Survey On Movie Rating And Review Summarization In Mobile Environment

Pradnya Mehta
Department of Computer Engineering

Abstract

This paper makes a survey of various design and development strategies required for movie-rating and review-summarization system in a mobile environment. The product feature extraction is done by various methodologies such as maximum entropy model, latent semantic analysis (LSA) algorithm, statistical approach, Support vector machine (SVM) technique and, Naive Bayes model. The categorization of movie rating is done by sentiment analysis result. The main goal of review mining and summarization is extracting the features on which the reviewers articulate their opinions and determining whether the opinions are positive or negative. The rating and review summarization system can be extended to other product-review domains effortlessly. The above mentioned approaches are compared in this paper.

Index Terms—Feature extraction, natural language processing (NLP), text analysis, text mining, Machine learning approaches.

1. Introduction

Now a day, for issuing various types of services in social network the opinion of people is very significant. Mostly, when we are not aware of a particular product, we inquire to different sources to suggest one. Online opinions can recognize new openings for products and manage their status. Now days, the Internet compose people to explore for other people's opinions from the Internet before purchasing a product or seeing a movie. Meanwhile, cellular phones become certainly become the most important part of our lives. Since the digital contents displayed on cellular phone is limited because the cellular phones are physically small. Hence, a compressed description of documents will aid the delivery of digital content in cellular phones. This paper discover and designs a mobile system for movie rating and review summarization in which semantic orientation of comments, the limitation of small display capability of cellular devices, and system response time are considered [1].Along with the websites search engine is also one of the famous tool to get the opinion of people. Most of the websites provides user ratings in percentage and search engine monitors best matching web pages according to its pattern. But current search engine does not provide semantic orientation of the content in review. Online opinion is the best mean in business application to determine their scope of product in the market. To make a decision about movie rating i.e. whether it is positive or negative is based on binary classification. Because of the limited size of cellular phone summarization technique is used. The system will provide summary about the reviews. Meanwhile, movie review summarization is similar to customer review that focuses on product feature. The product features an be identified by various techniques such as maximum entropy model, latent semantic analysis (LSA) algorithm,Support vector machine (SVM) technique and, Naive Bayes model. The review mining and summarization will make the sentiment summary in which sentences from reviews that confine the author's opinion. Section 2, describes associated surveys. In Section 3, various machine learning approaches required for movie review and summarization is described. In Section 4 comparative discussion of different approaches are defined and in Section 5, the conclusion is described.

2. Related Work

2.1 Sentiment Analysis

The chore of sentiment categorization is to decide semantic orientation of words, sentences, or documents. Hatzivassiloglou and McKeown [1] employed textual combination such as "light and legal" or "naive but well received" to detach synonym and antonym words. Turney used point wise mutual information (PMI) as the semantic distance between two words so that the sentiment potency of word can be computed without difficulty. The Alta vista search engine consent to maximum distance between the two words is ten.SENTIWORDNET is the lexical resource expressed by Esuli and Sebastiani [1], with three numerical scores, i.e. (s), Pos(s), and Neg(s) gives opinion about the movie review.Numerous machine reviews from IMDB.Pang *et al* initiate that standard machine learning surpassed than human recommend baselines. Naive Bayes, maximum-entropy classification, and support vector machines (SVMs) are used for sentiment-classification.Popescu and Etzioni [6] developed an unsupervised information extraction system called OPINE extracting product

features and opinions from reviews. PMI-IR algorithm [6] is used to calculate the mutual information between the review and the polarized words to generate the weight of the ratings. Learning advances are used with customary text features to pigeonhole movie.

2.2 Feature-Based Summarization

Customers opinion about the product is deliberated by feature based summarization. Product features and opinion words play a vital role in feature based summarization. The statistical approach can be used to discover the product features but the consequence is intermittent features are ignored. Meanwhile, Zhuang *et al.* mined to make use of grammatical rules and keyword lists to hunt for feature-opinion pairs and generate feature-based summarization. Features in movie reviews are some proper nouns, including people names and movie names. Likewise, a name may be articulated in various forms, such as last name only, full name, first name, middle name or abbreviation. Name can be detected by a cast library, which is built as a special part of the feature word list by downloading and saving full cast of each movie first and removing people names that are not mentioned in training data. By removing the surplused names, the size of the cast library can be abridged extensively. Thus, we cannot identify the product features and opinion words in movie reviews using the POS tagging methodology.

Hu and Liu's work as pioneer on feature Extraction algorithm. They made the use of Apriory algorithm to extract the frequent feature words. Opinion words are identified by NLP Linguistic Parser (tagger) to make the score list and from them decide the review is positive, negative or neural. Based on part-of-speech (POS) tagging, Liu Bing carried out the successively extraction of frequent features, opinion words and occasional features through the cooccurrence relationship among them [1]. Lu *et al.*utilized POS tagging and chunking function of the OpenNLP2 toolkit to classify phrases in the form of a pair of head term and modifiers [1].

3. Machine Learning Approaches

3.1 Maximum Entropy Model

In maximum entropy model, the information required for categorization is gathered from various heterogeneous sources. The fundamental principle of maximum entropy is that without exterior knowledge, one should prefer distributions that are consistent. This job can be re-formulated as a classification problem, in which the job is to monitor some linguistic framework s belongs to S and calculate the correct linguistic class c belongs to C.We can implement classifier *cl*: S ->C with a conditional probability model by simply choosing the class c with the highest conditional probability in the framework *c*

$$Cl(x) = \text{argmax}\ p(c \mid s) \qquad (1)$$

The conditional probability *p* (c|s) is defined as follows [2]:

$$P(c|s) = (1 \div Z(c)) \prod_{i=1}^{k} \alpha_i^{fi(s,c)} \qquad (2)$$

$$Z(s) = \sum \prod_{\alpha i}^{fi(s,c)} \qquad (3)$$

where c refers to the result, *s* is the history (or context), *k* is the number of features and *Z*(s) is a normalization factor to ensure that $\sum p(c|s) = 1$. Each parameter α_i corresponds to one feature *fi* and can be interpreted as a weight for that feature. The parameters are projected by a procedure called Generalized Iterative Scaling (GIS) [2] where a feature is defined here as a function *f*: X x Y -> {0, 1} that maps a pair (*x*, *y*) to either 0 or 1. The feature is defined as follows:

$$f'(s, c) = \begin{cases} 1 & If\ c' = c\ and\ cp(s)\ true \\ 0 & \text{otherwise} \end{cases}$$

where *cp*(*x*) is contextual predication that returns true or false.

3.2. Naive Bayes Classifier

Naive Bayes classifier is used to categorize the review according to polarity of the data sets as either positive or negative. In this technique, the basic requirement is to determine the 'objective' labels of each document for categorization of review. The Documents in the test set are assigned to such a class that maximizes the probability of that class given the document in the test set. It consists of following things

c1=positive and c2=negative classes
x=length normalized documents
d=documents

$$x = \frac{1}{N(d)}(n_1(d); : : : ; n_m(d))$$

where,

ni (d) = Number of times feature i appears in document d or the presence of feature i in document d
N (d) = total number of words in document d.
$C' = \text{argmax}_c P(c|x) = \text{argmax}_c P(c) P(x|c)$
Where
C' = predicted class
P(c) and P (x|c) =Training Data

$$P(x|w) = \pi_{1}^{m} P(fi|c)^{ni(x)}$$

In this fi is the i th feature.

3.3 SVM

SVM is working same as that of Naive Bayes classifier. Training data contains the positive and negative reviews. Data provided in training classifier uses positive and negative reviews but they do not deal with complicated data which is very hard to classify. The reviews are said to be of low variance if they are giving only positive or only negative opinions otherwise they are having high variance.

In the two-category case, the basic idea behind the training procedure is to find a hyperplane, represented by vector $\vec{w}$, that not only separates the document vectors in one class from those in the other, but for which the separation, or margin, is as large as possible. This search corresponds to a constrained optimization problem; letting cj belongs to {1,-1} (corresponding to positive and negative) be the correct class of document dj , the solution can be written as

$$\vec{w} := \sum_{j} \alpha_j c_j \vec{d}_j \quad \text{where } \alpha_j >= 0$$

where the α_j 's are obtained by solving a dual optimization problem. Those vector dj such that $\alpha_j >= 0$ are called support vectors, since they are the only document vectors contributing to vector w. Classification of test instances consists simply of determining which side of vector w's hyperplane they fall on.

It uses a chunk of features framework where the features are predetermined, usually n-grams. The occurrence of features appearing in document d.

$$x = \frac{1}{N(d)}$$

where ,

N(d) = the total number of words in document d.

ni(d) = the number of times feature i appears in document d
As labelled data is available, so a classifier can be trained using the supervised learning technique of SVM where a response variable yi defined for each data point xi as

yi = +1 if xi =1
or
yi= -1 if xi = - 1

3.4. Latent Semantic-Analysis

Automatic scoring by means of Latent Semantic Analysis (LSA) has been pioneered to advance the traditional human scoring system. The principle of the present study is to build up a LSA-based assessment system to evaluate sentence construction skills and to examine a hypothesis and technique to observe associations between a set of documents and the terms they contain by producing a set of concepts related to the documents and terms. In subtest 2 (two-character words sentence construction test), the rearrangement of the two-character words produced high similarities between grammatically incorrect sentences and the best answers provided by the automated scoring system[4]. LSA can be applied to any type of count data over a distinct domain, which is so-called two-mode data [5]. Assuming that D be the collection of documents {d1, d2 …dn) with terms from *W = {w*1. . . *wm}* are given, then the system can build *n* × *m* Coincidence matrix M, where each entry *Mij* indicates the number of times the term *wj* occurred in document *di*. Row vector of each document is represented as di, while a column vector represents each term as *wj*.

The term-document matrix *M* make the use *of* relevant singular-value decomposition (SVD) technique and a low-rank approximation of the matrix *M* could be used to decide patterns in the relationships between the terms and concepts contained in the text.

$$M = U\Sigma V \qquad (1)$$

Where,

U and V are matrices with orthonormal columns
M= term document matrix
Σ is a diagonal matrix whose diagonal elements are the singular values of *M*.

Algorithm 1: LSA based Product Feature Identification Algorithm

Input: Term – Document matrix M

Compressed dimension r
Product feature seed set ps
Number of mined seed features for each seed n

Output: A relationship array F
Product feature seed f
Its corresponding value is f's related product features

1. Start
2. Initialize relationship array F
3. U,∑',V = svd(M,r)
4. M' = U x ∑' x V
5. for f € ps do
6. wf=GetTermVectorFromTermDocMatrix(f, M)
7. initialize similarity list *sim*
8. Initialize i=1
9. For each column vector w of M' do
10. sim[i]=wf.w
11. Increment value of
12. i++
13. End
14. Sort(sim)
15. relatedFeatureList=GetTopRelatedFeatures(sim,n,M')
16. F(f)= relatedFeatureList
17. End
18. Return F
19. Stop

The reduced size of term document matrix could be estimated by novel term-document matrix. Dimension diminution is used to eliminate the irrelevant information and inconsistency in type and document vectors which referred to as "noise". Reduced matrix M' is shown in equation (2). After SVD and dimension reduction, M is the r-dimensional vector space which is called "semantic space".

$$\tilde{M} = U\tilde{} \Sigma V \approx U \Sigma V = M. \quad (2)$$

The similarity is computed as the cosine of the vector representation of the sentences in equation 3)

$$\text{sim}(s1, s2) = \frac{d1d2^T}{\|d1\| \, \|d2\|} \quad (3)$$

d1 represents the vector representation of the best answer, *S*1, represents the best answer, and *d*2 represents the vector representation of the participant's answer, and *S*2, represents the participant answer.

LSA-based automated scoring equation (Equation 4) is presented as follows:

$$scoreitem = \text{sim}(s1,s2) * sitem$$

sitem represents the maximum score in each item, sim(S1, S2) represents the semantic similarity between the correct answer and the participant's answer and scoreitem represents the participant's sentence construction score in each item. Almost, the number of dimensions maintain in LSA is an experimental issue. We conducted the experiments under different dimensions in the experiment section.

Algorithm 1 have the inputs include a term-document matrix, several product-feature seeds, the reduced dimensionality in SVD operation, and the number of mined features for each seed. In Algorithm 1, the term-document matrix is performed on lines 3 and 4 by SVD operation, the similarities between the seed product-feature vector and pair wise, the other term vectors. To gain the term-vector representation of product feature the function **getTermVectorFromTermDocMatrix** is used. The similarities between the seed and the other terms are shown in line 7.sorting in non increasing order is the next step to gain product features which are on the top list.

4. Discussion

The sentiment classification is done based on unigrams, bigrams, negation, frequency and presence features and location in LSA algorithm. Naive Bayes classifier is a simple method to count number of features. In this technique the less training data will be required because it converge quickly. But the main disadvantage is it can not become skilled at interactions between the features. SVM gives high accuracy and work well even if data is not linearly separable in base feature space. SVMs use overfitting protection, which does not necessarily depend on the number of features; they have the potential to handle these large feature spaces.SVM are very well suited for text categorization. Another advantage of SVMs over the conventional methods is their robustness. SVMs show good performance in all experiments, avoiding disastrous failure. Maximum entropy gives better accuracy than the Naive Bayes but less than the SVM. Maximum entropy reduces classification error by more than 40% compared to naive Bayes. There is evidence that basic maximum entropy suffers from overfitting and poor feature selection. One promising aspect of maximum entropy is that it naturally

handles overlapping features. For example, we could supplement our word features with bigram, phrase, and even non-text features. Maximum entropy will not be hurt by strong independence assumptions, as would naive Bayes with these features. LSA-based system can identify a related term set for each seed. LSA based filtering mechanism is planned to employ the semantically related terms to reduce the size of summary. Only the sentences containing the terms will be presented to users. Moreover, the LSA based product feature-identification approach could be generalized to other product-review domains, since the linear algebra SVD (Singular Value Decomposition) operation could be applied to any language.

5. Conclusion

In this paper, a survey of review summarization is done by using several machine learning methodologies. One of the most intimidating tasks in opinion mining is feature extraction from a training data. Once features and synonyms are extracted the next task is to determine the review is positive or negative. Based on this the review summarization is done.

Feature based summarization can be done by Naive Bayes approach or Maximum Entropy, or SVM, or LSA algorithm. In this paper from all above techniques the movie review and summarization is done by LSA algorithm. Because the efficiency and accuracy is considered for Product features and opinion words. A frequency criterion is used to summarize the review.SVM model is used to categorize the positive and negative reviews. The same method can be used for other domain also.

REFERENCES

[1] Chien-Liang Liu, Wen-Hoar Hsaio, Chia-Hoang Lee, Gen-Chi Lu, and Emery Jou "Movie Rating and Review Summarization in Mobile Environment" MAY 2012

[2] Chen-Huei Liao Effectiveness of Automated Chinese Sentence Scoring with Latent Semantic Analysis, April 2012

[3] David Shaw "Opinion Mining of Movie Reviews" 2009

[4] A Maximum Entropy Model for Product Feature Extraction in online Customer Reviews 2008.

[5] A Feature Dependent Method for Opinion Mining and Classification 2008

[6] L. Zhuang, F. Jing, and X.-Y. Zhu, "Movie review mining and summarization," in Proc. *15th ACM Int. Conf. Inf. Knowl. Manage.*, 2006, pp. 43–50.

[7] B. Pang, L. Lee, and S.Vaithyanathan, "Thumbs up?: Sentiment classification using machine learning techniques," in *Proc. ACL-02 Conf. Empirical Methods Natural Lang. Process.*, 2002, pp. 79–86.

[8] Yohan Jo "Aspect and Sentiment Unification Model for Online Review Analysis"

[9] Naveed Anwer, Ayesha Rashid, SyedHassan Feature Based Opinion mining of Online Free Format Customer Reviews Using Frequency Distribution Bayesian Statistics

[10] Gamgarn Somprasertsri, Pattarachai Lalitrojwong A Maximum Entropy Model for Product Feature Extraction in Online Customer Reviews 2008

[11] Feature Based Opinion Mining of Online Free Format Customer Reviews Using Frequency Distribution and Bayesian Statistics

[12] Multi-Document Summarization of Product Reviews 2012

[13] Siva Ramakrishna Reddy, Ajay R. Dani, Sentiment classification of Text Reviews Using Novel Feature Selection with Reduced Over-fitting.