Natural Language Processing Project Report

Topic: Comparable entity mining from Comparative Questions

Team Name: Tokenny

Rahul Aravind Mehalingam

Thiagarajan Ramakrishnan

Mary Pratima Yedluri

Problem Description

Human decision making process usually involves comparing one entity with another entity. However, it is not always easy to know what to compare and what are the alternatives. When people want to compare two products, they read reviews from blogs, newspapers, advertisements etc. When people want to compare one item with other item, they would post a question online so that it would be answered by some other people who have information. This is called a comparative question. A key point to be noted here is that even though the person has all information about the product and the product he/she is comparing with, it requires high knowledge to distinguish the two. In addition to this, when people compare one thing with another, they might be interested in alternatives too.

The project aims at providing alternatives to a comparative question posted by people online. This project intends to find a set of comparable entities given a user's entity. This would help the user to consider various other alternatives before making a decision.

Problem Explanation:

Let us consider a sample example to illustrate the idea behind the scope of this project.

A comparative question posted by user in some forum online

Google or Bing? Which is better? Need help?

User Entities: Google, Bing

Alternatives: Yahoo, Baidu

We have implemented the idea behind the research paper "**Comparable Entity Mining from Comparative Questions**" (IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 25, NO. 7, JULY 2013) and proposed heuristics approach using rules and lexical keywords to achieve the system's functionality.

Proposed Solution

Initially, a Bootstrapping procedure is executed. An initial seed pattern is applied to the large question archive and a large number of question patterns and comparator pairs are extracted and stored in the respective archives. Then the questions are accepted from users. The question input is classified as comparative or non-comparative by applying pattern based and heuristics based methods. The question is considered as comparative only if it passes successful from both aforementioned methods. The comparators from the user's comparative question are extracted. The semantic similarity between the comparators are identified using Freebase (Google Api) and if the comparator pairs are similar, then the alternatives are mined from the comparator pair archive using frequency based method.

The NLP system is developed by leveraging the information extraction discipline of NLP and achieves the objective of the project by leveraging the promising collection of syntactic, lexical and semantic features.

The significance of the pattern based methods and heuristic based methods are evaluated and it was found that the heuristics based method achieved significantly a higher F score when compared with the pattern based method.

Important Terms

Indicative Extractor Pattern:

Sequence S (S1, S2... Si)

Si can be a word or a POS (Parts of Speech) tag or a symbol denoting either a comparator ($C), or the beginning (#start) or the end of a question (#end).

Precision:

Precision is the ratio of the number of relevant records retrieved to the total number of irrelevant and relevant records retrieved.

Recall:

Recall is the ratio of the number of relevant records retrieved to the total number of relevant records in the database.

F-Score:

F-Score is the harmonic mean of Precision and Recall.

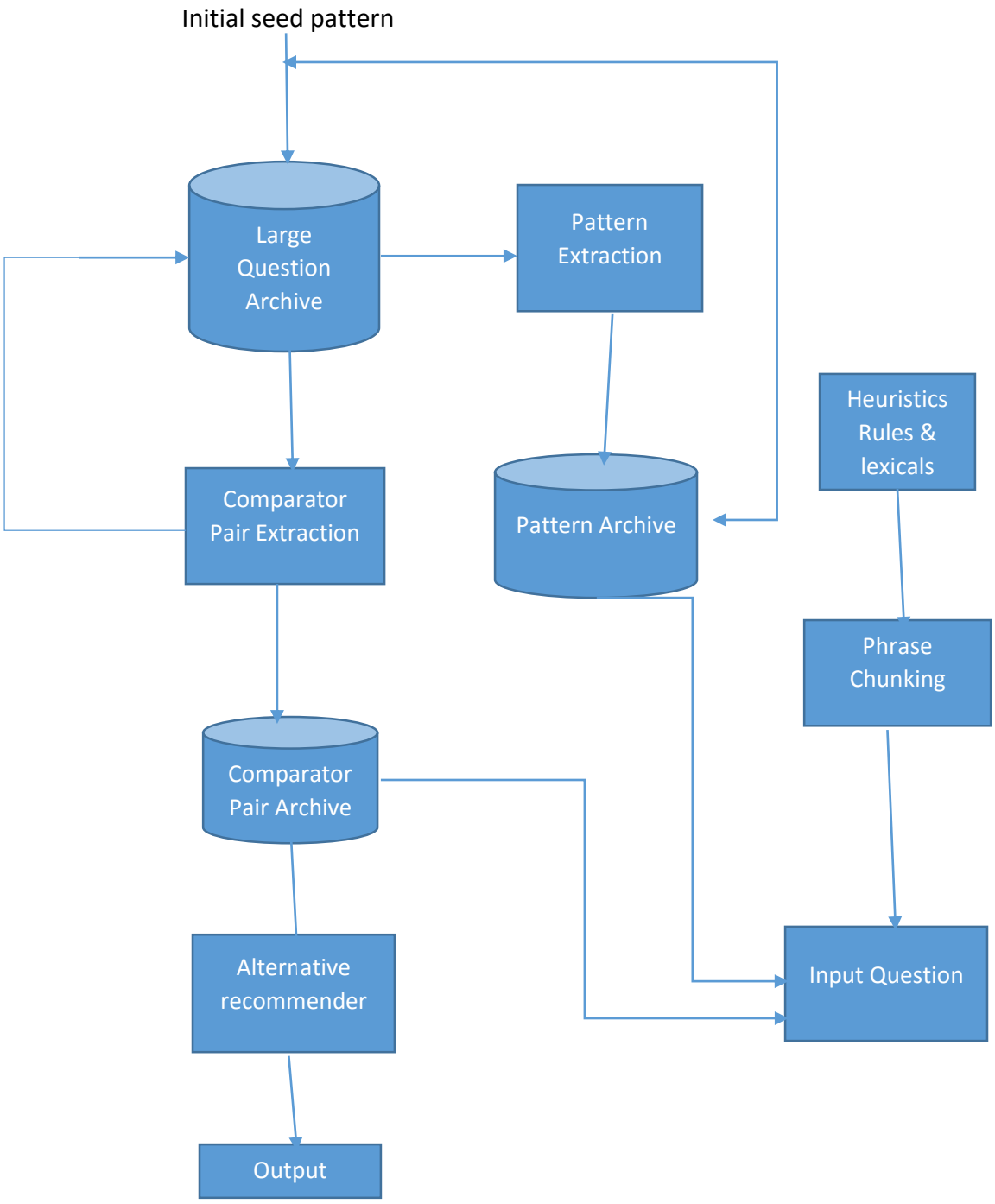
Programming tools:

Programming language: Java

Tools: Stanford POS Tagger, Eclipse IDE

APIs: Freebase JAVA API (Freebase is a community curated database of well-known people, places and things, Stanford CoreNLP Library, Apache Commons library.

Architecture Diagram:



Implementation

Dataset:

A promising collection of datasets were collected and was manually annotated and classified as comparative and non-comparative questions.

Total No. of Questions = 2047

Comparative Questions = 238

Non-Comparative Questions = 1805

Pattern Learning Method:

First, each and every question in the question corpus is pos tagged and converted to a sequential pattern. A sequential pattern is defined as a sequence S (S1, S2, S3 ...Sn) where Si can be a word, a POS tag, or a symbol denoting either a comparator ($C), or the beginning (#start) or the end of a question (#end). A sequential pattern is called an indicative extraction pattern (IEP) if it can be used to identify comparative questions and extract comparators in them with high reliability. After this process, bootstrapping is carried out where an initial seed pattern "$C/NN vs. $C/NN" is applied on a Large Question Archive where "$C/NN" is the comparator tag. The Questions matching the pattern are extracted and are regarded as comparative questions. Then for each comparative question extracted, the comparators in the comparator tag slots are extracted and are regarded as a comparator pair. The comparator pairs extracted are stored in the comparator repository after which the extracted comparative questions are deleted. For each comparator pair extracted, the questions containing the comparator pair are extracted and regarded as comparative questions. The slots corresponding to the comparator pair are replaced with the comparator tag "$C/NN" and are regarded as patterns. The patterns are added to the pattern repository and the comparative questions extracted are deleted. For each pattern in the pattern repository, the process described for the initial seed pattern is executed in loop.

After execution of the bootstrapping process, a large number of patterns and comparator pairs were extracted and stored in the respective archives.

Limitations of Pattern Learning method:

Though the bootstrapping algorithm was stable regardless of the significantly different number of Indicative extraction patterns, it failed to identify certain questions as comparative questions for example,

Is red robins more cheap or is ruby Tuesday more cheap? This was not identified by the bootstrapping method because the pattern $c vs $c was not precise enough to classify such questions. One way to address this, is to relatively have a huge diversity of different forms of question set. The other way to address this would be, If the initial seed pattern applied was $c/NN or/CC $c/NN, this question could have been classified as a comparative question because the question after converting to a sequential pattern would be in the form of Is red/NN robin/NN more cheap or/CC ruby/NN Tuesday/NN more cheap?

Heuristics Method:

Some heuristics rules were applied to classify the questions as comparative or non-comparative and extracting the comparator pairs from the comparative questions.

Consider the following example:

“is/VBZ apple/NN better/JJR than/IN Microsoft/NN ?/. “

The Question after pos tagged contains JJR tag which is a comparative adjective that can be used to identify a comparative question. In order to precisely identify the Noun phrases and adjective/adverb modifiers, phrase chunking rules were applied to the pos tagged output of the questions.

Consider the following example:

Eg., “is Ford GT 100 faster than Ferrari”.

POS Tagger Output:

“Is/VBZ Ford/NNP GT/NN 100/CD faster/JJR than/IN Ferrari/NNP“

Phrase Chuncker Output:

[is, ford gt 100, faster, than, ferrari]

[VBZ, NN, JJR, IN, NN]

Phrase chunking rule set:

NN + NNP -> NNP	NNPS + CD -> NNPS
NN + NNPS -> NNPS	NN + CD + NN -> NN
NNP + NN -> NN	NN + CD + NNS -> NNS
NNPS + NN -> NN	NN + CD + NNP -> NNP
NNS + NNP -> NNP	NN + CD + NNPS -> NNPS
NNS + NNPS -> NNPS	NNS + CD + NN -> NN
NNP + NNS -> NNS	NNS + CD + NNS -> NNS
NNPS + NNS -> NNS	NNS + CD + NNP -> NNP

These were the **syntactic feature set** that were noted and applied in our system.

The **lexical feature classes** that were considered as part of this heuristic process.

1) If the question have a “more” keyword followed with an adjective, then the question is identified as a comparative question.

   More + JJ (Adjective) -> JJR => Comparative question.

   Ex: Is Dell more compact than HP.

2) If the question has a “most” keyword followed with an adjective, then the question is identified as a comparative question.

   Most + JJ (Adjective) -> JJS => Comparative question.

   Ex: Which is the most expensive? Dell or Alienware?

3) If the questions has keyword such as “difference”, “versus/vs/VS/Versus”, the question is classified as comparative question.

   Difference between google and bing?

   Dell vs HP?

   Dell versus Alienware?

4) If the question has compare and if there are atleast 2 Nouns associated with it, then the question is classified as comparative question.

   Compare Dell and Apple laptop?

5) If there are nouns associated to the left and to the right of CC tag, then the question is classified as comparative question.

   Dell or/CC HP, need help?
   Red walls or/CC green walls for my room?

6) Checking the nouns with a radius of 2 around the "between" keyword

   What is the exquisite distinguishing feature between dog and cat?

Improved Implementation:

The heuristics method may sometimes misclassify a user's comparative question as a non-comparative question if there is no significant collection of feature classes. Hence the improved implementation proposed was to use both the pattern based and heuristics based method. The user's question would be run through a set of patterns to see if there is match and If there is no match, then heuristics rules would be applied to see if the question is a comparative question or not.

If the question is classified as comparative question by any one of the methods, then the comparative entities are extracted.

Ex: What would be the ideal paint for a guy's living room? Red or green paint.

Comparative entities: [Red, Green]

Alternatives: [Yellow, White]

Semantic Handling:

There may be cases where a user's comparative question would be like,

Dell or cookie, which is a better laptop for Gaming?

The comparative entities "Dell" and "Cookie" are irrelevant words and there is no similarity between them. To resolve these kind of ambiguities, We used freebase Google database which is a community curated database which is well known for famous people, places and things.

Freebase API was used to find the semantic similarity or common domain between the two entities.

For following entities, the following domain set was returned as a http response by freebase api.

Dell – [Electronic Computer Manufacturing Business, Computer Business, Computer hardware Business, Anthropologist, Publishing Business]

HP - [Computer hardware Business]

Hence, a set of possible domains were extracted from freebase database for the mined comparatives from the user's question and a validation is performed to check if there is a common domain between the two sets, and If there is, then the alternatives are mined accordingly.

The common domain for Dell and HP is Computer hardware business.

Results:

Pattern Method:

```
**************************************************************************
PATTERN METHOD:

COMPARATIVE ENTITY MINING:

Loading the data set and pre-processing it.........
Successfully Done in 1secs...!!!
Loading the data set and pre-processing it.........
Successfully Done in 0secs...!!!
**********************************
Pattern Method Evaluation Results:
Number of Comparative Questions found manually in the corpus: 328

Total No. of Questions: 2074
Total Questions Identified as Comparative: 172
Irrelevant Questions identified as Comparative: 34
Comparative Questions identified Correctly: 138
Comparative Questions not Identified: 190

Precision:  80.23255813953489%
Recall:           42.073170731707314%
F-Score:    55.199999999999996%
**********************************
```

Heuristics Method:

```
**********************************
```

```
HEURISTICS METHOD:

COMPARATIVE ENTITY MINING:

************************************
Heuristics Method Evaluation Results:
Actual number of Comparative Questions found manually in the question corpus
:328

Total No. of Questions: 2074
Total Questions Identified as Comparative: 275
Comparative Questions identified Correctly:233
Comparative Questions not Identified:95
Irrelevant Questions identified as Comparative:42

Precision: 84.72727272727273%
Recall: 71.03658536585365%
F-Score: 77.28026533996685%
```

User Inputs:

```
Loading the patterns...... Please Wait....

Loading the data set and pre-processing it.........
Reading POS tagger model from tagger/english-left3words-distsim.tagger ...
done [1.4 sec].
Successfully Done in 1secs...!!!

Pattern loading Complete....!!!!

Loading the data set.........

Enter your query:
Google or Bing, which is better?
ENTITIES FROM USER INPUT: google,bing

[bing, google, mozzila]
Continue? (Y/N)
Y
Enter your query:
Dell or HP, Which is better for Gaming?
ENTITIES FROM USER INPUT: dell,hp

[hp, dell, alienware]
Continue? (Y/N)
N
Thank you!!!!
```

Summary of the problems encountered during the project

1. Implementing phrase chunker was a little challenging task for us, as we had to analyze the different forms of questions and had to come up with the logic of grouping the nouns and cardinals to appropriate noun set.

   Ex: HP or Dell Inspiron model 1500 HD, which is better for gaming?

   In the above question, the complete Dell Inspiron model 1500 HD is one whole comparative entity and hence these noun forms were grouped into one Noun phrase.
2. The task of mining comparator entities from the comparative questions took a bit of our time as we had to explore the different types of comparative questions to see the nature in which they appear. At the end of this process, we defined a right set of heuristic rules to identify a question as a comparative question and mine the entities.

   There was one of the feature class which we framed as that If a question contains the keyword “compare/comparison”, then the question was tagged as comparative. But for examples,

   Who made a good comparison? – This is not a comparative question. Hence, we redefine the feature class as if the question contains a compare/comparison keyword, it must also contains nouns associated with a coordinating conjunction, then it is a comparative question.
3. Implementing weakly supervised bootstrapping was a challenging task as since the task involves mining the patterns and mining comparatives simultaneously, the questions from the dataset had to be removed once that patterns are extracted out of that question so that pattern duplication can be avoided.
4. We had to modify the output of the question pattern from Stanford pos tagger for the task of comparing the question with the patterns from the pattern archive.

   What are the differences between a struct in $c and in $c ?. This has to be pos tagged as

   What/WP are/VBP the/DT differences/NNS between/IN a/DT struct/NN in/IN $/$ c/CD and/CC in/IN $/$ c/CD ?/.

   But Ideally, the $c marks the noun entity because $c is replaced in place of comparative entity as part of the bootstrapping process. And when a question is compared with this pattern, then this pattern has to be tagged appropriately as below

   Ex: What/WP are/VBP the/DT differences/NNS between/IN a/DT struct/NN in/IN $c/NN and/CC in/IN $c/NN ?/.

5. For performing evaluation of the pattern based and heuristics based method, we manually collected a dataset and manually tagged each question in the dataset as comparative or non-comparative for F score metrics.
6. When we were attempting to find a semantic similarity between the extracted entities, WordNet model was used but then word net didn't have coverage upon all famous people, things and places. We were put to explore a lot around the web and found a research paper on finding semantic similarity using freebase. We had to tweak the api call to send the request and extract the entities information from the json response.

Pending Issues:

1) The NLP system fails to extract the comparative entities if the comparative questions contains acronyms. Acronym handling is not taken care.

   Ex: USA vs UK, which has good defense?
2) Semantic ambiguity still exists for comparative questions such as

   Apple or Samsung? Which brand is reliable? When alternatives are getting recommended for the entities Apple and Samsung, the other entities such as Orange, grape may get recommended with the algorithm if the comparator pairs such as <Apple, Orange>, <Apple, Grape> exists. Semantic handling has to be done when recommending alternatives as well.
3) Red or green walls for my room?

   The extracted entities are [Red, Green walls]. Both Red and Green must be associated with walls as they represent that, but it is not taken care of.

Future Work:

1) The Aliases of the entities should be handled more precisely and must be separated as per its context. Ex: Pizza vs Villa? (Pizza and Villa are Tamil movies). Pizza vs burger?(Pizza and burger are fast foods).
2) Automatic recommendation of alternatives must be suggested based on semantics. Ex: Dell vs Alien ware? Which is better for gaming?. Then the recommended alternatives should be the laptop names that have occurred in comparative questions asked by users relating to gaming experience. The other laptops which are not good in gaming should not be recommended just because the entity was compared.
3) The system assumes that the user input questions would be spelling free and syntactic error free. But the system must be modified to map the misspelled comparative entity to the appropriate entity using lemmatizer.