An Efficient Discovery of Comparable Entities from the Perspective of Comparative Question using Data Mining

Ms. Megha K. F
Dept of CS&E
KVGCE, SULLIA, DK-574327
VTU, Belgaum

Mr. Ujwal U J
Prof/HOD, Dept of CS&E
KVGCE, SULLIA, DK-574327
VTU, Belgaum

Abstract**-The study proposes a novel way to automatically mine comparable entities from comparative questions users post online. Comparison made between one thing and another thing which is a called as a human decision-making process. It is not an easy way to know what to compare, what the alternatives are and what not to compare with. To address these difficulties, a weakly supervised Bootstrapping method for comparative question identification and Apriori TID algorithm for comparable entity extraction from the questions. The study is similar to J&L's method based on supervised mining method, which achieved high precision but suffered from low recall. Achieving high recall is crucial the proposed study outperforms the existing method by achieving high precision and high recall. The study also reduces the burden of making manual comparison by the users. This method is mainly used in the field of e-commerce such as online shopping, search engines, online ticket booking and also providing useful information to companies which wants to identify their competitors.**

Keywords-Apriori algorithm, Bootstrapping algorithm, Inductive Extraction Pattern, Weakly supervised mining.

I. INTRODUCTION

A good Decision making is an essential step in day by day activity. For example, if someone is interested in certain products such as mobiles and cameras, one would want to know what the alternatives are and compare different products before making a purchase. The comparison activity is very common in our daily life, but requires high knowledge and skills. Magazines and online Medias also strive in providing editorial comparison content and surveys to satisfy this need. Various search engines are used for comparing the things, For example, Google, Yahoo and Amazon. Some Also go for web pages to collect relevant information, like Commerce search or Product recommendation system [1][2].

A comparison activity involves, searching for relevant web pages for relevant products containing useful information, find competing products and identify advantages and disadvantages. which may result in false prediction. Hence to avoid this false prediction, comparable entity identification and extraction process is used in this study. Here, the main aim is to focus on finding a set of comparable entities given by the users input. To mine comparators or extract entities from comparative questions, first the question must be comparative. A comparative question has to be a question with the intent to compare at least two entities. A question containing at least two entities is not a comparative question if it does not have comparison intention. A question is very likely to be a comparative question if it contains at least two entities. for example,

- "What are the different features of iphones i5 and smart phone ace advance"?
- "Which is better apple or Samsung"?
- "What is ipod and ipad"?

Here Apple and Samsung, ipod and ipad are the comparable entities taken. Which are the target for comparison, ipod and ipad is called as the comparators. Based on this a weakly supervised bootstrapping method for comparison question identification and Apriori method for entity extraction are developed.

II.RELATED WORK

The study on comparator mining is related to the research on entity and relation extraction in information extraction (Cardie, 1997; Califf and Mooney, 1999; Soderland, 1999; Radev et al., 2002; Carreras et al., 2003). Specifically, the most relevant work is by Jindal and Liu (2006a and 2006b) on mining comparative sentences and relations. Their methods applied class sequential rules (CSR) and label sequential rules (LSR).The same techniques can be applied to comparative question identification and comparator mining from questions. However, their methods typically achieved high precision but suffer from low recall. However ensuring high recall is crucial in this study for intended application scenario where users can issue arbitrary queries. To address this problem, a weakly-supervised bootstrapping pattern learning method by effectively leveraging unlabeled questions is developed. Bootstrapping method is showed to be very effective in previous information extraction research (Riloff, 1996; Riloff and Jones, 1999; Ravichandran and Hovy, 2002; Mooney and Bunescu, 2005; Kozareva et al., 2008). The propsed study is similar to them in terms of methodology using bootstrapping technique to extract entities with a specific relation. The task is different from theirs which requires not only extracting entities that is comparator extraction but also ensuring that the entities are extracted from comparative questions that is comparative question identification, which is generally not required in IE task[1][3].

J&L used CSR and LSR rules, CSR is a classification rule and LSR is a labelling rule. It maps a sequence pattern S $(s1s2 \ldots)$ to a class C. In our problem, C is either comparative or non-comparative. Given a collection of sequences with class information, every CSR is associated to two parameters: support and confidence. Support is the proportion of sequences in the collection containing S as a subsequence. Confidence is the proportion of sequences labelled as C in the sequences containing the S. These parameters are important to evaluate whether a CSR is reliable or not. LSR maps an input sequence pattern $(s1s2 \ldots si \ldots sn)$ to a labelled sequence $S'(s1s2 \ldots li \ldots sn)$ by replacing one of the token $(si\ )$ in the input sequence with a designated label $(li\ )$. This token is referred as the anchor. The anchor in the input sequence could be extracted if its corresponding label in the labelled sequence is what we want (in our case, a comparator).LSRs is also mined from an annotated corpus, therefore each LSR also have two parameters, support and confidence. They are similarly defined as in CSR [2][3].

III. EXISTING SYSTEM

A Naives Bayes classifier is based on a supervised mining method used with CSR and LSR rules for identifying comparative sentences. This method was effective but suffered from drawbacks:

- A large annotated training corpus is necessary for achieving high recall that is many labels are created and compared with each other [4].
- There was no assurance of completeness of the keyword list, many keyword created manually. The comparative sentence heavily depended on these inductive keywords.
- J&L's method suffered from more errors resulting in high precision but low recall because of POS tags and keywords [5].

Based on these key points showed above so many online shopping based applications have been developed, the major drawback of these applications are lack of customer satisfaction, the comparison of product is manual hence no useful information obtained, resulting in false prediction.

IV. PROPOSED SYSTEM

In proposed approach two techniques that is Bootstrapping and Apriori algorithm are used. First the bootstrapping algorithm is used to identify comparative question posted online, and the comparative questions are stored in a database. The database consists of both comparative and non comparative questions. Using apriori algorithm, comparator patterns or comparators (entities) from the comparative questions are extracted.

A. Weakly Supervised Method for Comparator Mining

The proposed techniques is a pattern based approach similar to J&Ls method, but it is different in many aspects: Instead of using separate CSRs and LSRs, The method uses sequential patterns which can be used to identify comparative question and extract comparators simultaneously.

1. <#start which city is better, $C or $C ? #end>
2. <, $C or $C ? #end>
3. <#start $C/NN or $C/NN ? #end>
4. <which NN is better, $C or $C ?>
5. <which city is JJR, $C or $C ?>
6. <which NN is JJR, $C or $C ?>

A sequential pattern is defined as a sequence S(s1s2 … si … sn) where si can be a word, a POS tag, or a symbol denoting a comparator ($C), or the beginning (#start) or the end of a question (#end). A sequential pattern is called an indicative extraction pattern (IEP) if it is used to identify comparative questions and extract comparators in them with high reliability [1]. The IEPs acts as a template for automatic extraction of comparable question.

Once a question matches an IEP, it is classified as a comparative question and the token sequences corresponding to the comparator slots in the IEP are extracted as comparators. When a question can match multiple IEPs, the longest IEP is used. Therefore, instead of manually creating a list of indicative keywords, a set of IEPs are created. IEPs are acquired automatically using a bootstrapping procedure with minimum supervision by taking advantage of a large unlabeled question collection [6][7].

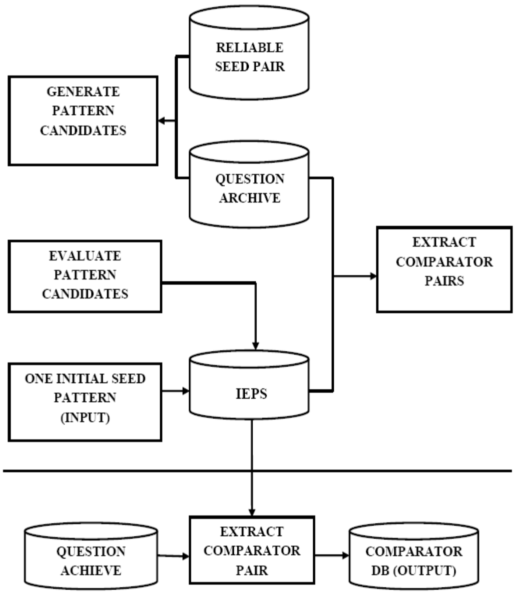

Figure.1: Bootstrapping algorithm

B. Mining Indicative Extraction Patterns and Bootstrapping Algorithm

Based on these two assumptions, bootstrapping algorithm as shown in Figure 1 is designed.

- If a sequential pattern can be used to extract many reliable comparator pairs, it is very likely to be an IEP.
- If a comparator pair can be extracted by an IEP, the pair is reliable.

The bootstrapping process starts with a single IEP. From it, extract a set of initial seed comparator pairs. For each comparator pair, all questions containing the pair are retrieved from a question collection and hence called as comparative questions. Patterns evaluated as reliable ones are IEPs and are added into an IEP repository. Then, new comparator pairs are extracted from the question collection using the latest IEPs. The new comparators are added to a reliable comparator repository and used as new seeds for pattern learning in the further iteration. All questions from which reliable comparators are extracted are removed from the collection to allow finding new patterns efficiently in further iterations. The process iterates until no more new patterns can be found from the question collection. The working of bootstrapping algorithm is shown below.

Step 1: Scan the Storage servers.

Step 2: Extract questions from storage servers.

Step 3: Generate set of IEPs [an indicative extraction pattern (IEP) if it can be used to identify comparative questions and extract comparators in them]

Step 4: Classify Comparative and Non Comparative Questions.

Step 5: Determine efficient comparative questions.

Step 6: Extract comparable entities from comparative questions.

C. Apriori TID Algorithm

The result obtained from the bootstrapping method is taken, that is the bootstrapping method scan the database and identifies the comparable question, and from these comparable questions various comparator or entities are extracted. This result is used by the apriori

TID Algorithm. It is an algorithm for frequent item set mining and association rule learning for transactional databases. It proceeds by identifying the frequent individual items in the database and extending them to larger and larger item sets as long as those item sets appear sufficiently often in the database [8]. The frequent item sets of Apriori can be used to determine association rules[8]. The database is scanned and the comparable entities are determined. These entities are used to calculate support (S)and confidence (C)of the item set(L1), This is used to generate frequent item set. Join Lk-1 generate the set of candidate k item set. The steps are repeated until frequent item is NULL For each non empty subset determine the confidence. If confidence is greater than or equal to this specified confidence. Let us take P1,P2,P3 items, From these item set (P1,P2) is taken If The confidence is high, It is easy to predicted that the user who buy an item P1 may also buys P2.Hence a comparable dataset is obtained as a result. The main steps to generate the frequent item sets are shown below.

Step 1: Scan the comparison data set and determine the support(s) of each item.

Step 2: Generate L1 (Frequent one item set).

Step 3: Use Lk-1, join Lk-1 to generate the set of candidate k item set.

Step 4: Scan the candidate k item set and determine the support of each item.

Step 5: Add to frequent item set until C=ϕ.

Step 6: For each item in the frequent item set generate all non empty subset.

Step 7: For each non empty subset determine the confidence. If confidence is greater than or equal to this specified confidence .Then add to Strong Association Rule.

Step 8: Determine the relationship between comparable entities.

V. PERFORMANCE DISCUSSION

The performance of proposed system is more in terms of Precision and recall. Since this method achieves high precision and high recall compared to the previous work of J&L. Hence automatically mined entities are obtained from the comparable question. Rather than manual comparison of products The customers get satisfaction with the automatic comparison, hence they are provided with best result

VI. CONCLUSION

A weakly supervised Bootstrapping method for identify comparative questions and Apriori algorithm extract comparator pairs is presented. This achieves high precision and high recall. The comparator mining results can be used in the field of e-commerce such as a commerce search or product recommendation system. For example, automatic suggestion of comparable entities can assist users in their comparison activities before making their purchase decisions. Also, our results can provide useful information to companies which want to identify their competitors.

REFERENCES

[1]. Comparable Entity Mining from Comparative Questions Li, Shasha, et al. "Comparable entity mining from comparative questions."Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2010.

[2]. Mining Opinions in Comparative Sentences Ganapathibhotla, Murthy, and Bing Liu. "Mining opinions in comparative sentences." Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1. Association for Computational Linguistics, 2008.

[3]. Natural Language Programming Using Class Sequential Rules Carlos, Cohan Sujay. "Natural Language Programming Using Class Sequential Rules." IJCNLP. 2011.

[4]. Annotated corpus Palmer, Martha, Daniel Gildea, and Paul Kingsbury. "The proposition bank: An annotated corpus of semantic roles." Computational Linguistics 31.1 (2005): 71-106.

[5]. Khan, Rafiqul Zaman, and Haider Allamy. "Training Algorithms for Supervised Machine Learning: Comparative Study." International Journal of Management & Information Technology 4.3 (2013): 354-360.

[6]. Bootstrapping algorithm Lin, Wen-Pin, Matthew Snover, and Heng Ji. "Unsupervised language-independent name translation mining from Wikipedia infoboxes." Proceedings of the First Workshop on Unsupervised Learning in NLP. Association for Computational Linguistics, 2011.

[7]. Zhang, Zhu. "Weakly-supervised relation classification for information extraction." Proceedings of the thirteenth ACM international conference on Information and knowledge management. ACM, 2004.

[8]. Rakesh Agrawal and Ramakrishnan Srikant Fast algorithms for mining association rules in large databases. Proceedings of the 20th International Conference on Very Large Data Bases, VLDB, pages 487-499, Santiago, Chile, September 1994.