Relevant Words Extraction Method for Web Recommender System

Naw Naw, and Ei Ei Hlaing

***Abstract*— Web is the most popular tool for the people in 21st century. Users search their desired information and knowledge from the web in different ways. But they get the constraints to obtain their information. They are more preferable to use the convenience system from many sources such as web pages, email, social network and so on. Sometime, demand information and received information do not match. Information Extraction is able to solve these inaccurate problems. Moreover, some users vacillate which one is useful or useless. So many web sites tried to recommend users the matched one. This system implements effective web recommender system by using proposed relevant words extraction algorithm (Key Finder Method), Content-based filtering approach and Jaccard Coefficient method. This paper developed the simple rules to extract the entities and their relationships. Context- free grammar is the most suitable one for creating these rules. The proposed algorithm extracts relevant words in the short time from the user's free text. The extracted relevant words are used for recommendation system that will help the customers who want to buy the car by providing car information related with their request.

Keywords—information extraction, relevant words extraction, rule-based, context-free grammar, recommendation system, content-based filtering, Jaccard Coefficient

I. Introduction

INFORMATION Extraction is a weapon used by most famous web sites that compelled to get the customers in various ways. They can provide accurate information matched with user demands with the help of Information Extraction. Users can describe their requirements with free text. They do not need to worry about the structures and grammars. Developers tried to solve the input free text to get the structure data format which can be useful for the database. Most researches tried to develop the autonomous system which recognizes the user's desires. But, there are some problems in grabbing of user input. Information Extraction can solve this problem by using efficient approaches. There are two approaches in information extraction. They are knowledge engineering approach and automatic training approach. Knowledge engineering approach uses grammars and rules by hand using knowledge of the application domain. Automatic training approach uses machine learning technique. So it needs many training data to be faster than above approach.

The proposed system uses knowledge engineering approach to extract the relevant car information. By using this approach, we need one expert to generate rules. In this system, rules generated area is very easy to understand for everyone. The main advantage is that we can reduce the training time and constructing time for the classifier as automatic training approach.

II. Information Extraction

A. Knowledge Engineering Approach

This approach is processed by the rules and grammars. The rules and grammars are developed by the knowledge engineer who is familiar with the application domain. Typically the knowledge engineer constructs the rules that are related with the current domain relevant text. The performance of the system is based on the knowledge engineer. He can control the system with his skill. However, this approach requires test and debug cycle. Knowledge engineer has to analyze the rules and debug the rules when it is required. Knowledge engineering approach becomes the problematic if the knowledge engineer has the lack of the domain knowledge [16].

B. Automatic Training Approach

The Automatic Training Approach is quite different. This approach does not need to control by the expert who has details knowledge of the domain and write the rules and grammars. The input texts are required to annotate and extract to provide the appropriate information. The large amount of training data is needed in this approach. If the system trains with the large amount of data, the accuracy of the system will be improved.

The main advantage of this approach is that there is no need to have any expertise to build the Information Extraction (IES). This would allow people without knowledge about the process of building an IES or about the process of creating rules. Another significant advantage is the easy adaptation to new domains. The disadvantage of the automatic training approach is that it is based on training data. Training data may be in short supply, or difficult and expensive to obtain. If the domain is complex to annotate, the system will be slower, expensive and difficult. The difference between two approaches is that knowledge engineering approach focuses on

Naw Naw is with the University of Technology (Yatanarpon Cyber City), Pyin Oo Lwin, Myanmar (nawnaw1986@gmail.com).

Ei Ei Hlaing, was with University of Computer Studies, Taung Ngu, Myanmar. (shmgmg@gmail.com).

the rule production and automatic training approach focuses on producing training data [17].

This system proposed the rules concerned with the car information. The domain knowledge engineer is skillful about this area and the rules are easy to understand and use. The performance level is comparable to the complex automatic training approach. There is no need to have the complex training data. The rules are easy to build and modify as the situation changes. The relevant data can be extracted with the help of these rules. The extracted data is ready to use for the recommendation system. Recommendation system uses CBS to recommend the users about the car information. So the accuracy and completeness level of this system is also increased.

C. Content Based Filtering

Recommendation System (RS) is a most popular tool that helps users to recommend according to their interests. RS 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 [11]. In the CBF system, each user can operate independently and will be recommended the most closely information of the items according to their request.

D. 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.

$$Sim_{jaccard} = S_{Key} / S_{Key} + D_{Key} \quad (1)$$

Where,

S_{Key} = Number of Similar keys

D_{Key} = Number of Dissimilar keys

E. Context Free Grammar

We use context free grammar in extracting the key phrases. A context-free grammar is a 4-tuple (V, Σ, R, S) where:

1. V is a finite set of symbols called the variables or nonterminals.
2. Σ is a finite set of symbols, disjoint from V, called terminals.
3. R is a finite set of rules (or specification rules) of the form lhs → rhs, where lhs ∈ V, rhs ∈ (V ∪ Σ)
4. S ∈ V is the start variable [15]

III. RELATED WORKS

Mahmudul Sheikh and Sumali Conlon proposed a rule-based system to extract financial information for aiding investment decisions. They used the Greedy Search algorithm and a similar model trained by the Tabu Search algorithm. Precision and Recall of the information system is higher than early approaches. But there are many complexes in building the rules [19].

Benjamin Rosendfeld, Ronen Feldman and Moshe Fresko proposed TEG system (a hybrid approach to information extraction). They combine the knowledge engineering approach and automatic training approach. The simple rule and smaller training data can be comparable to other pure systems. But the training time is the considerable fact for that system [18].

Ignazio Gallo and Elisabetta Binghi proposed Information Extraction and Classification from Free Text Using a Neural Approach. There are two steps in their system. Firstly, they use the matcher that separate the input sentence as token. Then the classifier tried to classify the class label of each token. But the construction of knowledge base and thesaurus are very complex and need the help the expert [21].

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].

We formerly proposed relevant words extraction algorithm in recommender system. Our system was comparable to naïve bayes method with high precision, recall and F1. But the rules merely produced for entity extraction [22].

IV. PROPOSED SYSTEM

A. Proposed System Framework

User: In our Online Automobile System, user can request their desired information from the system in free text. Then, He waits the system's reply in few seconds.

Key Finder: Key finder is responsible for changing to structured data. When the key finder receives request from user, it tried to extract the relevant automobile keyphrase from this request. In sentence level extraction, each sentence from user request can be identified as relevant or not with candidate keys. Knowledge engineer needs to predefine the candidate key. The more candidate system defined, the more accurate information can be delivered to the user. However, the system discards non relevant sentence as following.

I am a manager from Shwe
Automobile Company. ⇐ Not relevant Sentences

After getting the relevant sentence, key finder check further to distinguish between positive or negative sentence. If it found the negative terms, it discarded this sentence. Negative term will be predefined by the system such as not, hate etc.

I want to ride a light blue car. ⇐ Positive Sentence
I don't like Ford cars. ⇐ Negative Sentence
I hate red color. ⇐ Negative Sentence

When positive sentences are found, key finder does more things to produce relevant key pairs. Key pairs mean key and value. System uses context free grammar for key pair. Rule matching process can work easily and save the time in finding key words. The performance level of the system is totally depending on key finder.

Recommender: The relevant key pairs are received by recommender; it uses the content based technique. In content based, Jaccard similarity method will be worked. Weight value the most significant factor in calculating similar values. This value can change according to the application domain. In this system, brand is the most important fact to be greater weight value than others. In recommendation, the lists may be according to their priority value. User can filter their needs by deciding which can help them. The performance of recommender can be increased by using the output of information extraction system.

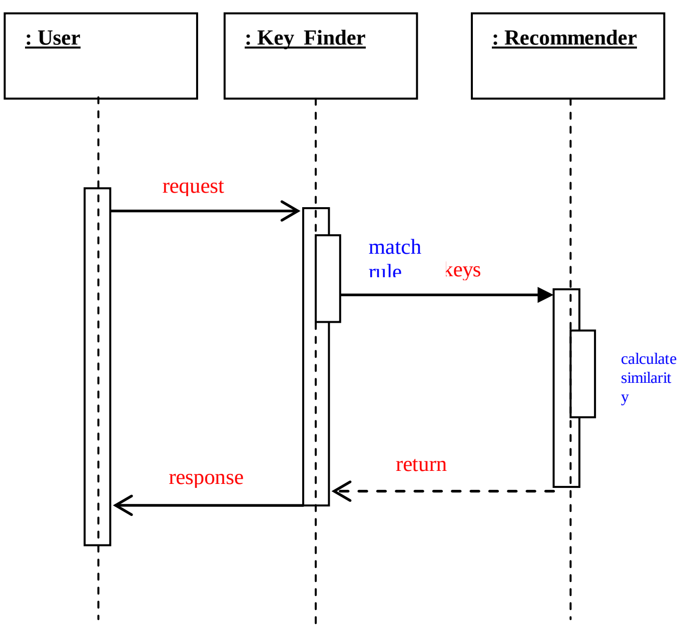

Fig. 1 System Framework

Rules are starting points. Non terminals and terminals worked together to produce key phrases. Non terminals are brand, article, number, preposition and so on. They can expand to any single token. 1, a, audi and so on are non terminals. Those rules are processed from left hand side to right hand side format. In key extraction, non terminals are investigated whether they are matched with our effective database system. And some non terminals have the optional value such as number, notation and so on.

This proposed system intends to save time in extracting information from user request. Nowadays, the Myanmar's citizens interested to buy cars. This system tried to satisfy the customers dealing with finding product that they desired. In extracting information, there are two types of extraction tasks. Firstly entity extraction works to extract the type, model, body number and so on. Relationship/link extraction task works to find the relationship between above entities. The simple rule structure makes the system more portable and easy to modify according to current conditions.

B. Proposed Algorithms for Information Extraction

1) Sentence Extraction (Preprocessing)

Input : User's free text
Output : Relevant Sentences
Process :
Process For all Sentences
Sentence Lists ← Search the relevant sentence by comparing with candidate keys
End For

2) Key Extraction (Information Extraction)

Input : Sentence Lists
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
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

3) Recommended Data (Recommendation)

Input : Keys
Output : Recommended Data
Process :
S_{Key} = Number of Similar keys
D_{Key} = Number of Dissimilar keys

$Sim_{jaccard} = S_{Key} / S_{Key} + D_{Key}$
if $Sim_{jaccard} < 0.5$ then
    discard
else
    return recommendated data
end if

C. Rules for Proposed System

The following sample rules will provide the above key-finder algorithm. These rules provide the entity extraction and relationship extraction. In entity extraction, Rule 1 to 13 (expect 6 and 10) tried to extract for each entity such as brands of the cars, production year, the color, mileage, model number, engine power, oil type and body number, the money that can be afforded by the customers, types of car, door type and their driving system.

In relationship extraction, some entities extracted by the rules have the relationship between them. For example, Rule 1 and Rule 4 have the relationship in some case such as the extracted entity is “a red car”. Relationship extraction also performs in Rule 4 and Rule 6. For example, when the user requests “a green toyota2008 car”, Rule 4 and Rule 6 need to work together. Rule 1 has the relationship with both Rule 4 and 6. Instead of using Rule 1 with 4 and 6, Rule 10 performs as the relationship extraction in this case. Moreover, model number is defined by the combination of the brand and production year. So Rule 6 performs the extraction behalf of Rule 1 and Rule 2. As you see, rule 6 and 10 operate as a relationship extraction.

Rule 1 ⇒ (< article> | <number>) <brandOf>
<article> ⇒ a | an | one | two | three | four | five | six | seven | eight | nine | ten
<number> ⇒ 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
<brandOf> ⇒ AlfaRomeo | Audi | BMW | Chrysler | Citroen | car | Daihatsu | Honda | Isuzu | Mazda | Matsubishi | Nissan | Subaru | Suzuki | Toyota

Rule 2 ⇒ <preposition> <year> <notation> <preposition>
<preposition> ⇒ at | <ago> | for | in | <since> | <later> | early| <before> | <after> | last | from | to | during | till | until | within | up to | past | between | by
⇒ #
<ago> ⇒ <
<before> ⇒ <
<after> ⇒ >
<later> ⇒ >
<since> ⇒ >
<notation> ⇒ ‘s | s
<notation> ⇒ #
<year> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <year> <year>
⇒ #

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

Rule 4 ⇒ <color>
<color> ⇒ blue | light blue | dark blue | black| grey | pearl | red | silver | white | yellow | green | indigo | purple | light yellow | dark yellow | wine | wine-red | aqua | light green | dark green | beige | gold | cream | sugar cane

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

Rule 6 ⇒ <brandOf> <year>
<brandOf> ⇒ AlfaRomeo | Audi | BMW | Chrysler | Citroen | car | Daihatsu | Honda | Isuzu | Mazda | Matsubishi | Nissan | Subaru | Suzuki | Toyota
<year> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <year> <year>
⇒ #

Rule 7 ⇒ <engine> <notation>
<engine> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <engine> <engine>
⇒ #
<notation> ⇒ cc

Rule 8 ⇒ <oil>
<oil> ⇒ diesel | petrol

Rule 9 ⇒ <chassis_id>
<chassis_id> ⇒ TRY230 | XZU554 | XZU605 | XZB50 | KDH206 | KDY231 | XZU508 | TRX200 | XZU344 | XZU378 | XZU308 | XZU414 | XZU388 | XZU334 | XZU348 | NCP55V | XZU424 | XZU368 | KDH201 | KDY221 | KDY290 | NCP55 | RZU300 | TRH221 | XZU338 | XZU488 | XZU504 | KDY220 | KM70 | XZU304 | NCP51 | KDY230 | KDH200 | KDY23 | KDY240 | KM80 | XZB51 | XZU346 | XZU401 | XZU413 | XZB40 | XZU423 | KM75 | TRY220 | XZU306 | XZU421 | XZU311 | XZU301 | XZU341 | XZU411 | KDY280 | KM85 | XZU331 | XZU351 | BU301

http://dx.doi.org/10.15242/IIE.E0314068

Rule 10 ⇒ <article> <color> <brandOf> <year>
<article> ⇒ a | an | one | two | three | four | five | six | seven | eight | nine | ten
<color> ⇒ blue | light blue | dark blue | black| grey | pearl | red | silver | white | yellow | green | indigo | purple | light yellow | dark yellow | wine | wine-red | aqua | light green | dark green | beige | gold | cream | sugar cane
<brandOf> ⇒ AlfaRomeo | Audi | BMW | Chrysler | Citroen | car | Daihatsu | Honda | Isuzu | Mazda | Matsubishi | Nissan | Subaru | Suzuki | Toyota
<brandOf> ⇒ #
<year> ⇒ 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
⇒ <year> <year>
⇒ #

Rule 11 ⇒ <typeOf>
<typeOf> ⇒ bus |container | crane | cruiser |van | light truck | truck |machine | minibus | light ace | town ace | express | sedan | sport | wagon | pick up | double pick up | markii | super custom| super saloon | se

Rule 12 ⇒ <door>
<door> ⇒ 1 | 2 | 3 | 4

Rule 13 ⇒ <driveSystem>
<driveSystem> ⇒ left | right

D. 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 priority of available selling car. For Example, Brand takes place in first position priority.

The system works as a car agent. If the user asks for many specifications, agent must try to find the same one exactly. However, agent can provide many data if the user describe his needs in general. Jaccard is strong to produce the accurate recommended lists that are two-third similar with user's specifications.

$$Sim_{jaccard} = S_{Key} / S_{Key} + D_{Key}$$

In here, calculated different similarity value is determined by threshold 0.5 for both cases. 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. The following is the sample of how system operates on the user request:

- Key pairs = [light blue, toyota, 1500 to 2000, >2005]
  Attributes = [blue, toyota, 1800, 2006]
  M_{key} = 3
  T_{Key} = 5
  $S_{Jaccard}$ = $3/5=0.6 \geq 0.5$

- Key pairs = [light blue, toyota, 1500 to 2000, >2005]
  Attributes = [dark blue, toyota, 1500, 2007]
  M_{key} = 3
  T_{Key} = 5
  $S_{Jaccard}$ = $3/5 = 0.6 \geq 0.5$

- Key pairs = [light blue, toyota, 1500 to 2000, >2005]
  Attributes = [light blue, toyota, 1500, 2004]
  M_{Key} = 3
  T_{Key} = 5
  $S_{Jaccard}$ = $3/5 =0.6 \geq 0.5$

- Key pairs = [light blue, toyota, 1500 to 2000, >2005]
  Attributes = [green, toyota, 1300, 2004]
  M_{Key} = 1
  T_{Key} = 7
  $S_{Jaccard}$ = $1/7 = 0.14 < 0.5$

- Key pairs = [light blue, toyota, 1500 to 2000, >2005]
  Attributes = [silver, toyota, 1800,2003]
  M_{Key} = 2
  T_{Key} = 6
  $S_{Jaccard}$ = $2/6 =0.33 < 0.5$

- Key pairs = [light blue, toyota, 1500 to 2000, >2005]
  Attributes = [light blue, toyota, 1300,2005]
  M_{Key} = 3
  T_{Key} = 5
  $S_{Jaccard}$ = $3/5 = 0.6 \geq 0.5$

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

V. EXPERIMENTAL RESULTS

These experiments are evaluated on 10000 extracted keys. 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:

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

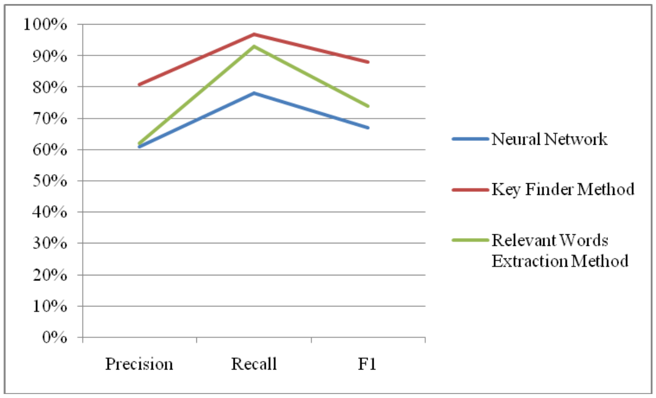

Fig. 2 Experimental Results

Figure 2 presents the experimental results between automatic training approach, our previous relevant words extraction method [22] and proposed key finder method. Current proposed method can retrieve more related words because of modification rules. As the results, we can produce higher precision, recall and F1 measure. Moreover, that can skip stop word removal step. So the system is more accurate and save the processing time than previous one.

VI. Conclusion and Future Works

This system tried to use knowledge engineering approach in information extraction rather than automatic training approach. Context free grammar is suitable to generate rules. These rules are simple and easy to extract the car information. Proposed key finder method is also very easy to understand. Every relevant word can be recognized by key finder. Key finder uses the Rule Matching technique to complete its extraction process. Accuracy and completeness is comparable with the complex systems with simple technique. When the key finder found the desired key pairs, these key pairs are supporter to generate recommended data according to user's request. 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 is more accurate by combining this information extraction system.

We can extend the system with the semantic framework. We can apply this technique on Mobile Computing.

References

[1] Ashwini Madane, "Identifying Keywords and Key Phrases", IJSCE, July 2012.
[2] Yize Li, Jiazhong Nie, Yi Zhang, and Bingqing Wang, "Contextual Recommendation based on Text Mining", 2012.
[3] Gunnar Schröder, Maik Thiele, and Wolfgang Lehner, "Setting Goals and Choosing Metrics for Recommender System Evaluations", 2012.
[4] Huda Yasin, Mohsin Mohammad Yasin, and Farah Mohammad Yasin, "Automated Multiple Related Documents Summarization via Jaccard's Coefficient", International Journal of Computer Applications, January 2011.
[5] Rares Vernica, Michael J.Carey, and Chan Li, "Efficient Parallel Set-Similarity Joins Using MapReduce", June 2010.
[6] Raymond J.Mooney, and Razvan Bunescu, "Mining Knowledge from Text Using Information Extraction", 2009.
[7] SAS Institute, *Getting Started with SAS® Text Miner 4.1*, ISBN 978-1-59994-999-4, 2009.
[8] Munyaradzi Chiwara, Mahmoud Al-Ayyoub, Mohammad Sajjad, Hossain, and Rajan Gupta, "CSE 634 – Data Mining: Text Mining", 2009 .
[9] Hany Mahgoub, Dietmar Rösner, Nabil Ismail, and Fawzy Torkey, "A Text Mining Technique Using Association Rules Extraction",International Journal of Information and Mathematic Sciences ,2008 .
[10] Zakaria Elberrichi, Abdelattif Rahmoun, and Mohamed Amine Bentaalah, "Using WordNet for Text Categorization", The International Arab Journal of Information Technology, January 2008.
[11] Francesco Ricci, "Content-Based Filtering and Hybrid Systems", 2005.
[12] Sundar Varadarajan, Kas Kasravi, and Ronen Feldman,"Text-Mining: Application Development Challenges", 2004.
[13] Haralampos Karanikas, Christos Tjortjis, and Babis Theodoulidis, "An Approach to Text Mining using Information Extraction", 2001.
[14] Damian Fijalkowski, and Radolsaw Zatoka, "An 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.
[15] P.M. LEWIS II, D.J. ROSENKRANTZ, and R.E. STEARNS, *Complier Design Theory*, ISBN 0-201-14455-7, November 1978.
[16] Appelt, D.E., Israel, D.J., "Introduction of Information Extraction Technology", A Tutorial Prepared for IJCAI-99, Menlo Park, Canada, 1999.
[17] Katharina Kaiser and Silvia Miksch, "Information Extraction : A Survey", Vienna University of Technology, May 2005.
[18] Benjamin Rosendfeld, Ronen Feldman and Moshe Fresko, "TEG – a hybrid approach to information extraction, DOI , 2007.
[19] Mahmudul Sheikh and Sumali Conlon, "A Rule-based system to extract financial information", Journal of Computer Information Systems, summer 2012.
[20] Rajasekar Krishnamurthy, Yunyao Li, Sriram Raghavan, Frederick Reiss, Shivakumar Vaithyanathan, and Huaiyu Zhu, "System T: A System for Declarative Information Extraction", SIGMOD Record, Vol.37,No.4, December 2008.
[21] Ignazio Gallo and Elisabetta Binghi, "Information Extraction and Classification from Free Text Using a Neural Approach", 2009.
[22] Naw Naw, Ei Ei Hlaing, "Relevant Words Extraction Method for Recommendation System", International Journal of Emerging Technology and Advanced Engineering, Vol 3 Issue1, January 2013.