Comparative & Evaluative QA System in Tourism Domain

Pinaki Bhaskar
Department of Computer Science and Engineering, Jadavpur University,
Kolkata – 700032, India
+91-33-24361628
pinaki.bhaskar@gmail.com

Bidhan Chandra Pal
Department of Computer Science and Engineering, Jadavpur University,
Kolkata – 700032, India
+91-9434650420
bidhan.cstbesus@gmail.com

Sivaji Bandyopadhyay
Department of Computer Science and Engineering, Jadavpur University,
Kolkata – 700032, India
+91-33-24146648
Sivaji_cse_ju@yahoo.com

ABSTRACT

Comparative or Evaluative questions are categorized as non-factoid questions where user asked to compare between entities (for comparative) based on some criteria and constraints or asked to evaluate certain criteria of the entities (for Evaluative). The answers of this type questions can't be directly lifted from the underline document collection rather answers are hidden. To answer the Comparative or Evaluative questions system must have potentiality to understand the comparative evaluative expression and user needs. This paper contains trivial approach to give answer of Comparative or Evaluation questions.

Keywords

Comparative & Evaluative Feature (CEF); Comparative & Evaluative Question (CEQ); Domain Knowledge Base (DKB), Comparative Evaluative Question Answering System

1. INTRODUCTION

Answering of the Comparative or Evaluative questions needs some extra effort mainly because of two reasons. The first reason is the extraction of the Comparative or Evaluative keywords and features (CEF) from the question and syntactically and semantically analyzing them. Secondly, the non-quantifiable Comparative or Evaluative expressions have to be transformed into quantifiable criteria so that appropriate answer can be generated.

The component of Comparative or Evaluative questions are comparative or evaluative expression, entity, and constrains. The comparative question must have a comparative expression that mostly either adjective phrase (like good, better, best, cheap, cheaper, cheapest etc.) or as-as phrase (like as good as, as tall as etc.) or proceeded by much, many, more, most keywords. According to nature of comparison the comparative expression is of three types general, comparative and superlative and this is called degree of comparison. Evaluative expressions are asked to evaluate the entities depending on some criteria. Evaluative question needs semantic knowledge over the criteria. For example the question like "What is the age of Abraham Lincoln when he became the president of USA?" is the evaluative question and to answer this question system has to know the meaning of age for person entity and how it is been calculated. Evaluative expression contains quantifiers, evaluative keywords (like age, height, weight for person entity, morning, noon, evening, night for time entity etc.). Entities are the set of objects that are compared or evaluated according to criteria specified in comparative or Evaluative expression. For example entity of "What are the morning flights to Boston from NY?" is all flights that travel from Boston to NY. Constrains are the phrase that specified the user preference and their needs.

The Comparative & Evaluative questions answering system needs to understand the basic property of each entity and how the property relates with the entity. For example a person entity has physical property like age, weight, height etc. psychological property like behaviour, personality etc. economical feature like wealth, and so on. These all features can define a person. Now the Comparative Expression like "good man" can be derived by the linear combination of above properties. So it is important to acquire semantic knowledge (how they relate with each other and as well as with the entity) of all properties of the entities and knowledge to evaluate all these properties of that entity. Without this type of knowledge system couldn't compare between entity members. This prior knowledge stored in Knowledge Base of system so that system can semantically understand the properties and evaluates them. Requirement of this type of knowledge makes the system domain dependent.

Tourism domain has richest set of CEQs. After extensively studied CEQ's questions it was noticed that users usually have described their situations, plans, tour details after that they have placed their questions.

This paper will describe the answer extraction system that builds on tourism domain and can handle the entity like location, hotels, transportation, tourist spot etc. Here are some sample questions:

Q1: We plan to visit Andhra Pradesh in December. We live in Kolkata, and will start and end our journey at Vizag and have seven days in hand. We are three families with kids and our budget is moderate. *Kindly suggest an itinerary, which must include Araku Valley.*

Q2: My family is planning a trip to Khashmir in late October. We plan to spend six days there and will visit Srinagar, Gulmarg, and Pahalgam. *Can you suggest good hotel in range of Rs. 3000-4000?*

Q3: My husband, son and I want to visit Stuttgart, Heidelberg, Salzburg and maybe Munich in May 2010. We live in Mumbai. *Is it cheaper to fly to Frankfurt first or to Stuttgart?*

2. RELATED WORK

Till now, very few potential works have been done in answer extraction of comparative and evaluative question. There is very few proposed technique of analysis comparative and evaluative question.

C. Kennedy [2] proposed that comparisons might be in relation to properties within the same object, degree of comparisons of the same property between different objects, or different properties of different objects. The properties at stake in the comparison are embedded in the semantics of the words in the question, and possibly in the context that comes with the question.

Friedman [2] presents a general approach to process comparative expressions by syntactically treating them to conform to a standard form containing the comparative operator and the clauses that are involved in the comparison.

D. Olawsky [3] attempts to study the semantic context by generating a set of candidate interpretations of comparative expressions. Then, the user is prompted to choose among these to specify his intent.

Rose et. al. [4] and Saint-Dizier et. al. [5] have proposed the technique to handle comparative and evaluative question answering for business domain. They have proposed the procedure to identify the terms in the question based on which comparison or evaluation can be done.

Bidhan, et al [6] have been tried to identify the question focus, comparative and evaluative expressions, their nature of comparison, and entities who will be compared the features who will play an important role at the time of comparison, by using simple rule based technique in tourism related comparative evaluative question. He has denoted these main actors as a Comparative evaluative expression, Entity, and Constrains. Comparative and Evaluative expression (CEE) defines the type and mode of comparison, Entity are the objects based on which comparison is done and Constraints the user condition which guides the comparison.

In this paper we discussed a rule-based techniques to extract the answer of comparative and evaluative questions, which rose in tourism domain.

3. CHALLENGES

Answer extraction of comparative and evaluative question leads to several challenges.

3.1 Context Dependency of Comparative and Evaluative Expression

The comparative and evaluative expression carries different semantics in different context. And the parameter of comparison or evaluation also changed accordingly. For example the comparative criteria is different for comparative expression “best” in context of location and in context of hotels. So before comparison system must identify the context where comparative or evaluative expression is appears.

3.2 Predict the unspecified features

System will compare the entities based on their features. If some features are missing i.e. question does not explicitly mention the feature value then system has to predict the feature value with help of other feature value and comparative or evaluative expression. To do so, the system has to know the user intention. Here we inputted some rules so that system will predict those values. For example if user asked to give best hotels to stay with family within a certain budget for a specific location but didn’t mention the room type he/she preferred, then system will predict that room type will be double bed room or family suit. System predicts this value using the rule that a family person preferred to stay in double bed room or family suit.

3.3 Quantify non-quantifiable comparative or evaluative expression

Comparative or evaluative expression may be quantifiable (like cheapest, fastest, early etc.) or non-quantifiable (like best, suitable, comfortable etc.). System has to quantify the non-quantifiable comparative or evaluative expression. One way to quantify non-quantifiable expression is to divide the expression into features. For example “comfortable” expression for “Hotel” entity can be described as “various room types (like double bed, suit, air conditioned) which suits user as well as different hotel facility like restaurant, bar, summing pool etc. should be present, and also rents fall under the budget”.

3.4 Computing the value for comparison

System will compare the entity using the weighted linear combination of feature score. But it is a biggest challenge to decide the weight and the score of the individual feature. System has to know the importance of the feature and how to calculate the feature score. Feature value may be in strings or numerical or numeric range. System should have an appropriate technique to evaluate feature score. The weight value represents the importance of the feature in context of the question. Most significant feature has higher weight value.

3.5 Data Insufficiency

Data insufficiency is the biggest problem, which turns down the system performance. Tourism domain deals with constant and live data. We have found no way to get live data like cost of transportation (flight, train etc.), their timings, sometime constant data does not have sufficient information like types of room in hotels, weather information for a location etc.

4. QUESTION ANALYSIS

4.1 Question component

System has to identified following components:

4.1.1 Context of the question

The context of the question repesents user needs or demands. Context of the question is also important because same comparative and evaluative expression behaves differently in different context.

4.1.2 Comparative or Evaluative Expression (CEE):

CEEs are the phrases by which we can compare or evaluate entities.

4.1.3 Mode of comparison:

Mode of comparison is very important because it helps the selection of entities. There is three mode of comparison. They are general comparison, comparative comparison, and superlative comparison. General comparison is mainly yes/no type question. For example “Is Delhi good place to visit?” this type of question is rarely found and the question asked whether or not entity possess some criteria. Comparative comparison is asked to compare two entities or two set of entity with respect to some criteria. For example “Is ITC Sonar Bangla better than Taj Bengal?” so system has to compare hotel “ITC Sonar Bangla” and “Taj Bengal” and select which is better. Superlative type of comparison is asked to compare an entity with all other entities of same type or compare all the individual entity in entity set. For example “which is best hotel in Delhi?” has superlative type of comparison.

4.1.4 Entity:

Entities are the set of objects, which are been compared with respect to some comparative and evaluative expression. Accurate selection of entity has the direct relation with system performance. Entity can be group of same object. For example “Which is the best hotels in Delhi?” here entity is the all hotels situated in Delhi.

4.1.5 Constraints:

Constraints are the criteria, which guides the comparison. Constraint represents the user preferences. For example “Which are best family restaurant in Delhi?” here “family” is the constraints which guides the comparison between entities.

4.2 Extraction of Question Components:

To identified the component of question system first extracts the CEF (Comparative and Evaluative Features). CEF are the phreses that hold the meaning of the Question. We have identified 17 different CEF which can be present in the question. Here are some CEF types: LOCATION TO, LOCATION FROM, TIME TO GO, TEAM SIZE, BUDGET, ADJECTIVE MODIFIERS etc. The CEF are extracted from question using simple rules. For Example:

*We plan to visit (Andhra Pradesh)/**LOCATION_TO** in (December)/**TIME_TO_GO**. We live in (Kolkata)/ **LOCATION_FROM**, and will start and end our journey at (Vizag)/**LOCATION_FROM** and have (seven days)/ **TIME_LIMIT** in hand. We are ((three families)/ **TEAM_MEMBER** with kids)/**TEAM_DETAILS** and our (budget is moderate)/**BUDGET**. Kindly suggest an itinerary, which must include (Araku Valley)/ **MUST_INCLUDE_LOCATION**.*

The question component can be identified by following technique

4.2.1 Context of Question:

Context of question is identified by question classification. Questions are classified into seven classes like Itinerary, Accommodation, Transport, Getting Around, Time and Cost related classes. Rule based technique are used to classify question.

4.2.2 Comparative or Evaluative Expression (CEE):

Syntactically the CEEs are the adjective or adverb, which appears before noun chunk. Comparative expressions are mainly comparative or superlative adjective or adverb. Evaluative expressions are mainly countable noun or general adjective.

4.2.3 Mode of Comparison

Mode of Comparision are determined from Adjective Modifier CEFs. The rules to idetified Mode of comparision given in table 1.

4.2.4 Entity and Constraints

Entity and Constrains are identified from CEFs. Entity and constraints are different in different context. Table 2 shows the rule to identify entity and constraints. We fallow the approach of Bidhan et al (2011) to analysis the question and find the context, CEEs, mode of comparison, entity and constraints.

5. GENERAL APPROACH TO ANSWER EXTRACTION

In question analysis system could know the context on which question appears, the comparative and evaluative expression, their mode of comparison, entity and constraints. Now we have discussed our approached to extract the answer.

Table 1: Rules for Mode of comparision

#	Type of Comparison	Rules
1	General or Positive or Negative	1. Adjectives form the list {good, suitable, clean, new, appropriate, preferable, dirty, easy, happy, pretty, reasonable, bad, cheap, large, big, small. fast} appeared in noun chunk. 2. ADJP chunk staring with much or many
2	Comparative	1. Adjective with -er extention and appeared in ADJP chunk and followed by preposition than 2. Adjective or phrase inside as-as (like as soon as possible) 3. ADJP chunk staring with more 4. ADJP chunk containing too
3	Superlative	1. Adjective with -est extension appeared in Noun Chunk or ADJP chunk 2. ADJP chunk staring with most

Table 2: Rules for Entity and Constraints for different context

#	Context of Question	Entity to be Compared	Constraints to be considered
1	Itinerary	Location To, Location From, Must Include Location, Location Preference,	Time to Go, Time Limit, Budget, Purpose of Visit
2	Accommodation	Hotel Type	Location To, Location From, Must Include Location, Team Details, Budget, Hotel Specification, Purpose of visit
3	Reach Destination	Transportation Mode	Location To, Location From, Time to Go, Team Details, Budget,
4	Time related Info.	Time to Go	Purpose of Visit, Location To, Must Include Location, Location Preference
5	Getting Around	Getting Around Preference	Time to Go, Team Details, Budget, Purpose of Visit
6	Cost Related Info.	Location To, Must Include Location, Location Preference	Budget, Purpose of Visit

5.1 System Pre-requirements

Before extracting answer system needs Data and the knowledge to extract answer.

5.1.1 Data Collection

Data are collected in two ways. They are HTML static page collection and Live Feed information. Here we have collected static web page of wikitravel.org and live feed from LonelyPlanet.com, Kayak.com, etc.

5.1.1.1 Question Collection

Questions are collected from 'Ask Markopolo' section of 'OutlookTraverller' magazine. The comparative or evaluative questions are sorted out from collected questions. 186 different comparative evaluative questions and their answers are used as gold standard. Out of 186 questions 100 are used as training set and the others are used as evaluation. The questions have a pattern that user used to tell their situation then asked their question.

5.1.1.2 HTML Web Page collection

System crawled http://wikitravel.org/en website and stored 4366 webpage (size 183.5MB) that describe 4366 distinct tourist places around the world. Every Wiki-travel web page follows a unique writing format and related details are separated by their heading and always maintain same sequence of information (Understand → Get in → Get around → See → Sleep → Stay safe →Get out). Wiki-travel pages are easy to understand both syntactically and semantically because information is grouped into special HTML tag.

5.1.1.3 Database Table

Three database tables are used to select appropriate answer. They are location hierarchy table, document indexing table and airport and railway transport details table. Location hierarchy table contains the city name, and its corresponding provision or state name, country name, geo-location i.e. latitude, longitude, population, language speak, type of tourism etc. Document indexing table has three columns. In first column it contains address where document is residing, second column contain direct index i.e. the location name appeared in headings and third column contains indirect index i.e. the location name appeared in body of text. Airport and railway details table contain city name and its corresponding airport name, airport IANA code, railway station name, railway station code. Same city may have more than one airport and railway station name.

5.1.1.4 *Feed Collection*

Different websites has provided RSS feed to access their online information. System has taken information from live feed. Feeds that are available in tourism domain are discussed below:

- **Lonely Planet:** Lonely planet feed[1] gives co-ordinate (lat & long) of the place, point of interest (poi) of the place that include eat, sleep, see, shop, night, do.
- **Google Map:** Google Map gives the map of the location specified by geo co-ordinates. Google Map[2] also shows location that is specified by system in map.
- **Kayak:** Kayak feed[3] gives the flight information (time, fare, sit availability) if two locations are specified.

5.1.2 *Domain Knowledge Base Generation*

Domain knowledge base makes the system familiar with domain i.e. the semantic and syntactic pattern of phrases that will appear in that domain. As we work in tourism domain the domain knowledge base gives the tourism knowledge to the system. Domain knowledge base consists of human crafted rules so that system will know how a human is interpreted phrases, keywords and noun chunk in tourism domain.

Domain knowledge base serves the following purpose:

- **Extraction of features:** Syntactic rule to find the features from question as well as from entity. Question features are the context of the question, Comparative or evaluative keywords, entity and the constraints. Entity features are the properties of entity those describe the entity.
- **Prediction of unspecified features:** Using DKB system can predict the unmentioned features.
- **Decomposition of Entity information into entity-property:** An entity can be decomposed into series of feature or property of that entity which describe the entity. For example an hotel entity can be decomposed into following features:

$$\text{Hotel feature} = \left(\begin{array}{l}\text{Hotel Status}\\ \text{Variety of Rooms}\\ \text{Quality of Room}\\ \text{Rent of Rooms}\\ \text{Availability of other facility}\end{array}\right)$$

Now system will extract those values to quantify the entity.

- **Semantic importance of features**: Each question features has semantic importance according to the context it appears. DKB provides semantic knowledge of individual feature of question and their weight or significance with respect to different comparative expression.

DKB contains three types of rules.

- **Extraction rules:** Extraction rules help to classify the question, extract features from question, comparative evaluative expression, features from entity etc.
- **Initialization Rule:** Initialization rules help to initialize constrains those are missing in the question. Initialization rules take the helps of other feature to initialize constrains. Constrain plays important role to choose appropriate entity.
- **Decomposition rule:** Decomposition rules are helped the system to decomposed the entity as well as non-quantifiable comparative or evaluative expression into weighted sum or linear combination of features.

5.2 Answer Extraction Concept

The general idea of getting answer of comparative and evaluative question is to compare the question feature value with the

[1] Lonely planet website [Online] Available: http://developer.lplabs.com/index.php?title=The_Lonely_Planet_Content_API

[2] Google Map feed website [Online] Available: http://code.google.com/apis/maps/documentation/javascript/basics.html

[3] Kayak feed website [Online] Available: http://www.kayak.com/labs/api/search/

property value of each individual object in entity. To do so, system has to know all features value in the question, the property value of the entity objects, and the method of comparison. Here we described some of key concept to find most relevant answer of the question. System tries to find the top five answers instead of single answer.

5.2.1 Predict The Unmentioned Features of the Question

To extract the answer of the question, system requires all feature value of the question. The question features are categorized into two classes. They are independent or primary feature like (place to visit, budget etc.) or dependent feature (like type of hotel, hotel room preference etc.) System can predict the dependent features based on the constraints or other primary features. In table 3 we have shown some rules to predict a feature "Hotel room Preference" in different circumstances.

Table 3. Rules to predict unmentioned features using other features.

Team Member Type	Time of visit	Purpose of Visit	Accom-modation Type	Room Preference
Family	-	Tourism	Hotel	Double bed room
Family with kids	-	Tourism	Hotel	Family suit
Family	Summer	Tourism	Hotel	Double bed room with AC
Family with kids	Summer	Tourism	Hotel	Family suit with AC
Friends	-	Tourism	Hotel	Single Bed room, Dormitory

5.2.2 Extracting Individual Object of Entity

Proper entity selection is very important to get relevant answer of the question. Question might or might not mention the individual entity to be compared with. It is observed that superlative comparative question mention entity types or location where entity belongs e.g. "Which are the best hotels in Delhi?" here system has to compare between the hotel and finds the best hotels but question does not mentioned any hotel name. So system first finds the individual hotels in "Delhi" and their information before comparison. In three ways entity can be mentioned in the question:

i. **Location Specific Entity.** User mentioned the Location where entity belongs. For example "hotels in Delhi", "tourist spot near Delhi"

ii. **Type specific Entity.** User mentioned the type of entity. For example type of place entity like "honeymoon place", "hill place", type of hotel entity like "five star hotel" , time specific entity like "Summer", "Winter" etc.

iii. **Location & Type specific entity.** User mentions both entity type and the location where it is located. For example "five star hotel in Delhi".

Extraction of entity is some time very difficult. For example the questions like "What is the best honeymoon spot in India in summer?" It is very difficult to find proper entity because we don't have any ontology in tourism. System has no choice other than matching of words with the place description for that type of question.

5.2.3 Calculate the Feature Score

System will match the entity property value with its corresponding feature value of the question. The feature score is calculated by the amount of feature value matched with entity property. The feature is either contains string or numerical or numerical range. System calculates the feature score using the following formula:

- **String Valued Feature:** the feature score is ratio of number of key-phrase matched between feature value and entity property.

- **Numerical valued feature:** Numerical valued feature can be either numeric range or single numerical value. If the value is single numeric then question demands to check either greater than or lass than. Now there is four possibilities 1) feature value is range and property value is single or multiple numeric value: this is the most common case. We use Gaussian distribution and find the feature score for that numeric value. 2) Feature value is numeric range and property value is also numeric range: feature score is calculated by ratio of portion of the range is common and feature range. 3) Feature value is numeric and property value is also numeric: if less than is asked then we divide the feature value with property value, do opposite if greater than is asked. 4) Feature value is numeric and property value is numeric range: if less than is asked then we divide the feature value with difference between feature value and lowest property value, do opposite if greater than is asked.

5.2.4 Ranking Individual Entity

System will calculate the final comparative or evaluative score for each individual object in the entity. Final score is the sum of weighted feature score. The weight of feature is different for different comparative and evaluative expression. The weight of feature represents the meaning of comparative or evaluative expression as well as user preference. For example, for hotel entity "cheapest" comparative expression has the highest weight on budget feature, "comfortable" comparative expression has the highest weight on room type feature, and those features, user mentioned explicitly in the question, have higher weights than unmentioned features etc. System decides the weight of feature using the domain knowledge rules.

5.3 System Architecture

System Architecture has three parts 1. Question Analysis Part, 2. Answer Extraction Part and 3. Storage. Figure 1 shows the system architecture. Three sections are shown in different colour. System takes comparative or evaluative question. System gives the top five-ranked answer to the user. Here we briefly described answer extraction part.

5.3.1 Document Extraction and Getting Feed Result

Document extraction is done using the location mentioned in the question. If location type (e.g. coast area, hill, forest) is mentioned instead of specific location then system will find the location using location type. The document-indexing table has direct and indirect indexing. First system searches the location name in direct index if it not found in direct index then it goes to indirect index. If location name is matched then matched document is fetched using the path address.

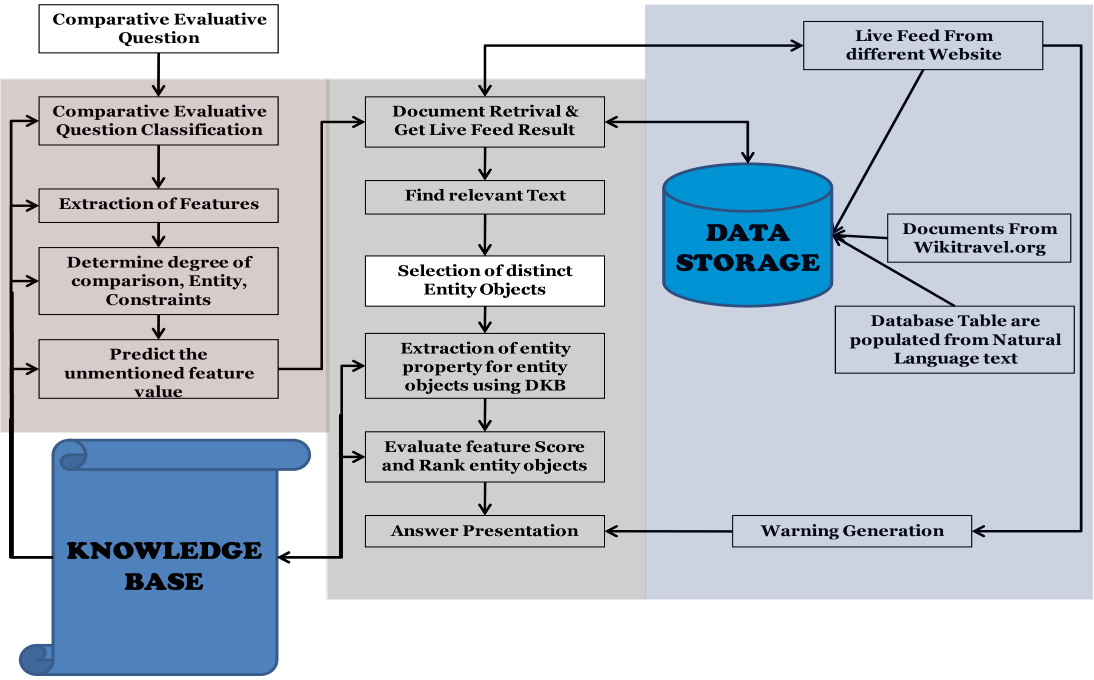
Comparative Evaluative Question
Comparative Evaluative Question Classification
Extraction of Features
Determine degree of comparison, Entity, Constraints
Predict the unmentioned feature value
KNOWLEDGE BASE
Document Retrival & Get Live Feed Result
Find relevant Text
Selection of distinct Entity Objects
Extraction of entity property for entity objects using DKB
Evaluate feature Score and Rank entity objects
Answer Presentation
Live Feed From different Website
DATA STORAGE
Documents From Wikitravel.org
Database Table are populated from Natural Language text
Warning Generation

Figure 1. Comparative and Evaluative Question Answering System Architecture

Feed gives the real time data like for transportation details i.e. flight information, train information are collected from Kayak feed and Tourist spot, hotel, shopping mall information collect from Lonely Planet feed using the location name.

5.3.2 Selection of Entity objects

The individual entity object is extracted form relevant information. System parses the relevant text portion to find individual member of the entity and the information associated with it. After parsing system build a distinct entity set using their name so that no member is repeated in the entity set.

5.3.3 Identification of entity property value

Each entity member has some property, which can describe the entity. The property of each entity member is extracted from the information associated with it using extraction rule in DKB. The property list for the entity is described on table 3.

5.3.4 Compute feature score

Each property of entity is matched with corresponding feature to calculate the feature score for all entity members. The calculation of feature score is described earlier. The final feature score of a entity member is the weighed linear combination of individual feature score. Weight of a feature depends on the comparative expression and the user preference i.e. the constraints. In case of evaluative expression the weights only depends on the user preference.

5.3.5 Answer Ranking & presentation

Each member of the entity is ranked according to their final feature score. The top five ranked member is shown as the answer of the question. The associated information of the member and the information source is also included in answer as a proof.

6. EVALUATION

Evaluation is done by precision and recall criteria. To make appropriate evaluation of system we have modified the rules to calculate precision and recall values.

6.1 Subjective Evaluation

Subjective evaluation is done manually. The human evaluator gives evaluation score from 0 to 4 according to their satisfaction, where 0 for worst and 4 for best. The precision and recall value is calculated by given formula.

- **Precision Measurement**: All the four evaluators gave a score between 0-5 for each output. '0' score for the worst output and '5' score for the best output. The precision for each evaluator is calculated with equation 1.

$$\text{Precision} = \frac{\sum_{i=1}^{no.\,of\,questions}\left(\left(\sum_{j=1}^{10} s_{ij}\times w_j\right)\div\left(10\times\sum_{j=1}^{10} w_j\right)\right)}{5\times no.\,of\ Input} \qquad (1)$$

  where, S_{ij} = score of jth answer of ith question and W_j = weight for jth answer.

- **Recall Measurement**: All the four evaluators gave a score in scale of 0-5 whether evaluator is satisfied with the output against given input. '0' score for complete dissatisfaction and Table 3. The property list and its extraction rule for entity '5' score for complete satisfaction. The recall for each evaluator is calculated with the equation 2.

$$\text{Recall} = \frac{\sum_{i=1}^{no.\,of\,questions} r_i}{5\times no.\,of\ Input} \qquad (2)$$

  where, r_i = score given for ith question.

Entity	Property List	Extraction Rule (Keywords)
Place	Type of Place	coast area, mountain, forest, historical
	Average weather	hot, humid, cold
	Number of places to see	-
	Variety of places to see	park, zoo, fort, palace, sun rise point, sun set point
	Language spoken	Language List
	Recommended tourism	historical, wild life list, preferred honey moon spot, wild safary
	Other tourism attraction	scuba diving, paragliding, cave jumping
Hotel	Hotel Status	5-star, 3 star, government, guest house etc
	Variety of Rooms	double bed, single bed, dormitory, suit, cottage, etc.
	Quality of Room	AC, Non-AC, Deluxe, super deluxe, kitchen
	Rent of Rooms	Cost expression = “$”, “USD”, “`”, “Rs”, “INR”, “£”, “EUR”, “€”, “GBP” etc. followed by Number Expression which consists tag “(CD”.
	Availability of other facility	Bar, Disco, casino, summing pull, restaurant, etc
Flight	Time of Departure	Inner HTML of Span class=“time” after origin
	Time of Arrival	Inner HTML of Span class=“time” after destination
	Duration of the flight	Inner HTML of span class=“duration”
	Cost of the ticket	Inner HTML of div class= “price”
Tourist Spot	Type of the tourist spot	Palace, fort, park, fountain, tomb, garden, temple, church, monastery, sea, beach, mountain, forest etc.
	Open and close time and date	Noun chunk contain (CD with keyword AM. & P.M
	Price of entrance	Cost expression
	Main attraction of that spot	Noun chunk contain keyword attraction, must see,
	Amount	Cost expression

6.2 Keyword Base Evaluation

Keyword base evaluation is done by matching keywords. The system generated output is matched with the gold standard output. The precision and recall value is calculated by the no of matching keywords with the standard percision and recall formula.

6.3 Evaluation Results

6.3.1 Subjective Evaluation

Subjective evaluation is done for the system to know the human satisfaction of system generated answer. Table 4 shows the result of subjective evaluation, which is done by the four expert human evaluators, for the entire system.

6.3.2 Keyword Base Evaluation

Answer Extraction part takes a tagged question as input and gives the most relevant text portion as the output of tagged question. Answer evaluation system is evaluated by the matching keywords between the human generated answer (gold standard answer) and system generated answer. Table 5 shows the evaluation results for different classes of question.

Table 4. Subjective Evaluation for entire System

Measurment	Evaluator 1	Evaluator 2	Evaluator 3	Evaluator 4	Avg.
Precession	0.80	0.72	0.65	0.71	0.72
Recall	0.92	0.83	0.79	0.87	0.85
F-score	0.86	0.77	0.71	0.78	**0.78**

Table 5. Evaluation result of answer Extraction Method

Question Context	Precision	Recall	F-score
Itinerary	82%	87%	84.4%
Accommodation	61%	85%	71.0%
Reach Destination	26%	31%	28.8%
Getting Around	59%	72%	64.8%

7. CONCLUSIONS AND FUTURE WORKS

System is somewhat biased because all the rules and the domain knowledge base is developed manually. In future machine learning technique will be used to extract the rules and developing the knowledge base and try to extract more comparative evaluative features from the question. The system also used structured “wikitravel” document for its document collection. In future we try to include unstructured web-base document and collect information from those document.

8. ACKNOWLEDGMENTS

The work has been carried out with support from Indo - French Centre for the Promotion of Advanced Research (IFCPAR) funded Project “An Advanced platform for question answering systems” (Project No. 4200-IT-1).

9. REFERENCES

[1] Kennedy C., Allen K..: Comparatives, Semantics of Lexical and Logical Semantics; Encyclopedia of Language and Linguistics, 2nd Edition, Elsevier, Oxford. (2006)

[2] Friedman C.: A General Computational Treatment of the Comparative. In: Proceedings of the 27th Annual Meeting of the ACL. (1989)

[3] Olawsky D.: The Lexical Semantics of Comparative Expressions in a Multi-level Semantic Processor, In: Proceedings of the 27th Annual Meeting on ACL, USA. (1989)

[4] Rose N., Lim T., Saint-Dizier P., Gay B., Roxas R.E.,: A Preliminary Study of Comparative and Evaluative Questions for Business Intelligence. In: International Symposium on Natural Language Processing (IEEE-SNLP), Bangkok. (2009)

[5] Saint-Dizier P., Roxas R., Rose N., Lim T.: Some Challenges in the Design of Comparative and Evaluative Question Answering Systems. In: ACL-KRAQ workshop, Singapore. (2009)

[6] Pal B.C., Bhaskar P., Bandyopadhyay S.: A Rule Based Approach for Analysis of Comparative or Evaluative Questions in Tourism Domain. In: ACL-KRAQ workshop, Tailand. (2011)