Equivalent Entity Mining Commencing On Relative Uncertainty
Sangeetha. K1, D.T.V.Dharmjee Rao2
#1Student of M.Tech (CSE) and #2 Professor in Department of Computer Science Engineering, AITAM, Tekkali, Srikakulam

Abstract. Contrast entities are an significant part of decision making progression. To assist judgment making it is useful to evaluate entities that share widespread utility but have distinguishing peripheral characteristics. One possible slant is comparable entity mining from comparative uncertainty. The method used is weakly supervised bootstrapping tactic which identifies the relative question and take out the comparable entity. This is done by distinguish whether a particular question is comparative or not. A sequential prototype is generated and is called an indicative extraction pattern (IEP) if it can be worn to identify comparative questions and dig up comparator pairs with soaring reliability. This technique achieves the F1-measure of 82.5 percent in comparative question credentials and 83.3 percent in comparable entity mining. Inwards proposed system Clique grow examination is used in which the comparable associations are extended and it is used to control assorted query logs from users. Ranking system is used to rank the comparable entities for user input ingress and the results show substantially relevance to user’s similarity intent.

Keywords: High Utility Mining, Pattern Growth, frequent item sets.

I. Introduction:
Comparing alternative options is one essential step in decision-making that we carry out every day. For example, if someone is interested in certain products such as digital cameras, he or she would need to know what the marginals are and compare dissimilar cameras before making a purchase. This mode of comparison activity is very common in our daily life but requires high knowledge aptitude. Magazines such as Consumer Reports and PC Magazine and online media such as CNet.com strive in providing editorial comparison content and surveys to satisfy this need. In the World Wide Web era, a comparison activity typically involves: search for relevant web pages suppress information about the targeted products, find competing products, identify pros, read reviews, and cons. In this paper, we converge on finding a set of comparable entities specified a user’s input entity. For example, give a entity, Nokia N95 (a cellphone), we have to find comparable entities known as Nokia N82, iPhone and so on. usually, it is difficult to decide if two entities are comparable or not since people do compare apples and oranges for numerous reasons. For example, “Ford” and “BMW” might be comparable as “car manufacturers” or as “market segments that their products are targeting”, but we infrequently see people comparing “Ford Focus” (car model) and “BMW 328i”. Things also get more complicated when an entity has several functionalities. For example, one might compare “iPhone” and “PSP” as “portable game player” while compare “iPhone” and “Nokia N95” as “mobile phone”. Fortunately, plenty of comparative questions are posted online, which provide evidences for what people want to compare, e.g. “Which to buy iPhone or iPod?”. We call “iPhone” and “iPod” in this illustration as comparators.

II. Related Work
A. Overview
In terms of discovering related items for an entity, their 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 [2]. For example, Amazon recommends products to its customers based on their own previous purchase; similar customers’ previous purchase, and similarity between products. 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. In the case of comparison, they 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 they are not to 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. Their comparator mining is related to then research
on entity and relation extraction in information extraction [1], [3], [4].

B. Jindal and Liu (J&L) [5], [6]
Major contribution of Jindal and Liu on mining comparative sentences and relations, their methods used in the class sequential rules (CSR) and label sequential rules (LSR). CSR maps a sequence pattern S(s1s2...sn) to class C. Class C is either comparative or non-comparative, and LSR maps an input sequence pattern S'(s1s2...sn) to a labeled sequence S''(s1s2...sn) by replacing one token si in the input sequence with a designated label (li). This token is referred as the anchor.

J&L work on Supervised Comparative Mining Method and they treated comparative sentence identification as a classification problem and comparative relation extraction as an information extraction problem. They first manually created a set of 83 keywords such as beat, exceed, and outperform that are likely indicators of comparative sentences. These keywords were then used as pivots to create part-of-speech (POS) sequence data.

III. Architecture:
In figure i) first the input query is given by the user. It analyze whether the given question is comparative question or not. If the given question is comparative question then the comparator pairs are extracted. Bootstrapping approach is used to identify the comparative question and extract comparator simultaneously. The features of the comparator pairs are extracted. IEP is indicative extraction pattern, once a question matches an IEP it is classified as comparative question and the token sequence corresponding to the comparator slots in the IEP are extracted as comparators. From the comparative question and the comparator pair all possible sequential patterns are generated and evaluated by measuring the reliability score. Clique grow analysis is used, it defines the multiple patterns. If a comparator is compared to many other important comparators which can also be compared to the input entity, it would be considered as a valuable comparator in ranking. The best comparator are detected and decisions are made on analyzing the features of an entity.

IV. Experiments
Experiment Setup
Source Data
All experiments were conducted on about 60M questions mined from Yahoo! Answers’ question title field. The reason that we used only a title field is that they clearly express a main intention of an asker with a form of simple questions in general.

Evaluation Data
Two separate data sets were created for evaluation. First, we collected 5,200 questions by sampling 200 questions from each Yahoo! Answers category. Two annotators were asked to label each question manually as comparative, non-comparative, or unknown. Among them, 139 (2.67%) questions were classified as comparative, 4,934 (94.88%) as non-comparative, and 127 (2.44%) as unknown questions which are difficult to assess. We call this set SET-A. Because there are only 139 comparative questions in SET-A, we created another set which contains more comparative questions. We manually constructed a keyword set consisting of 53 words such as “or” and “prefer,” which are good indicators of comparative questions. In SET-A, 97.4% of comparative questions contains one or more keywords from the keyword set. We then randomly selected another 100 questions from each Yahoo! Answers category with one extra condition that all questions have to contain at least one keyword. These questions were labelled in the same way as SET-A except that their comparators were also annotated. This second set of questions is referred as SET-B. It contains 853 comparative questions and 1,747 noncomparative questions. For comparative question identification experiments, we used all labelled questions in SET-A and SET-B. For comparator extraction experiments, we used only SET-B. All the remaining unlabeled questions (called as SET-R) were used for training our weakly supervised method. As a baseline method, we carefully implemented J&L’s method. Specifically, CSRs for comparative question identification were learned from the labelled questions, and then a statistical classifier was built by using CSR rules as features.

We examined both SVM and Naïve Bayes (NB) models as reported in their experiments. For the comparator extraction, LSRs were learned from SET-B and applied for comparator extraction. To start the bootstrapping procedure, we applied the IEP “<#start nn/Sc vs/cc nn/Sc ?> / #end>” to all the questions in SET-R and gathered 12,194 comparator pairs as the initial seeds. For our weakly supervised method, there are 26 top level categories in Yahoo! Answers. are four parameters, i.e. a, b, γ, and λ, need to be determined empirically. We first mined all possible candidate patterns from the suffix tree using the initial seeds.

Fig. System Architecture
From these candidate patterns, we applied them to SET-R and got a new set of 59,410 candidate comparator pairs. Among these new candidate comparator pairs, we randomly selected 100 comparator pairs and manually classified them into reliable or non-reliable comparators. Then we found that maximized precision without hurting recall by investigating frequencies of pairs in the labeled set. By this method, w was set to 3 in our experiments. Similarly, the threshold parameters β and γ for pattern evaluation were set to 10 and 0.8 respectively. For the interpolation parameter λ in Equation (3), we simply set the value to 0.5 by assuming that two reliability scores are equally important.

As evaluation measures for comparative question identification and comparator extraction, we used precision, recall, and F1-measure. All results were obtained from 5-fold cross validation. Note that J&L’s method needs a training data but ours use the unlabeled data (SET-R) with weakly supervised method to find parameter setting. This 5-fold evaluation data is not in the unlabeled data. Both methods were tested on the same test split in the 5-fold cross validation. All evaluation scores are averaged across all 5 folds. For question processing, we used our own statistical POS tagger developed in-house.

### Experiment Results

#### Comparative Question Identification and Comparator Extraction

Table shows our experimental results. In the table, “Identification only” indicates the performances in comparative question identification, “Extraction only” denotes the performances of comparator extraction when only comparative questions are used as input, and “All” indicates the end-to-end performances when question identification results were used in comparator extraction. Note that the results of J&L’s method on our collections are very comparable to what is reported in their paper.

<table>
<thead>
<tr>
<th>Identification only (SET-A+SET-B)</th>
<th>J&amp;L (CSR)</th>
<th>Our Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM</td>
<td>0.601</td>
<td>0.537</td>
</tr>
<tr>
<td>NB</td>
<td>0.847</td>
<td>0.851</td>
</tr>
<tr>
<td>F-score</td>
<td>0.704</td>
<td>0.659</td>
</tr>
</tbody>
</table>

Table: Performance comparison between our method and Findal and Bing’s Method (denoted as J&L). The values with * indicate statistically significant improvements over J&L (CSR) SVM or J&L (LSR) according to t-test at p < 0.01 level.

### V. Proposed System:

In Proposed system Clique Grow analysis is used, it is a graph based approach used to manipulate various query logs from users. It aims to improve extraction pattern application and mine rare extraction pattern. It is used to identify the ambiguous entities. It provide results based on user logs and profession information. This method is used to ensure high precision and high recall value and is used to predict transitivity of known comparable relations. Automatic suggestion of comparable entities can assist users in their comparison activities before making their purchase decisions.

A. Modules Comparative question analysis: A question is a linguistic expression used to make a request for information, or the request made using such an expression. Questions assessing comparative judgments are often phrased as directed comparisons, that is, product1 is to be compared to product2. The comparative questions have keywords such as difference, vs and so on. Admin analyze the questions whether it is comparative or not.

IEP implementation: This defines the sequential patterns and identifies the starting and ending of the sentences. A sequential pattern is called an indicative extraction pattern if it can be used to identify comparative questions and extract comparators with high reliability. Then formally define the reliability score of a pattern. 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.

Pattern evaluation: The bootstrapping algorithm is pattern based approach used to analyze the seed points. It is used to extract the features based on the seed points. Evaluate the patterns with several features. 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 process iterates until no more new patterns can be found from the question collection.

Decision making: Given a question, select the longest one among patterns which can be applied to the question. Provide the decision making based on pattern evaluation. If a comparator is compared to many other important comparators which can be also compared to the input entity, it would be considered as a valuable comparator in ranking.

Performance Evaluation: It evaluate the good comparative question identification pattern and extract the good comparators and a good comparator pair should occur in good comparative questions to bootstrap the extraction and identification process. Then calculate FP rate to check whether the decision making is correct or not.

### VI. Conclusion

In this paper, we present a novel weakly supervised method to identify comparative questions and...
extract comparator pairs simultaneously. We 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. By leveraging large amount of unlabeled data and the bootstrapping process with slight supervision to determine four parameters, we found 328,364 unique comparator pairs and 6,869 extraction patterns without the need of creating a set of comparative question indicator keywords. The experimental results show that our method is effective in both comparative question identification and comparator extraction. Significantly improves recall in both tasks while maintaining high precision. Our examples show that these comparator pairs reflect what users are really interested in comparing. Our comparator mining results can be used for 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. In the future, we would like to improve extraction pattern application and mine rare extraction patterns. How to identify comparator aliases such as „LV” and „Louis Vuitton” and how to separate ambiguous entities such as „Paris vs. London” as location and „Paris vs. Nicole” as celebrity are all interesting research topics. We also plan to develop methods to summarize answers pooled by a given comparator pair.

VII. References:

KONA SANGEETHA is a student of Computer Science Engineering from Aditya Institute of Technology And Management, Tekkali. Presently pursuing M.Tech (CSE) from this college.

D.T.V. Dharmajee Rao is a Professor of Aditya Institute of Technology And Management, Tekkali. He received M.Tech (CSE) from AU University, Visakhapatnam.