A Novel Method to Extract Comparison of Products Using Comparative Questions

Rashmi A G
M.Tech Student, Dept of CSE
GSSS Institute of Engineering and Technology for Women
Mysuru, India
E-mail Id- rashmiag90@gmail.com

Mrs.Swarnalatha K
Associate Prof, Dept of ISE
GSSS Institute of Engineering and Technology for Women
Mysuru, India
E-mail Id- swarnapradyu@yahoo.co.in

Abstrac*t*—Comparing one thing with another is a typical part of human decision making process, especially during an online purchase scheme. To assist decision making it is useful to compare entities that share common utility but have distinguishing peripheral features. Without comparing it is not fair to purchase a product, since it would not give an ideal performance. To get rid of this difficulty, my paper presents an ideal way for automatically mine comparable entities from comparative questions that users posted online. It gives an opportunity to improve the search experience by automatically offering comparisons to user. A weekly supervised bootstrapping algorithm is employed here for comparative question identification and comparable entity extraction by collecting a large online question archive. This is done by detecting whether a given question is comparative or not. A sequential pattern is generated and is called an indicative extraction pattern (IEP) if it can be used to identify comparative questions and extract comparator pairs with high reliability. The results will be very useful in helping users' exploration of alternative choices by suggesting comparable entities based on other users' prior requests.

Abstract—**This electronic document is a "live" template. The various components of your paper [title, text, heads, etc.] are already defined on the style sheet, as illustrated by the portions given in this document. (Abstract)**

Keywords— Comparative question, Bootstrap algorithm, Comparable entity mining, Indicative Extraction Pattern, Sequential pattern mining.

I. INTRODUCTION

In decision-making process, comparing alternative options is one of the necessary steps that we carry out on a daily basis. The essential step while making decision is comparing alternative options because without comparing number of options available, it is not possible to make best decision. However this requires high knowledge expertise. For e.g., during online shopping of a laptop one must have detailed knowledge of its specifications like Processor, Memory, Storage, Graphics, Display, etc. In such case, it becomes difficult for a person with insufficient knowledge to make a good decision on which laptop to buy and also comparing the alternative options for the same. For example, if someone is interested in certain products such as mobiles, he or she would wants' to know what the alternatives are and compare different mobiles types before making a purchase. In general, it is difficult to decide if two entities are comparable or not since people do compare apples and oranges for various reasons. For example, "*Ford*" and "*BMW*" might be comparable as "car manufacturers" or as "market segments that their products are targeting", but we rarely see people comparing "*Ford Focus*" (car model) and "*BMW 328i*". Things also get more complicated when an entity has several functionalities.

The comparative questions and comparators can be thus defined as:

Comparative question: A question that in- tends to compare two or more entities and it has to mention these entities explicitly in the question.

Comparator: An entity which is a target of comparison in a comparative question.

According to these definitions, Q2 below is not comparative questions while Q1 and Q3 are comparative questions. "*Canon*" and "*Nikon*" are comparators of Q1. "Nokia" and "Samsung" are comparators of Q3.

Q1. "Which camera is better Canon or Nikon?"

Q2. "Which one is better?"

Q3. "What's the difference between nokia and Samsung?"

To mine comparators from comparative questions, we first have to detect whether a question is comparative or not. A question is said to be comparative question if it compare at least two entities. Please note that a question containing at least two entities is not a comparative question if it does not have comparison intention. However, we observe that a question is very likely to be a comparative question if it contains at least two entities. A weakly supervised method is used for this purpose.

Comparisons are one of the convincing ways of evaluation. For example in the business environment whenever a new product comes into market, the product manufacturer wants to know consumer opinions on the product and how the product compares with those of its competitors. Extracting such information can significantly help businesses in their marketing and product benchmarking efforts. Clearly the product comparisons are not only useful for product manufacturers but also to potential customers as it enable customers to make better purchasing decision.

II. EASE OF USE

In terms of discovering related items for an entity, our work is similar to the research on recommender systems, which recommend items to a user. Recommender systems mainly rely on similarities between items and/or their statistical correlations in user log data[3]. For example, Amazon recommends products to its customers based on their own purchase histories, similar customers purchase histories, and similarity between products. However, recommending an item is not equivalent to finding a comparable item. In the case of Amazon, the purpose of recommendation is to entice their customers to add more items to their shopping carts by suggesting similar or related items. While in the case of comparison, we would like to help users explore alternatives, i.e. helping them make a decision among comparable items.

For example, it is reasonable to recommend "*iPod speaker*" or "*iPod batteries*" if a user is interested in "*iPod*", but we would not compare them with "*iPod*". However, items that are comparable with "*iPod*" such as "*iPhone*" or "*PSP*" which were found in comparative questions posted by users are difficult to be predicted simply based on item similarity between them. Although they are all music players, "*iPhone*" is mainly a mobile phone, and "*PSP*" is mainly a portable game device. They are similar but also different therefore beg comparison with each other. It is clear that comparator mining and item recommendation are related but not the same.

A. CSR and LSR

CSR is a classification rule. It maps a sequence pattern $S(s1s2 \ldots sn)$ 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 labeled as *C* in the sequences containing the *S*. These parameters are important to evaluate whether a CSR is reliable or not.

LSR is a labeling rule. It maps an input sequence pattern $(s1s2 \ldots si \ldots sn)$ to a labeled sequence $S'(s1s2 \ldots li \ldots sn)$ by replacing one 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 labeled sequence is what we want (in our case, a comparator). LSRs are also mined from an annotated corpus, therefore each LSR also have two parameters: support and confidence. They are similarly defined as in CSR.

B. Existing System

- It's a manual process of comparing the products & it's based on the users mind set which may not be accurate, which leads to confusions, time consuming, less efficient and lack of customer satisfaction.
- The topic is about, identifying the comparative sentences.
- An important application area of opinion identification is business intelligence as a product manufacturer always wants customer opinion on its product.
- An example opinion sentence is "The sound quality of CD player X is poor"
- An example of comparative sentence "The sound quality of CD player X is not as good as CD player Y"
- These two sentences give different information this paper studies the problem of identifying comparative sentence in the text document.

C. Limitations

- These methods typically can achieve high precision but suffer from low recall.
- Lack of customer satisfaction, time consuming, less efficient.
- Users need to compare the products manually which may lead to confusions and inaccurate results.

D. Information Extraction

Information extraction is the task of automatically extracting structured information from unstructured readable documents. Almost every day people are faced with a situation that must decide upon one thing or the other. To make better decisions probably attempt to compare entities that the customer are interesting in. These days many web search engines are helping people look for their interesting entities. Therefore a comparison mining system, which can automatically provide a summary of comparisons between two entities from a large quantity of web documents, would be very useful in many areas such as marketing. The work is divided into two tasks to effectively build a comparison mining system. First classify the sentences into comparatives and non-comparatives and the second is related to comparative mining.

III. METHODOLOGY

The bootstrapping method is also called self-training, is a form of learning that is designed to use even less training examples, therefore sometimes called weakly-supervised. A great advantage of bootstrap is its simplicity. Weakly supervised method is a pattern-based approach and aims to learn the sequential patterns which can be used to identify comparative question and extract comparators simultaneously. Moreover, it is a suitable way to control and check the durableness of the results. Bootstrapping is a method for assigning the accuracy.

Weakly supervised method is a pattern-based approach similar to J&L method, but it is different in many aspects: Instead of using separate CSRs(Class sequential rule) and LSRs(Label sequential rule), our method aims to learn sequential patterns which can be used to identify comparative question and extract comparators simultaneously. The weakly supervised IEP mining approach is based on two key assumptions: 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.

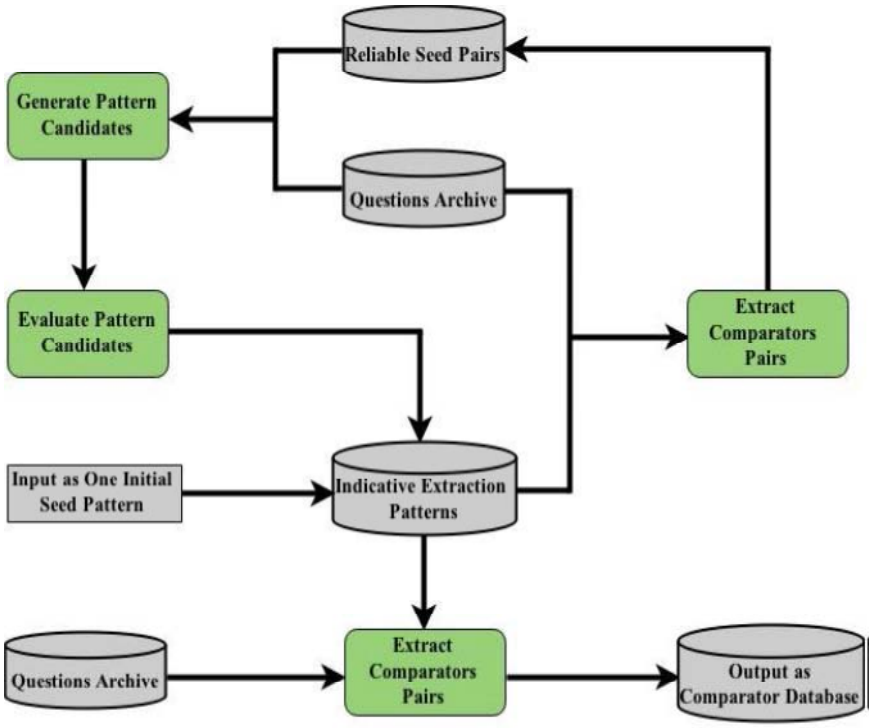


Figure 1: System Architecture

Based on these two assumptions, we design our bootstrapping algorithm as shown in Figure 1. The bootstrapping process starts with a single IEP. From it, we extract a set of initial seed comparator pairs. For each comparator pair, all questions containing the pair are retrieved from a question collection and regarded as comparative questions. From the comparative questions and comparator pairs, all possible sequential patterns are generated and evaluated by measuring their reliability score. 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 next iteration. The overview of bootstrapping algorithm is shown below, where the databases store seed pairs and question archive and from them relevant data is extracted. All questions from which reliable comparators are extracted are removed from the collection to allow finding new patterns

efficiently in later iterations. The process iterates until no more new patterns can be found from the question collection.

A. Association Rule Mining

Association rule mining is one of the most important and well researched techniques of data mining. It aims to extract interesting correlations, frequent patterns, associations or casual structures among sets of items in the transaction databases or other data repositories.

There are two important basic measures for association rules, *support*(*s*) and *confidence*(*c*). Since the database is large and users concern about only those frequently purchased items, usually thresholds of support and confidence are pre-defined by users to drop those rules that are not so interesting or useful. The two thresholds are called minimal support and minimal confidence respectively, additional constraints of interesting rules also can be specified by the users. The two basic parameters of Association Rule Mining (ARM) are: support and confidence.

Support(s) of an association rule is defined as the how many transaction that contain X∪Y to the total number of transaction in the database. The count for each item is increased by one every time the item is encountered in different transaction T in database D during the scanning process.

Support(*s*) is calculated by the following formula:

$$\text{Support(s)} = \frac{Support\ count\ of\ XY}{Total\ number\ of\ transaction\ in\ D}$$

From the definition we can see, support of an item is a statistical significance of an association rule. Suppose the support of an item is 0.1%, it means only 0.1 percent of the transaction contain purchasing of this item. The retailer will not pay much attention to such kind of items that are not bought so frequently, obviously a high support is desired for more interesting association rules. Before the mining process, users can specify the minimum support as a threshold, which means they are only interested in certain association rules that are generated from those itemsets whose supports exceed that threshold. However, sometimes even the itemsets are not so frequent as defined by the threshold, the association rules generated from them are still important

Confidence of an association rule is defined as how many transactions that contain X*U*Y to the total number of transaction that contain X, where if the percentage exceeds the threshold of confidence an interesting association rule X➔Y can be generated.

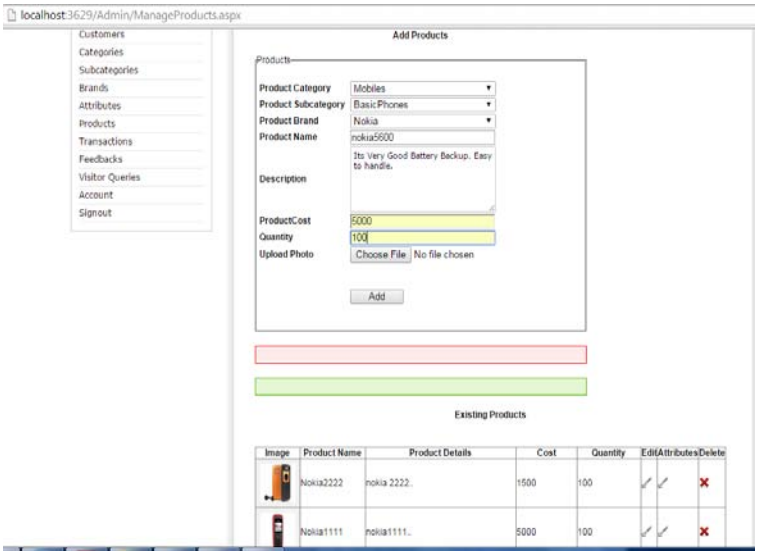
$$Confidence(X \Rightarrow Y) = \frac{Support(XY)}{Support(X)}$$

Confidence is a measure of strength of the association rules, suppose the confidence of the association rule X➔Y is 80%, it means that 80% of the transactions that contain X also contain Y together, similarly to ensure the interestingness of the rules specified minimum confidence is also pre-defined by users.

Define abbreviations and acronyms the first time they are used in the text, even after they have been defined in the abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc, and rms do not have to be defined. Do not use abbreviations in the title or heads unless they are unavoidable.

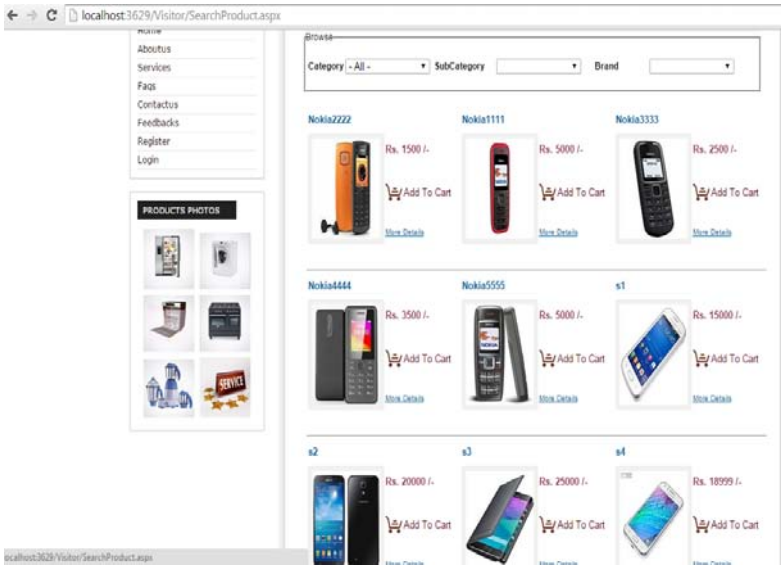
IV. RESULT AND ANALYSIS

In this paper, I present a novel weakly supervised method to identify comparative questions and extract comparator pairs simultaneously. And also I use data mining technique, association rule mining to calculate support and confidence. Using apriori tid algorithm I display the comparable products for the selected product based on the subcategory of that selected product. In future for using all this I develop online application like flipcart. I rely on the key insight that a good comparative question identification pattern should extract good comparators, and a good comparator pair should occur in good comparative questions to bootstrap the extraction and identification process.



Snapshot1-Add products

In this administrator/owner of the application need to enter the product information like product category, product subcategory, product brand, product name, description, product cost, quantity, and finally photo of the product. All this information is useful to the customer while doing online shopping. Administrator can delete, update, add the products.



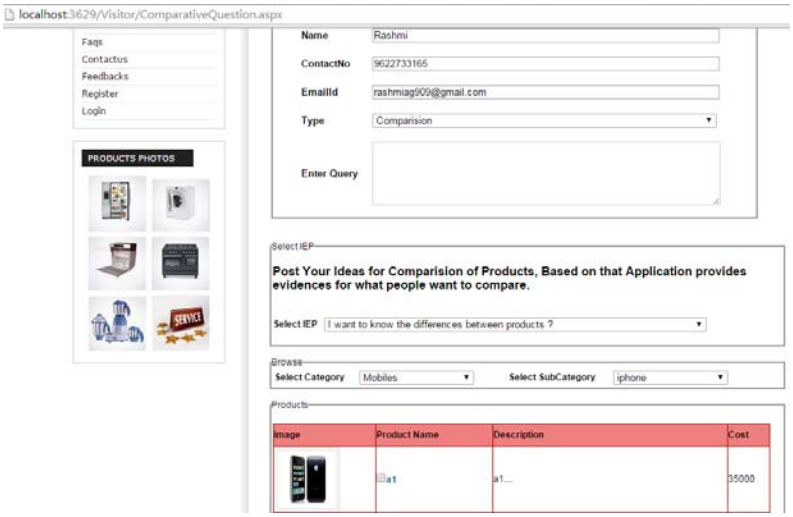
Snapshot2-Ptoduct list

Here customer can get the list of products based on the selected category, subcategory and brand. If the customer need to do the transaction, click on add to cart button to place order. After that he/she can get the message that the item will be delivered within 7 to 10 working days if quantity are available, otherwise it takes some days extra depending on the quantity and items.

In Snapshot 3 post **c**omparative question module user can post the comparative question on online based on those comparative questions user can get the comparable products for the selected product.

Here customer can post the questions regarding the application services, registration, online payment, login, shopping, comparison etc; from these questions we should identify the comparative questions for which we make use of data mining technique.

Plenty of comparative questions are posted online, which provide evidences for what people want to compare. Mining comparable entities from comparative questions that users posted online. Comparable entities extraction based on the selected product by a user in online shopping application. So in this way we can satisfy the customers in an efficient way. Here we make use of data mining technique for comparable products extraction.



Snapshot3-Post Comparative Questions

V. CONCLUSION

- The goal of this work is determining comparators from relative questions.
- Novel way to automatically determine comparable entities from relative questions that users posted online.
- To reduce the complications when an entity has several functionalities.
- Predicting Comparable Entities from plenty of relative questions posted online, which provide evidences for what people want to compare.
- Helping user's exploration of alternative choice by suggesting comparable entities based on other user's prior requests.
- Comparable entities (products) extraction based on the selected product by a user in online shopping application.
- Its hepful for decision making.
- It will provide useful information to companies which want to identify their competitors.

REFERENCES

[1] S. Li, C.-Y. Lin, Y.-I. Song, and Z. Li, "Comparable Entity Mining from Comparative Questions," IEEE Transactions On Knowledge And Data Engineering, vol. 25, no. 7, 2013, 1498-1509.
[2] S.Li, C.-Y. Lin, Y.-I. Song, and Z. Li,"Comparable Entity Mining from Comparative Questions," Proc. 48th Ann. Meeting of the Assoc. for Computational Linguistics (ACL '10), 2010.
[3] Nitin Jindal and Bing Liu. 2010. Identifying comparative sentences in text documents. In Proceedings of SIGIR '06, pages 244–251.
[4] Nitin Jindal and Bing Liu. 2006b. Mining comparative sentences and relations. In Proceedings of AAAI '06.
[5] Raymond J. Mooney and Razvan Bunescu. 2005. Mining knowledge from text using information extraction. ACM SIGKDD Exploration Newsletter, 7(1):3–10.
[6] Ellen Riloff and Rosie Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In Proceedings of AAAI '99 /IAAI '99, pages 474–479.
[7] E. Riloff, "Automatically Generating Extraction Patterns from Untagged Text," Proc. 13th Nat'l Conf. Artificial Intelligence,pp. 1044-1049, 1996.
[8] S. Soderland, "Learning Information Extraction Rules for Semi-Structured and Free Text," Machine Learning, vol. 34, nos. 1-3, pp. 233-272, 1999.
[9] Dragomir Radev, Weiguo Fan, Hong Qi, Harris Wu, Amardeep Grewal,"Probabilistic question answering on the web", Journal of the American Society for Information Science and Technology, pp. 408–419, 2002.
[10] D. Ravichandran and E. Hovy, "Learning Surface Text Patterns for a Question Answering System," Proc. 40th Ann. Meeting on Assoc.for Computational Linguistics (ACL '02), pp. 41-47, 2002.
[11] C. Cardie, "Empirical Methods in Information Extraction,"Artificial Intelligence Magazine, vol. 18, pp. 65-79, 1997.
[12] E. Riloff and R. Jones, "Learning Dictionaries for Information Extraction by Multi-Level Bootstrapping," Proc. 16th Nat'l Conf. Artificial Intelligence and the 11th Innovative Applications of Artificial Intelligence Conf. (AAAI '99/IAAI '99), pp. 474-479, 1999.
[13] Dan Gusfield,"Algorithms on strings, trees, and sequences: Computer science and computational biology", Cambridge University Press, New York, NY, USA, 1997.
[14] Greg Linden, Brent Smith, Jeremy York,"Amazon.com Recommendations: Item-to-Item Collaborative Filtering", IEEE Internet Computing, pp. 76-80, 2003.