Buletin Teknik Elektro dan Informatika
(Bulletin of Electrical Engineering and Informatics)
Vol. 2, No. 3, September 2013, pp. 169~176
ISSN: 2089-3191

Relevant Words Extraction Method for Recommendation System

Naw Naw*[1], Ei Ei Hlaing[2]
University of Technology, Yatanarpon Cyber City, Pyin Oo Lwin, Myanmar
*Corresponding author, e-mail: nawnaw1986@gmail.com[1], shmgmg@gmail.com[2]

Abstract

Nowadays, E-commerce is very popular because of information explosion. Text mining is also important for information extraction. Users are more preferable to use the convenience system from many sources such as through web pages, email, social network and so on. This system proposed the relevant words extraction method for car recommendation system from user email. In relevant words extraction, this system proposed the Rule-based approach in Compiling Technique. Context-free grammar is the most suitable for relevant words extraction. Recommendation System (RS) is a most popular tool that helps users to recommend according to their interests. This system implements efficient recommendation system by using proposed key extraction algorithm, Content-based Filtering (CBF) method and Jaccard Coefficient that will help the users who want to buy the car by providing relevant car information.

Keywords: *relevant words extraction, rule-based, compiling technique, recommendation system, content-based filtering, Jaccard coefficient*

1. Introduction

Most E-commerce sites compelled to get the customers in various ways. Some used the information retrieval system. But this can only provide the information for user by matching with user's input. There are many constraints in information retrieval system that users have the little chance to select their desires. So the ecommerce sites should let the users to input with free text as they like. The users do not need to worry about the structures and grammars. Most researches tried to develop the autonomous system which recognizes the user's desires. But, there are some problems in grabbing of user input. To solve this problem, one efficient way is to use text mining techniques. Text mining uncovers the underlying themes or concepts that are contained in large document collections. Text mining applications have two phases: exploring the textual data for its content and then using discovered information to improve the existing processes. Both are important and can be referred to as descriptive mining and predictive mining. Descriptive mining involves discovering the themes and concepts that exist in a textual collection. Predictive modeling involves examining past data to predict future results [9]. In Descriptive mining, the unstructured texts are difficult to extract the useful data because of the richness and ambiguity of natural language. So this system proposed the relevant words extraction method based on context-free grammar. In predictive mining, this system implements the content-based recommender system by using the output keys of relevant words extraction method.

2. Recommendation System

2.1. Text Mining

In Text mining, text is unstructured, amorphous, and contains information many different levels. Most text mining techniques tried to extract useful data from unstructured text mostly written in natural language. The automatic extraction of information from text is to produce structured output that can be put into a database or others. There is needed for any information extraction to gather, preprocess and extract the keywords based on corpus-oriented methods or document-oriented methods [6]. Most systems use machine learning techniques and a variety of features such as Support Vector Machine, K means. Some systems use Rule-based technique.

Received January 8, 2013; Revised May 6, 2013; Accepted May 27, 2013

The accurate level between these systems is different. Rule based technique gets 92% score. But Machine learning based approaches were able to achieve around 70% breakeven [15]. So this system proposed the rule-based text mining technique in extraction relevant words based on compiling technique. The context-free methods are powerful enough to describe almost all of the so-called syntactic features of programming languages. Indeed context-free grammars are often used in language manuals [19]. The output of the relevant words extraction method can be applied to the recommendation system.

2.2. Content Based Filtering

Recommender systems help customers to find what they really want. So this meets the requirements of customers in a short time. It helps users to find information, products, or by aggregating and analyzing suggestions from other users' activities. CBF techniques are developed for information retrieval and information filtering research [14]. In the CBF system, each user can operate independently and will be recommended the most closely information of the items according to their request.

2.3. Similarity Measuring

It is needed to manipulate the similarity between the contents. There are many similarity methods used in content-based recommender system. But Jaccard Coefficient is most proper method for this proposed system. The Jaccard Coefficient is a similarity measure which ranges between 0 and 1. Similarity value 1 means the two objects are the same and 0 means they are completely different. The nearer to 1 is, the more similar between two objects. Jaccard can resolve their various similarity values in different similarity level.

$$S_{jaccard} = M_{Key} / T \quad (1)$$

Where,

M_{Key} = Match keys between key pairs and total attributes
T = Total Attributes

This proposed system intend to save time in extracting information from web application by promoting the performance of e-commence. Nowadays, the Myanmar's citizens interested to buy cars. This system tried to satisfy the customers dealing with finding product that they desired. The main purpose of the system is to provide the relevant words extraction method which can enhance the content based filtering.

3. Related Work

Latha K, Kalimuthu S, Dr Rajaram R, proposed Information Extraction from Biomedical Literature using Text Mining Framework. There are three steps in this paper. Text gathering: The documents are collected from the existing biomedical databases. Thousand-sample sets of documents are collected from various biological domains and these documents are analyzed and given as the input to the second stage. Text preprocessing: The above documents are preprocessed for decreasing the workload in the Data analysis stage. Data analysis: This phase focuses on analyzing the documents of the previous phase by using support vector machine (SVM). But this research wasted the time to recognized the every terms that are not concerned with biomedical information [13].

Ashwini Madane proposed Identifying Keywords and Key Phrases. A new algorithm (Kea) is used for automatically extracting key phrases from text. Step 1 (Preprocessing): stop word removing, tokenization. Step 2 (Candidate Identification): Kea then considers all the subsequences in each line and determines which of these suitable candidate phrases are. Step 3 (Determining Candidate Phrases): Use stemming method (Lovins). Step 4 (Feature Calculation): Kea builds a document frequency file. Use TF-IDF technique. But it takes too much time in candidate identification [1].

IAN H. WITTE proposed the key phrase extraction method by using machine learning approach. To explore the phrase is "key phrase" or "non-key phrase"; there are two attributes to be considered. The first is the distance into the document of the phrase's first appearance. The second, and more influential, is the "term frequency times inverse document frequency," or

TF_IDF, score of a phrase. The classifier uses the Naïve Bayes method to calculate the two attributes. This is simple and effective but the training time is a critical factor in this system [16].

Stuart Rose, Dave Engel, Nick Cramer and Wendy Cowley proposed Rapid Automatic Keyword Extraction (RAKE), an unsupervised, domain-independent, and language-independent method for extracting keywords from individual documents. Firstly, RAKE removed the stop words from the document. And then, it defined the candidate keyword according to the domain requirements. RAKE calculated the word score based on the degree and frequency of word vertices in the graph: (1) word frequency (*freq (w)*), (2) word degree (*deg (w)*), and (3) ratio of degree to frequency (*deg (w)/freq (w)*). RAKE achieves a high recall but it experiencing a drop in precision [6].

Bernd Ludwig and Stefan Mandl proposed the transformation method from the expert centered knowledge to the user centered knowledge by using TF-IDF approach. That method tried to identify the topic by looking up the words in the document. The results were applied for the TV recommendation system. That method depended on the quality of matrices. To understand in more detail how user opinions influence the parameters. That system has to be trained to be user adaptive [7].

BalaKrishna Kolluru, Sirintra Nakjang, Robert P. Hirt, Anil Wipat, and Sophia Ananiadou proposed a Conditional Random Field (CRF) technique to extract the mention of microorganisms, habitats and the inter-relation between organisms and their habitats. Results indicate a good performance for extraction of microorganisms and the relation extraction aspects of the task (with a precision of over 80%), while habitat recognition is only moderate (a precision of about 65%). There are three principles in the workflow: PDF-to-text convertor, Named entity recognizer, and CRF-component. The disadvantage is pdf-to-text conversion can be quite noisy and this implicitly affects any sentence-based relation extraction algorithms [4].

4. Proposed System

4.1. Proposed System Framework

There are three main basic stages for recommendation system.

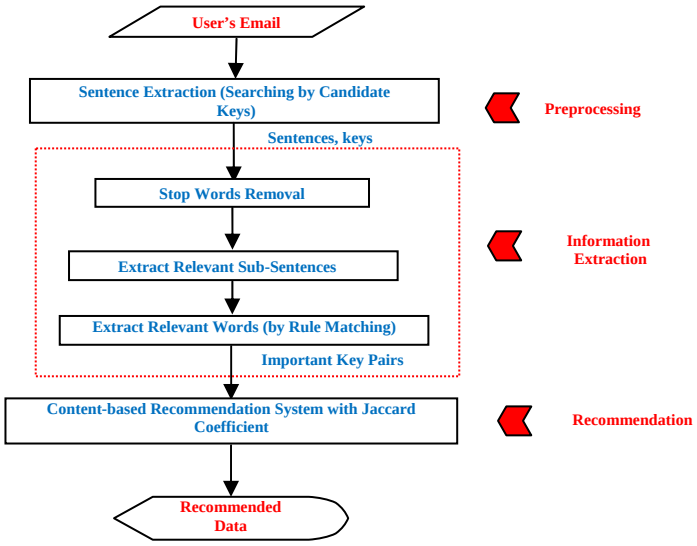

Figure 1. System Framework

Preprocessing: If the user sends the order from email, the system will extract the relevance sentences that concerned with their desired car information such as type of car, model number, amount of money they can afford, year, color, mileage, etc. That will be described detail in section (4.2.1). Not relevance sentences will be ignored in this stage. For example:

How are you? ⇐ Not relevance Sentences

Like this above sentence will be ignored in preprocessing. After extracting the relevance sentences, we need to look up the main verb to recognize the positive sentence or negative sentence. For example:

I like the white color. ⇐ Positive Sentence
I don't like the silver color ⇐ Negative Sentence

After distinguishing the sentences' positive or negative, the negative sentences will be ignored again.

Information Extraction: Firstly, we need to do is stop words removal. After stop words removing, the system can extract the relevant sub-sentences. The proposed algorithms for these steps will be described detail in following section (4.2.2). The resulted sub-sentences are matched with the proposed rules in section (4.3) that can develop the important key pairs.
Recommendation: After the system gets the important key pairs, content-based filtering approach will implement with Jaccard Coefficient method. Each user's prefer is different. If the available car is not exactly same with user desire, system needs to recommend the most similar results for the user requirements. This system proposed the weight value of important keys to provide Jaccard Coefficient method for the most similar results. So, defining weight value is very important in this recommendation system. Finally, the system will generate the recommendation list for the users according to their requests.

4.2. Proposed Algorithms for Information Extraction

4.2.1. Sentence Extraction

```
Input    :   Email's content
Output   :   Relevance Sentences
Process  :
  Process For all Sentences
      Sentence List ← Search the relevance sentence by comparing with candidate keys
  End For
```

4.2.2. Key Extraction

```
Input    :   Sentences List
Output   :   Keys
Process  :
  Process Sentence-Level Identification
  For each processed sentence
    Keys ← important-key-finder (sentence, automobile-key)
  End For
```

Important-key-finder (sentence, automobile key)

```
begin
    sub-sentence← stop word removal
     For each sub-sentence
      For each  RULE
        Rule matching process
          If matched rule then
            generate key-pair
          end If
      end For
    end For
  return generated key-pair
end
```

4.3. Rules for Proposed System

The following sample rules will provide the above key-finder algorithm. In this system, there are eight rules to provide important keys for recommendation of Jaccard Coefficient Similarity method.

Rule 1 ⇒ (< article> | <number>) <typeOf>
<article> ⇒ a | an | one | two | three | four | five | six | seven | eight | nine | ten
<number> ⇒ 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<typeOf> ⇒ Alfa | AlfaRomeo | Audi | BMW | Chrysler | Citroen | car | Daihatsu | Fiat | Ford | GM | Hino | Honda | Isuzu | Mazda | MarkII | Matsubishi | Mercedes Benz | Opel | Peugeot | Renaut | Rover | Nissan | Subaru | Suzuki | Toyota | Volkswagen | Volvo

Rule 2 ⇒ <preposition> <year>
<preposition> ⇒ at| ago| for| in| since
<year> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <year> <year>
⇒ ε

Rule 3 ⇒ <number> <notation>
<number> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <number> <number>
⇒ ε
<notation> ⇒ Lks | l | L | kyats | $

Rule 4 ⇒ <color>
<color> ⇒ blue | light blue | dark blue | black| grey | pearl | red | silver | white

Rule 5 ⇒ <mileage>
<mileage> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <mileage> <mileage>
⇒ ε

Rule 6 ⇒ <model number>
<model number>⇒ 86 | Allex | Allion | Alphard | Altezza | Altezza Wagon | Aqau | Carinaed | Cavalier | Celica| Century | Chaser | Coaster | Corolla | Corolla2 | GRX120 | GX110 | JRX110 | Umax_Toy

Rule 7 ⇒ <engine>
<engine> ⇒ 06 | AT | IAT | FA | F6

Rule 8 ⇒ <equipment>
<equipment> ⇒ 1500cc | 1800cc | 2000cc | 2500cc

4.4. Jaccard Coefficient Method

When the system gets the important key pairs, the similarity value is provided by using Jaccard Coefficient method. The basic idea behind this approach is degree of similarity or vibration of user desired keys is calculated for different weight of available selling car. Different weight of similarity value is:

$$S_{jaccard} = M_{Key} / T$$

In here, calculated different similarity weight value is determined by threshold 0.5. If the threshold values less than 0.5, unrelated recommended lists will be shown to the users. The accuracy of the recommended lists will be higher if the threshold value is greater than or equal 0.5. If it is only greater than 0.5, the **sparsity** problem will be occurred.

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [MarkII, 130 Lks, Pearl, GRX120]
  M_{Key} = 4
  T = 4
  $S_{Jaccard}$ = 4/4 = 1 ≥ 0.5

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [MarkII, 130 Lks, Silver, GRX120]
  M_{Key} = 3
  T = 5
  $S_{Jaccard}$ = 3/5 = 0.6 ≥ 0.5

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [MarkII, 150 Lks, Grey, GRX120]
  M_{Key} = 2
  T = 6
  $S_{Jaccard}$ = 2/6 =0.33 < 0.5

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [MarkII, 130 Lks, Silver, GX110]
  M_{Key} = 3
  T = 5
  $S_{Jaccard}$ = 3/5 = 0.6 ≥ 0.5

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [Nissan, 130 Lks, Black, JRX110]
  M_{Key} = 1
  T = 7
  $S_{Jaccard}$ = 1/7 =0.14 < 0.5

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [MarkII, 125 Lks, Pearl, GRX120]
  M_{Key} = 3
  T = 5
  $S_{Jaccard}$ = 3/5 = 0.6 ≥ 0.5

- Key pairs = [MarkII, 130 Lks, Pearl, GRX120]
  Attributes = [Honda, 80 Lks, Blue, 86]
  M_{Key} = 0
  T = 8
  $S_{Jaccard}$ = 0/8 =0 < 0.5

So, the proposed system generates the recommended list, if the weight of the similarity values is greater than or equal 0.5.

5. Experimental Results

These experiments are evaluated on 682 email letters. This measurement is based on precision and recall that are two most frequent and basic measures for information retrieval effectiveness. Precision (P) is the fraction of retrieved key phrases that are relevant. Recall (R) is the fraction of relevant key phrases that are retrieved [3].

$$\text{Precision (P)} = T_p / (T_p + F_p) \quad (2)$$

$$\text{Recall (R)} = T_p / (T_p + F_n) \quad (3)$$

Where,

T_p = True positive
F_p = False positive
F_n = False negative
T_n = True negative

	Relevant	**Nonrelevant**
Retrieved	True positive	False positive
Not Retrieved	False negative	True negative

F_1 tries to combine precision and recall into a single score by calculating different types of means of both metrics. The F_1 is calculated as the standard harmonic mean of precision and recall:

$$F_1 = 2*P*R / (P+R) \quad (4)$$

Table 1. Experimental Results

Methods	**Extracted Keys**	**T_n**	**P**	**R**	**F**
Proposed method	6028	3015	0.66	0.56	0.61
Machine Learning (using Naïve Bayes)	5893	2950	0.61	0.57	0.59

Table 1 presents the experiment results for proposed relevance key words retrieval and machine learning approach. For each method which corresponding rows in a table, the above information is shown: the total number of extracted key words, relevance extracted key words, precision, recall and F1-measure. This proposed method has the higher precision and less recall than machine learning approach while higher F1-measure than machine learning. Moreover, this method does not require training set as machine learning approach.

6. Conclusion

Today, our government opens the car market. So there are many demands on car. This system helps the user who wants to know the car information from their email and recommends the closely relevant car information such as type of car, model, year, color, price and mileage for the requested user. Most information extraction systems use the machine learning technique. So they are very complex and time consuming. This proposed system can reduce these complexes by using Compiling technique. This system can decrease the preprocessing time with sentence level identification. Recommendation system performance will increase by combining this information extraction system.

References

[1] Ashwini Madane. Identifying Keywords and Key Phrases. *IJSCE*. 2012; 2(3).
[2] Hany Mahgoub, Dietmar Rösner, Nabil Ismail, Fawzy Torkey. A Text Mining Technique Using Association Rules Extraction. *International Journal of Information and Mathematic Sciences*. 2008; 4(1).
[3] Zakaria Elberrichi, Abdelattif Rahmoun, Mohamed Amine Bentaalah. Using WordNet for Text Categorization. *The International Arab Journal of Information Technology*. 2008.
[4] Latha .K, Kalimuthu.S, Dr.Rajaram.R. Information Extraction from Biomedical Literature using Text Mining Framework. *IJISE*. USA. 2007; 1(1).
[5] Bernd Ludwig, Stefan Mandl. Centering Information Retrieval to the User. *RSTI – RIA*. 2010; 24: 95-118.

[6] BalaKrishna Kolluru, Sirintra Nakjang, Robert P Hirt, Anil Wipat, Sophia Ananiadou. Automatic extraction of microorganisms and their habitats from free text using text mining workflows. *Journal of Integrative Bioinformatics*. 2011; 8(2).
[7] Huda Yasin, Mohsin Mohammad Yasin, Farah Mohammad Yasin. Automated Multiple Related Documents Summarization via Jaccard's Coefficient. *International Journal of Computer Applications*. 2011; 13(3).
[8] Yize Li, Jiazhong Nie, Yi Zhang, Bingqing Wang. *Contextual Recommendation based on Text Mining*. 2012.
[9] Gunnar Schröder, Maik Thiele, Wolfgang Lehner. *Setting Goals and Choosing Metrics for Recommender System Evaluations*. 2012.
[10] Rares Vernica, Michael J.Carey, Chan Li. *Efficient Parallel Set-Similarity Joins Using MapReduce*. 2010.
[11] Stuart Rose, Dave Engel, Nick Cramer, Wendy Cowley. *Automatic Keyword Extraction from Individual Documents*. 2010.
[12] Raymond J Mooney, Razvan Bunescu. *Mining Knowledge from Text Using Information Extraction*. 2009.
[13] Munyaradzi Chiwara, Mahmoud Al-Ayyoub, Mohammad Sajjad, Hossain, Rajan Gupta. *CSE 634 – Data Mining: Text Mining*. 2009.
[14] Francesco Ricci. *Content-Based Filtering and Hybrid Systems*. 2005.
[15] Sundar Varadarajan, Kas Kasravi, Ronen Feldman. *Text-Mining: Application Development Challenges*. 2004.
[16] IAN H. WITTEN. *Adaptive Text Mining: Inferring Structure from Sequences*. 2003; 0(0): 1–23.
[17] Haralampos Karanikas, Christos Tjortjis, Babis Theodoulidis. *An Approach to Text Mining using Information Extraction*. 2001.
[18] Damian Fijalkowski, Radolsaw Zatoka. *Architecture of a Web Recommender System using Social Network User Profiles for E-commerce*. Proceedings of the Federated Conference on Computer Science and Information Systems 2001: 287-290.
[19] SAS Institute. Getting Started with SAS® Text Miner 4.1. First Printing. USA. 2009.
[20] PM Lewis II, DJ Rosenkrantz, RE Stearns. Complier Design Theory. Third Printing. Philippines. 1978: 135-179.