RESEARCH ARTICLE OPEN ACCESS

Opsum: Topic Based Opinion Summarization And Sentiment Analysis

Bhushan Gawde*, Simran Motwani**, Aishwarya Jadhav*** and Bhargav Makwana****
*(Software Engineer, Samsung R&D Bangalore
**(Department of M. Engineering, University of California Berkeley, California,USA
***(Software Engineer, Morgan Stanley
****(Software Engineer, KAL Atm
Corresponding Author: Bhushan Gawde

ABSTRACT
We present a system code-named OpSum for topic-based opinion summarization and sentiment analysis of mobile phone reviews. It enables users to decide whether to purchase or not based on a summary of the reviews for that mobile phone. Our system organizes the reviews based on product aspects extracted from the dataset and it also provides a sentiment analysis of these reviews. Selection of useful reviews is done from these collections of organized reviews. We discuss the effectiveness of using GRNNs and LSTMs for sentiment analysis and their trade-offs. We alsoperformed extractive summarization using Integer Linear Programming to extract the most distinct and important sentences from the clusters which will represent the reviews in the entire cluster.

Keywords **-**GRU. Integer Linear Programming. LSTM. Sentiment Analysis. Summarization. Text Mining. Introduction

---

Date Of Submission:05-09-2018 Date Of Acceptance: 21-09-2018

I. INTRODUCTION

In today's digital world, where Social Media and Online Shopping are enjoying a boost, product reviews have a very special place. People turn to online reviews before finalizing to buy any product.

A survey conducted by Trademarks Productions in United States [1] which was published in October 2017 has found few important facts about the online purchase of products and relationship between reviews and purchases done by customers. The following results were reported by this survey:

- 8.8/10 consumers trust online reviews asmuch as the word of the mouth.
- More than 10 online reviews are seen by an average person before fixating a procurement.
- 7/10 people believe in the views of existing buyers of the product posted online.
- Good and bad customer experiences are shared by buyers to 42 and 53 people respectively.

In this paper, we present an approach to identify the sentiments of people about different aspects of a mobile phone model like battery, camera, sim card etc. from online reviews and to summarize them. We first cluster the review sentences based on different features of the mobile phone being talked about in it. Then those sentences go through sentiment analysis using GRU/LSTM module to give the positivity percentage of that particular aspect. Then we summarizethe sentences in the cluster based on Maximum Coverage and Minimum Redundancy to form a briefand a concise summary from the reviews belonging to that particular cluster.Our work touches upon the domains of sentiment analysis and extractive summarization.

There has been a lot of research done in these domains. Twitter based corpus has also been used extensively by people for performing sentiment analysis [2], [3]. They have used SVM for sentiment analysis. Unsupervised approach like lexicon-based method is also used for this [4], [5]. For Summarization, latent semantic analysis has been a popular technique used by people [6], [7]. [8] proposed an approach in which Regression models are used for ranking the sentences afterdoing summarization. This model considers multipledocuments while summarization. The summarization approach is query-based.

Section 2 presents the entire flow of our system. It also discusses the method that we have used. Section 3 presents the details of our experiments and the results that we obtained by performing those. Finally,Section 4 provides the conclusion.

II. METHODOLOGY

Fig 1 shows the diagram explaining the flow of our system.

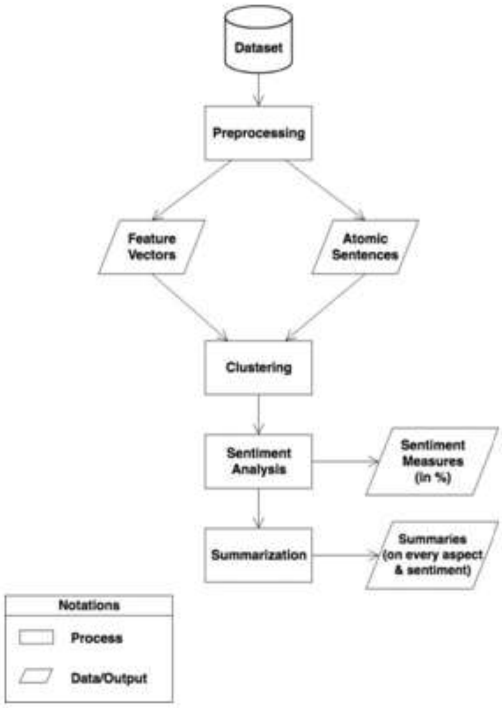

Fig. 1. Proposed Methodology of System

2.1 Topic BasedClustering

We obtained our dataset by web scraping of the Amazon product reviews web page. For each mobile phone there were anywhere between 10000 to 3 lakh reviews. We used data cleaning techniques like removal of punctuation and words irrelevant to either the task of sentiment analysis or summarizarion. The system creates summaries of the reviews of people on a particular product based on the different aspects (topics) of that product.

2.1.1 Extracting topics from the dataset.

A tfidf score has been assigned to each noun in the dataset. We included the product reviews of 5 other products disparate from mobile phones in the document corpus used for the tfidf calculation so that words pertinent to mobiles can get a high tfidf score and hence be extracted out as topics. In the next step, we cluster the nouns which have a high tfidf score together into “topic-groups” and these groups have been formed on the basis of their semantic meanings. We trained our own word embeddings model on the dataset formed by combining the reviews of 5 other mobile phones using the Word2Vec algorithm with the skip-gram model.

The tf-idf weight consists of two terms:

$$tf = count(word, document)/len(document) \quad (1)$$

$$idf = log(len(collection)/count(document)) \quad (2)$$

Finally, tf-idf is calculated as follows:

$$tfidf = tf * idf\ (term, collection) \quad (3)$$

2.1.2 Clustering sentences based on the topics-groups

One cluster per topic-group has been created. Before the clustering the sentences were atomized based on conjunctions. A sentence will be said to be an atomic sentence onlyif it consists of atleast one noun or pronoun and atleast one adjective or verb. Finally, an atomic sentence is assigned to a particular topic-group if it contains one or more topics from that topic group. It is possible for an atomic sentence to belong to more than onecluster as a review can talk about more than one aspect of the mobile in a phrase.

2.2 SentimentAnalysis

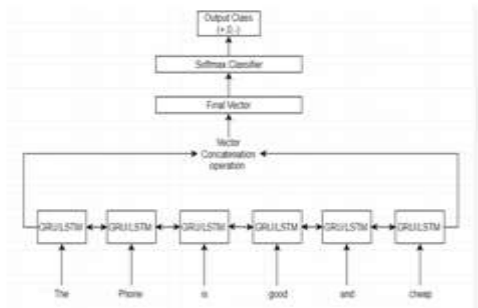

Fig. 2. Sentiment Analysis Neural Network

The input is a batch of review sentences with target entities, and output is the sentiment polarity of each sentence in the batch. The sentiment class is computed in two steps. First, each word in a given tweet is mapped to a real-valued embedding vector using sentiment specific word embeddings. Then they are connected bidirectionally. We experimented using Long Short-TermMemory unitsandGated Recurrent units both for this bidirectional connection. The sequence of gates which we get by connecting these unitsis called the Gated Recurrent Neural Network [9]. The output of this sequence of gates is a low dimensional vector which is called as the hidden state vector. Two low dimensional hidden state vectors are obtained; one for the accumulated weights in each direction. This bidirectional network helps to properly model not just the syntactic but also the semantic information at the sentence level. Before giving the output to the classifier, these two vectors are concatenated. This concatenated low-dimensional vector is given as an input to a linear classifier. The number of units should be equal to the size of the review sentence.

We usedSoftMax classifier for getting the final output value. The underlying semantic information of input review sentences scopes is explicitly captured by this model. This information may include dependency relations or co-references etc. We have tried withthree as well as two(only positive and negative) output classes.Fig 2 shows ourmodel.

2.2.1 Random Forest

The sentiment analysis process is usually done taking an entire review as an input and the rating given by the people is considered as its output labels. For the summarization of a review of a phone according to its different features, an entire review will not be of any use.Therefore, we used aspect-based sentiment analysis which does the task of analyzing sentiments according to the different aspects or the features of the product. We split the reviews into atomic sentences, each of which focuses on review of a single feature. Therefore, the actual final sentiment labels are not available. Hence, we used a pretrained model for gettingthe final output labels for our dataset. It was trained using a random forest classifier. The random forest classifier has been trained on around 5600 reviews. The model has been given in[10].

2.3 Summarization
2.3.1 Problem Formulation

In the summarization process, we find a subset of sentences which contains sentences very different from each other and will represent the main idea or the main content of the text. We use a set of criteria based on which we decide which sentences should be added to the subset and which shouldn't be. Alguliev et al. (2011) describes these criteria as Relevance, Redundancy and Length.

2.3.2 Mathematical Formalization

The model proposed by Aliguliev et al. (2011) [11] is used. We consider set of documents (reviews here) as our input and we represent this set as a set of sentences D:

$maximize\ f = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[sim(\vec{D},\vec{s}_i) +\right.$
simD,sj−simsi,sjxij (4)

$such\ that\ \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[len(\vec{s}_i) + len(\vec{s}_j)\right] x_{ij} \leq L;\ x_{ij} \in \{0,1\}, \forall\ i,j$ (5)

Let $D = \{s_1, s_2, \ldots\ldots, s_n\}$ be the document and $T = \{t_1, t_2, \ldots, t_m\}$ represent the terms occurring in the document. Where $\vec{s}\&\vec{D}$ denote the feature vectors of a sentence and a document and $sim(\vec{s_i},\vec{s_j})$ denotes the similarity between two feature vectors. The problem is formulated as shown.

2.3.3 Normalized Google Distance (NGD) Based Similarity

We have used NGD to measure similarity of two sentences. These two sentences are represented as sequence of terms.

$$sim_{NGD}(s_i, s_j) = \frac{\sum_{t_k\in s_i}\sum_{t_l\in s_j} sim_{NGD}(t_k,t_l)}{|s_i|\cdot|s_j|} \quad (6)$$

From (6), s_i is a sentence treated as a set of terms i.e. $s_i = \{t_1, t_2, \ldots, t_{|s_i|}\}$. Similarity between two terms is measured as $sim_{NGD}(t_k, t_l) = \exp(-NGD(t_k, t_l))$ where t_k and t_l represent terms of two different sentences.

2.3.4 Cosine Based Similarity

This similarity measure has been used to get the cosine of angle made by a pair of sentences represented as vectors:

$$sim_{cos}(\vec{s_i},\vec{s_j}) = \frac{\sum_{k=1}^{m} w_{ik}w_{jk}}{\sqrt{\sum_{k=1}^{m} w_{ik}^2 \cdot \sum_{k=1}^{m} w_{jk}^2}}, i,j = 1,\ldots\ldots,n$$
(7)

Units of the sentence vectors i.e. words are represented using tf-isf. Similarity is then measured using the weights for each textual unit obtained by this scheme of representation.

2.3.5 Objective Function

The objective function provides a numerical score for every possible pair of sentences from the dataset. This measure is then used to determine which pair is to be included in the set of summaries generating sentence sets. The function combines both the similarity measures discussed above:

$$maximize\ f_\propto = \propto . f_{cos} + (1-\propto). f_{NGD} \quad (8)$$

$where\ f_{cos} = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[sim_{cos}(\vec{D},\vec{s}_i) +\right.$
simcosD,sj−simcossi,sjxij,(9)

$f_{NGD} = \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[sim_{NGD}(\vec{D},\vec{s}_i) +\right.$
simNGDD,sj−simNGDsi,sjxij, (10)

Contribution of both similarity measure in the output of the given function is controlled using the variable $\propto \in [0,1]$.

2.3.6 Branch and Bound Algorithm

If we have$n$ sentences, we get ${}^{n}C_2$ pairs and there are $2^{{}^{n}C_2}$ ways in which we can choose these pairs.We used Branch and Bound algorithmto narrow down the combination of pairs that will ultimately constitute the final summary.Basically, what we do is we prune the branches of the Branch and Bound tree that won't give any better results than current optimal result. We keep a lower bound on the profit we get so that we always find the most optimal solution.Also, we don't have to examine every possibility. The summarization problem that we are tackling here is very similar to the 0-1 Knapsackproblem.

Table 1. 0-1 Knapsack and Summarization Analogy

0-1 Knapsack Problem	Summarization using ILP
Knapsack Capacity	Permissible Length ofSummary
Item	Pair of Sentences
Weight of an Item	Sum of length of pair of sentences
Profit of an Item	Score given to a particular pair

III. EXPERIMENTS AND RESULTS

There were a total of 7692 nouns in the dataset. We calculated their corresponding tfidfs. We considered only the nouns having tfidf above 0.002; 152 nouns satisfied this criterion.We created word vectors of dimension 100 for each of these nouns that satisfied the criterion.We can easily see the benefit of creating our own word embeddings in thescatter plot of figure 3 made using a few of the words from our topic list. These vectors have clearly captured the context as they were trained on mobile reviews dataset.

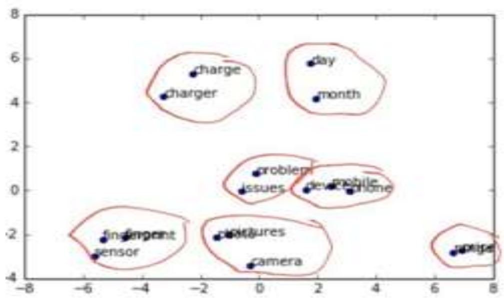

Fig 3. Scatter plot of words from topic list

The K-means clustering algorithm was used to group the word vectors corresponding to the topics together. Following are the parameters used for thealgorithm:

- Number of clusters or topic-groups: 25 (experimentally found to beoptimum)
- Random initialization of cluster centroids with points of thedata
- 500 such random initializations; we have chosen that topic-group which is best amongstthese.
- 2000 iterations in each initialization

Following are some of thetopic-groups:
1: ['night', 'photos', 'picture', 'camera', 'light', 'front', 'flash', 'rear', 'mode', 'depth 'effect', 'pictures', 'photo', 'cameras']
2: ['slot', 'card', 'memory', 'sim']
3: ['charger', 'charging', 'turbo', 'battery', 'life', 'backup', 'charge']

We used Sentiment Specific Word Embeddings for doing Sentiment Analysis [12]. The size of the word embeddings used is 50.

We usedaround 65200 sentences for training. The final labels were obtained from the pretrained Random Forest classifier which is discussed already. We experimented with many different values of learning rate. Finally,after a lot of tuning, alearning rate of 0.001 gave the best accuracy. The hidden layer size was taken as 128. We tried experimenting with different optimizers and final decided to use Adam Optimizer, since it gave the best results and is usually preferred over any other optimizer by people. Training accuracies of 73% and 70.5% was obtained for LSTM and GRU respectively for 3 classes. For 2 classes, the corresponding accuracies were 75% and 74.5%. The following were the test results obtained.

Table 2. Test Results

	3 Classes	2 Classes
LSTM	**60.3%**	**62.8**
GRU	**61%**	**63.6%**

For finding the similarity between two sentences, we have made use ofword embeddings instead of using just the frequencies of terms in the sentences. The word embedding similarity measure was replaced with cosine similarity discussed in the summarization model though it uses cosine similarity internally. The word embedding model we used contained almost 300 various dimensions of the words. This resulted in the overall improvement in the results and we obtained a better set of sentences. Following are the set of sentences obtained from a cluster representing positive reviews of a particular product aspect:-

- it gives two days of moderate use even though it packs only3000mah
- supports two nano-sims both of which can connect to 4g networks
- first of all let 's come to the pros :) premium looks primary dual cam (hoping to solve the shutter lag in an ota update) full hd display with corning gorilla glass 3 water repellent nano-coating 4k video recording (eis) snapdragon 625 soc quick charging support lot of the reviews says that the camera doesn't meet theirexpectation
- 5+ hours screen ontime
- rest of the times works fine
- phone does heat at thattime
- heatingproblem
- no heating issues except while using depth

mode that toomoderate

IV. CONCLUSION

Product Reviews is an integral factor for online shopping. In this project, we have developed a system to make it easier for the customers to easily go through the reviews just by reading the summary. Our system uses online reviews posted by people to determine the features of the product. Then, we calculate the sentiment of the reviews. Our system gives a textual summary of the product. The system produces two summaries for each feature, one portraying the negative sentiment and the other portraying positive sentiment.

REFERENCES

[1]. Goodlaxon, Sarah (2017, October 17). 15 Facts About Online Reviews and What They MeanforBusinessOwners.[https://www.tmprod.com/blog/2017/15-facts-online- reviews-mean-business-owners/]

[2]. Pak, Alexander & Paroubek, Patrick. (2010). Twitter as a Corpus for Sentiment Analysis and Opinion Mining. Proceedings of LREC. 10.

[3]. Jonathon Read. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In ACL. The Association for Computer Linguistics.

[4]. Qiu, Guang, Bing Liu, Jiajun Bu, and Chun Chen. Expanding domain sentiment lexicon through double propagation. In Proceedings of international Joint conference on artificial intelligence (iJcai-2009).2009.

[5]. Taboada, Maite, Julian Brooke, Milan Tofiloski, Kimberly Voll, and Manfred Stede. Lexicon- based methods for sentiment analysis. computational Linguistics, 2011. 37(2):pp.267307.doi:10.1162/COLI_a_00049

[6]. Shen C, Li T, Ding CH (2011) Integrating clusteringandmulti-document summarization by bi-mixture probabilistic latent semantic analysis (PLSA) with sentence bases. In:AAAI

[7]. Shen C, Li T, Ding CH (2011) Integrating clusteringandmulti-document summarization by bi-mixture probabilistic latent semantic analysis (PLSA) with sentence bases. In:AAAI

[8]. Yeh J-Y, Ke H-R, Yang W-P, Meng I-H (2005) Text summarization using a trainable summarizer and latent semantic analysis. Inf ProcessManag41:7595.doi:10.1016/j.ipm.2004.04.003

[9]. Ouyang Y, Li W, Li S, Lu Q (2011) Applying regression models to query-focused multi-document summarization. Inf Process Manag47:227–237

[10]. Meishan Zhang , Yue Zhang , Duy-Tin Vo, Gated neural networks for targeted sentiment analysis, Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix,Arizona

[11]. Sreeharibs(May 1,2017)Mining mobile reviews[https://github.com/sreeharibs/Review-Analysis]

[12]. Alguliev RM, Aliguliyev RM, Hajirahimova MS, Mehdiyev CA (2011) MCMR: maximum coverage and minimumredundanttextsummarizationmodel. ExpertSystAppl38:1451414522.doi:10.1016/j.eswa.2011.05.033

[13]. Tang, D.; Wei, F.; Yang, N.; Zhou, M.; Liu, T.; and Qin, B. 2014. Learning sentiment-specific word embedding for twitter sentimentclassification

Bhushan Gawde "Opsum: Topic Based Opinion Summarization And Sentiment Analysis "International Journal of Engineering Research and Applications (IJERA) , vol. 8, no.9, 2018, pp 53-57