI), 2011 23rd IEEE International Conference on* (pp. 414-421). IEEE.

Xue, G. R., Zeng, H. J., Chen, Z., Yu, Y., Ma, W. Y., Xi, W., & Fan, W. (2004, November). Optimizing web search using web click-through data. In *Conference on Information and Knowledge Management: Proceedings of the thirteenth ACM international conference on Information and knowledge management* (Vol. 8, No. 13, pp. 118-126).

Yang, B., & Datta, A. Rated Aspect Summarization of Online Reviews: A Semi-supervised Topic Model Based Approach.

Yurekli, B., Capan, G., Yilmazel, B., & Yilmazel, O. (2009, October). Guided navigation using query log mining through query expansion. In *Network and System Security, 2009. NSS'09. Third International Conference on* (pp. 560-564). IEEE.

Zemberek 2 is an open source NLP library for Turkic languages. http://code.google.com/p/zemberek/.

Zhang, K., Cheng, Y., Liao, W. K., & Choudhary, A. (2011). Mining millions of reviews: a technique to rank products based on importance of reviews. In *10th international conference on entertainment computing*.

Zhuang, L., Jing, F., Zhu, X. Y., & Zhang, L. (2006, November). Movie review mining and summarization. In *Conference on Information and Knowledge Management: Proceedings of the 15 th ACM international conference on Information and knowledge management* (Vol. 6, No. 11, pp. 43-50).