CHAPTER 3
SEMANTIC ANALYZER FOR QUESTIONS AND ANSWERS

3.1 INTRODUCTION

Many QAS supports syntactic analysis of questions and answers for providing the answers to the query. The syntactic analysis depends on the features of queries posted by the user. The extracted keyword supports for the retrieval of answers from the web sources using pattern matching method, passage retrieval or template based method. The keyword based retrieval helps for retrieval of answers based on their syntactic nature. But the relevant answers may not be able to retrieve because of the lack of semantic analysis. So, it is mandatory to analyze the questions as well as the answers semantically which helps to improve the accuracy of the system. Some QAS supports for semantic analysis, but they are lacking in reasoning of question and answers to map the suitable answer to the query. In this research work, the question and answers are pre-processed, updated to knowledge repository and retrieved with the support of case based reasoning.

3.2 FRAMEWORK FOR KBQA

The reasoning based question answering system is relying on the Knowledge Based Question Answering (KBQA) system for providing the most relevant answer to the query posted by the user. The conceptual diagram for KBQA is depicted in Figure 3.1 which consists of four phases includes the semantic analyzer for questions and answers, collaborative forum, mapping of questions and answers, and the ranking of answers for recommending the most relevant answers.

The semantic analyzer for questions and answers phase is the first phase of KBQA system, in which the questions and the answers are pre-
processed. The domain ontology is constructed which helps to improve the retrieval of answers from the repository. The keyword in the question and the answers are marked and retrieved with the keywords and its semanticness. Hence for a single question can have more number of relevant answers which are identified and retrieved.

The answers to every question are collected from the different sources like web, KBQA repository, and the collaborative forum. The KBQA repository is updated which supports to provide the updated most relevant answer for the question.

Case Based Reasoning (CBR) approach is applied in the KBQA system for retrieving the answers to the queries. Each query is considered as a case and they are analyzed. Based on the analysis, the relevant answers are retrieved from the repository with the help of features of the queries. If the answer for the question is not available in the repository or not meeting the threshold value, then the question is considered as new case. Then it is mandatory to update the ontology and acquire answers from the web for the query. Further, this question is passed on to collaborative forum.

In collaborative forum phase, the learners are allowed to give answers for the question posted by the user. The collaborative forum caters the need of the user query by providing the relevant answers for the question posted by the user. The question posted to the forum by using revised Send-A-Problem technique to provide more than one answer for a single query posted in the forum. The experts may provide answers to the query and the suitable answers are selected based on the content of the answers and the user’s opinion or voting. Based on the voting value as well as the threshold value, the answers are ordered in a prioritized way to map with the suitable question.
The retrieved answers are weighted with membership function based on the semantic closure. The member function is applied to every question and their answers are validated with fuzzification and defuzzification to improve the correctness of the answers. All the fuzzy rules are processed with the help of inference engine and the accepted answer will be stored in the knowledge base. The accuracy of the mapped question and answers are analyzed and the process is repeated until the suitable answer is produced by the KBQA system.

Figure 3.1: Conceptual diagram for KBQA

Consequently, in the **mapping phase**, the semantic rules are constructed for questions and the answers in the repository. Rule based questions and answers are mapped using LP-LSA (Logarithmic Probabilistic Latent Semantic Analysis). In this research work, LP-LSA is integrated with SPARQL (Simple Protocol And RDF Query Language) to retrieve the answers from the repository. Further, LP-LSA is also integrated with SWRL
(Semantic Web Rule Language) and Description Logic to create a rule based mapping system. During the time of validation, the correctness of mapping is checked based on their threshold value of answers for their relevance. If the validated answers are obtained the threshold value, then the answer is considered as the suitable answer otherwise it is irrelevant to the query.

All the relevant answers need to be ranked for the given query to recommend the most relevant answer to the user's query. In the **ranking phase**, all the answers are ranked based on dynamic NDCG method. Based on the relevance score obtained, the ranking process is initiated and processed through NDCG to get the most relevant answer to the query posted by the user. The answers which get the highest value are declared as the most relevant answer and the paired question and answer will be stored in the KBQA repository and directed towards the user's query.

### 3.3 DOMAIN ONTOLOGY CONSTRUCTION

Ontology learning is a knowledge acquisition activity that relies on automatic methods to transform unstructured data sources into conceptual structures. It provides the semantic meaning of concepts and defines the relationship between the concepts, which acts as the knowledge base for the domain. Developing ontology is an iterative process. It can be grown from the first pass to 'n' pass based on the search. Concepts in the ontology reflect the real domain that is being modelled.

Ontology development can be evaluated at each and every stage. The domain has to be selected with care and importance of the ontology, by enumerating the important terms and concepts of the domain. The hierarchy among the classes, properties of the class and relationship among the classes plays a major role in developing the ontology. In this research, the
computer subject “Data mining” is taken as the domain for the ontology and it is named as ONTO-KBQA that acts as the repository of the KBQA system.

While constructing ontology, the standards of W3C are adhered. It specifies a set of components, to define the class and their properties from which few are highlighted:

- **Class** - A type (Class) of elements.
- **Individual** - A class instance.
- **Object Property** - A relationship between two elements (for instance between two Individual).
- **Data Property** - Data associated with an Individual.
- **Annotation Property** - An annotation associated with an element.
- **Data Type** - A specific kind of data (for instance xsd: string).

There are basically two ways of describing classes such as prescriptive approach and definitional approach. The prescriptive approach allows for expressing a number of properties of a class whereas the definitional approach allows for providing the definition of a class. The questions, question type, the answers and its relationship are defined in the ONTO-KBQA with relations such as as is-a, part of, hasQuestion, hasAnswer, hasType relations etc. These above relationships are helped in defining the key feature and their semantic relations.

### 3.3.1 Characteristics of ONTO - KBQA

The ONTO-KBQA preserves the following characteristics to support the KBQA system.

- **Memory**
• Dynamism
• Polysemy
• Automation

While constructing the ontology, the concepts and their properties are stored in the memory with the semantic relations. The question ontology is constructed; their terms and concepts are preserved for the construction of answer ontology.

Whenever the ONTO-KBQA is identifying the new concepts and terms, the ontology is updated automatically to preserve dynamism. In polysemy characteristic, the ONTO-KBQA is having the ability to provide the polysemous forms of terms and concepts. Every term may have different meanings and every meaning of the word is preserved in the ONTO-KBQA. For example, the non wh words are mapped together for easy retrieval of the answers with the support of polysemy. In automation, the ONTO-KBQA is able to generate and enrich the ontology automatically. Based on the domain, the ontology has to be created automatically with the required information.

**3.3.2 Steps for Ontology construction**

Any ontology can be constructed by three methods such as single ontology approach, multiple ontology approach, and hybrid ontology approach. The single ontology approach uses single global ontology that shares the vocabulary and the terminology. Multiple ontology approach is having the combination of different local ontology. It will collect the local ontology from the different sources. Hybrid ontology approach may follow single and multiple ontology. The ONTO-KBQA is constructed with the support of multiple ontology approach which is having question ontology, answer ontology and mapping ontology. The different ontologies are
constructed and merged based on the user’s query and the answers are retrieved with the support of SPARQL query.

Figure 3.2: Domain Ontology for Data Mining Concepts

The Figure 3.2 shows the ONTO-KBQA construction for data mining domain. The KBQA system supports to construct domain ontology concepts to extract the domain attributes and associations from a set of relevant...
documents. Every class, subclass, individual, data property, object property and their relationship is constructed. The collected information is stored in the ONTO-KBQA which acts as the repository of the KBQA system.

The relationship between classes is depicted in figure 3.3 which shows the correlation between every class associated with ONTO-KBQA. It is necessary to have the relationship between every class, their subclass and their individuals which supports to find the keyword.

![Figure 3.3: Relationship between Classes](image)

The Ontology construction algorithm is explained in figure 3.4 which consists of three sets. Let Q be the set of Questions $Q = \{q_1, q_2, q_3, \ldots, q_n\}$, A be the set of Answers $A = \{a_1, a_2, a_3, \ldots, a_n\}$ and K be the set of terms $K = \{k_1, k_2, k_3, \ldots, k_n\}$. The domain “Data Mining” and their key terms are identified. For every term, the class, individuals, data property and object property are identified and Subject-Predicate-Object relationships are assigned. After analysing the basic requirements, the ONTO-KBQA is constructed Whenever a new term is arriving, the system will check the relation of the term with the ONTO-KBQA. Based on the relationship, the term is updated with the
ontology with the support of revise ontology. Every time the ONTO-KBQA is updated at the time of receiving the new term.

```latex
//Algorithm for Ontology Construction

Procedure OntologyConstruction()

Input: Term from the user

Output: Updated ONTO-KBQA repository

Begin

Let Q= \{q_1, q_2, q_3, \ldots, q_n\} be the generalised Domain of Question

A = \{a_1, a_2, a_3, \ldots, a_n\} be the Answer and K=\{k_1, k_2, k_3, \ldots, k_n\} be the Keyword

Step 1: Identify the Domain

Step 2: Enumerate the key term K

Step 3: Identify the Subject-Predicate-Object of the term

Step 4: Define the Class, Individuals, Object and Data Property

Step 5: Apply the suitable relationship of the terms

Step 6: Construct the ONTO-KBQA

Procedure Update ONTO-KBQA()

Step 7: If K is in the ONTO-KBQA

Step 8: Return

Step 9: Else

Step 10: Identify the suitable relation and revise the ONTO-KBQA

Step 11: Endif

End
```

Figure 3.4: Algorithm for Ontology Construction

The ONTO-KBQA is constructed with the support of protégé and the correctness of the ontology is preserved with the help of reasoners. At the time of construction, the inconsistent ontology is identified and removed with
the support of pallet reasoner. So the KBQA system provides reliable and authenticated answer for the question posted by user.

Table 3.1: Comparison of Reasoners

<table>
<thead>
<tr>
<th>Attributes</th>
<th>FaCT++</th>
<th>HermiT</th>
<th>Pellet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Language</td>
<td>C++</td>
<td>Java</td>
<td>Java</td>
</tr>
<tr>
<td>Methodology</td>
<td>tableau-based</td>
<td>hyper tableau based</td>
<td>tableau based</td>
</tr>
<tr>
<td>Soundness</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Completeness</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Expressivity</td>
<td>STROIQ(D)</td>
<td>STROIQ(D)</td>
<td>STROIQ(D)</td>
</tr>
<tr>
<td>Incremental Classification-addition</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Incremental Classification-removal</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Rule Support</td>
<td>No</td>
<td>SWRL</td>
<td>SWRL</td>
</tr>
<tr>
<td>Justifications</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>A Box Reasoning</td>
<td>Yes</td>
<td>Yes</td>
<td>SPARQL</td>
</tr>
</tbody>
</table>

There are three reasoners supports to construct the ontology such as FaCT++, HermiT and Pellet. The list of reasoners which are used in protégé was listed in table 3.1 which shows the comparison of the reasoner. When comparing the features of these three reasoners pallet is having better features other than the two reasoners. The KBQA system hardly depends on the rules which support to check the consistency of the ONTO-KBQA. FaCT++ does not support the rules which are used in the KBQA system. HermiT is supporting for SWRL but is not giving the justification for the inconsistent ontology. In every aspect, Pallet is having the better features
and it is used for the construction of ONTO-KBQA. There is a need to check the inconsistent ontology with the support of Pallet reasoner and the reasoner is actively monitoring the ontology at the time of construction.

**Pellet** is a reasoner developed by tableau based decision procedure and it contains two debugging services that help to preserve the consistency. The first debugging service is to identify the root contradiction of the graph and the second debugging facility is to support for tracing the axiom written by the user which helps to rectify the error in the axiom of the ONTO-KBQA. So the correctness of the system is preserved at the time of constructing the ontology.

![Figure 3.5: Asserted Model without Reasoner](image)

Figure 3.5: Asserted Model without Reasoner
Figure 3.6: Inferred Model for Question Type
The KBQA system uses Pallet reasoner as the default reasoner which highlights the bugs made by the user. At the time of constructing the ONTO-KBQA, the Pallet reasoner takes the collection of axioms written in OWL and offers a set of operations on the ontology's axioms which helps to validate the ONTO-KBQA. It is very helpful for the user to correct the error immediately and it assures that the KBQA system provides the error-free knowledge base.

The process of retrieval of relevant information from massive collection of documents, either multimedia or text documents is still a cumbersome task. Multimedia documents include various elements of different data types including visible and audible data types structural elements as well as interactive elements.

Once the ontology is constructed, the protégé supports to discover the asserted model which is depicted in figure 3.5. Asserted model is the model of ONTO-KBQA which is not relying on the reasoner. But the research strongly believe in reasoners which gives the strong recommendation for the users. After applying the Pallet reasoner, the asserted model will be converted into the inferred model which is depicted in figure 3.6. Every data property, object property, classes, subclasses and the individuals of the ONTO-KBQA passed through this reasoner and applied all the rules which are relevant to the user's query. Then the result will be displayed which is having more relationship with every data property, object property, classes, subclasses and the individuals.

3.4. QUESTIONS AND ANSWERS ANALYZER

The Figure 3.7 shows the framework for KBQA question processing which consists of three parts such as question classification, question analysis, and keyword extraction. The question posted by the user is pre
processed to classify the question and its types. The question type may be the wh question or nonwh question.

![Diagram of KBQA Question Processing](image)

Figure 3.7: Framework for KBQA Question Processing

Once the question type is identified, the syntactic and semantic information about the question are extracted with the support of ontology. From the extracted keyword, the question is retrieved from the question pool with the support of ontology. The pallet reasoner is applied to check the correctness of the processed question.

Question is a linguistic expression which consists of an interrogative form, addressed to someone in order to get answer in reply. It is a form of words addressed to a person in order to elicit information. In question pre-processing stage, the question structure, the question hierarchy and the question classification need to be analyzed. The figure 3.8 shows the structure of the question. The questions are classified into domain and non domain question and corresponding to their relevant and irrelevant question.
The Question hierarchy deals with the different kind of questions such as yes/no questions, wh questions, non wh questions, factoid question, choice question, quantity question, enumeration question, description question, and multiple choice questions. Based on the question hierarchy, the question needs to be classified into various group of question type which is used for retrieving more precise answers. The objective of this is to identify the category of answer that the user is searching for. The major question classification in this research is wh questions and non wh questions. Wh question and nonwh question are easily processed by the KBQA system.

Normally, Non wh questions are harder to process and it requires more processing method than wh questions. But KBQA system is designed to identify the nonwh question easily by mapping it with the wh question. For that it is necessary to understand the relations between concepts. The question classification is illustrated in the figure 3.9 in which the questions are broadly classified into Wh questions and Nonwh Questions.
The part of question ontology is represented in the figure 3.10, in which the class QA acts as the parent class and it consists of three sub class such as the question class, the question type class and the answer class. The question class is classified into wh and nonwh classes and the classification of both the types was clearly depicted in the question ontology.
There is a need to map the other question type into the “Wh” category. The question with define, explain, state, describe and how to define are mapped into what interrogative pronoun. The mapping ontology of other question type is depicted in the figure 3.11.

Figure 3.11: Mapping of Nonwh Question Type.

After the analysis of the question, the answer keyword has been identified based on their interrogative word or question word. Interrogative word is a function word which supports for classifying the questions such as who, when, what, where, whom, why and how. It also supports to identify the direct as well as the indirect questions. The question focus can be identified by looking at the question word or a combination of the question word and its succeeding words. For example, both the question word 'when' or the combination 'what time' indicates a temporal aspect which has to be found in the answer set. The 16 types of question words are identified and analyzed for getting the answer keyword. The domain ontology construction for the interrogative pronoun is depicted in the figure 3.12 and the ontograf for question type is illustrated in the figure 3.13.
Figure 3.12: Domain Ontology Construction for interrogative pronoun

1. who/whom - PERSON
2. when - TIME/DATE
3. where/what place - LOCATION
4. what time (of day) - TIME
5. what day (of the week) - DAY
6. what/which month - MONTH
7. what age/how old - AGE
8. what brand - PRODUCT
9. what - NAME
10. how far/tall/high - LENGTH
11. how large/big/small - AREA
12. how heavy - WEIGHT
13. how rich - MONEY
14. how often - FREQUENCY
15. how many - NUMBER
16. how long - LENGTH/DURATION
Figure 3.13: Ontograf for Question Type

3.4.1 Syntactic Analysis of Questions

Syntactic analysis is the process of analyzing the question words and their relationship with the support of POS tagging (Part-Of-Speech). Syntactic information can be extracted from tagging and parsing. Tagging is the task of labeling each word in a sentence with its appropriate part-of-speech like noun, verb, adverb, adjective etc. Parsing is the task of describing the structure of a sentence. The parser output is usually a tree structure with a sentence label as a root, various phrase labels as intermediate nodes, words or sentences in the sentence as leaf nodes.

To analyze the question, the KBQA system utilizes, a POS tagger which helps to identify the question words (why, what, how, where, when, which) and the keywords from the user query. The POS tagging for the sample question “What is Data Mining? Why it is required?” is shown in figure 3.14. The “what” interrogative word acts as the interrogative pronoun and the “why” interrogative word acts as the interrogative pro adverb. To
deepen the analysis process, the question is processed semantically with the support of parser.

<table>
<thead>
<tr>
<th>What- Pronoun (WP)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is- Verb (VBZ)</td>
</tr>
<tr>
<td>Data- Noun (NN)</td>
</tr>
<tr>
<td>Mining- Noun (NN)</td>
</tr>
<tr>
<td>Why-Adverb (WRB)</td>
</tr>
<tr>
<td>It- Pronoun (WP)</td>
</tr>
<tr>
<td>Is-Verb (VBZ)</td>
</tr>
<tr>
<td>Required-Verb (VBN)</td>
</tr>
</tbody>
</table>

Figure 3.14: POS tagging of Question

3.4.2 Semantic Analysis of Questions

Semantic analysis is the process of analysing the question based on their syntactic information. It is relating the structures and occurrences of the words, phrases, clauses, paragraphs in the particular question. Depending on question type, the definition terms are analyzed. For “What”, “Which”, “How”, and similar questions these terms are detected by selecting noun phrases appearing next to the Wh-term. Once question type and definition terms are analysed, the system generates the semantic context of the expected answer.

Semantic analysis requires rules to be defined for the question as well as the answers. The concept of linguistics is this sentence formation has a structure in it called Subject-Predicate-Object (S-P-O). Where "apple" is subject, "is" is predicate and "red" is an object. When question type has been successfully mapped to a top concept, only terms related to this concept will
be added to the term context representation. The table 3.2 shows the 13 types of TREC question categories.

Table 3.2: Question Category

<table>
<thead>
<tr>
<th>Question Categories</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Causal Antecedent</td>
<td>How does exception handling works in Java?</td>
</tr>
<tr>
<td>2. Goal Orientation</td>
<td>For what purposes destructor is used in Java?</td>
</tr>
<tr>
<td>3. Enablement</td>
<td>How does Java handle integer overflow and underflow?</td>
</tr>
<tr>
<td>4. Causal Consequent</td>
<td>What happened when the object was deleted?</td>
</tr>
<tr>
<td>5. Verification</td>
<td>Is sizeof a keyword?</td>
</tr>
<tr>
<td>6. Disjunctive</td>
<td>Does objects reusable or disposable?</td>
</tr>
<tr>
<td>7. Instrumental/Procedural</td>
<td>How to install java in windows 7?</td>
</tr>
<tr>
<td>8. Concept Completion</td>
<td>What is encapsulation?</td>
</tr>
<tr>
<td>9. Expectational</td>
<td>Why iterator doesn’t have a method to get next element directly without moving the cursor?</td>
</tr>
<tr>
<td>10. Judgmental</td>
<td>What should try block do when the exception occurs?</td>
</tr>
<tr>
<td>11. Quantification</td>
<td>How many basic interfaces are there in Java Collections Framework?</td>
</tr>
<tr>
<td>12. Feature Specification</td>
<td>What is the color of Java icon?</td>
</tr>
<tr>
<td>13. Request</td>
<td>Would you pass the arguments?</td>
</tr>
</tbody>
</table>

Based on the POS tagging, the parse tree is constructed with the help of Parser to understand the relationship. The sample parse tree for the question “What is data mining? Why it is required?” is represented in Figure 3.15 and 3.16.
3.5 ANSWER EVALUATION

The Figure 3.17 shows the framework for answer processing where the user is posting the question to get the answers. The answers are collected from the various sources and it is analyzed semantically. The answer is evaluated with KBQA similarity measure, collaborative forum and fuzzy reasoning. The question and its relevant answers are mapped with the support of rule based mapping. Finally, all the relevant answers are prioritized by NDGC ranking method and the most relevant answer is provided to the user.
3.5.1 KBQA Similarity Measure

Keyword extraction is the task to identify a small set of words, key phrases, keywords, or key segments from the user’s question that can describe the meaning of the answer. As a result of the question analyzing, the key feature of the question has to be extracted and analyzed with the support of term frequency analyzer. Let \( Q \) be the set of Questions \( Q=\{q_1,q_2,q_3,...,q_n\} \), \( A \) be the set of Answers \( A=\{a_1,a_2,a_3,...,a_n\} \) and \( K \) is the set of Keywords \( K=\{k_1,k_2,k_3,...,k_n\} \). Based on the extracted keyword in each answer, the count matrix is created. The row of the matrix acts as the count of the word and the column matrix represents the key term of the search keywords. Now the weight calculation is accomplished with the support of TF-IDF.

TF-IDF supports to find out the similarity between the question as well as the answers by calculating the weight of the keywords. The TF part intends to give a higher score to a question or the answer that has more occurrences of a term, while the IDF part is to penalize words that are popular in the whole collection of the questions and the answers. The TF-IDF
is calculated for the question and the answer and the count matrix is obtained based on the formula eqn 3.1 to eqn 3.3.

\[
TF(k) = \frac{X}{Y}
\]

\textit{eqn (3.1)}

Where \(X\) is the Number of times keyword ‘k’ appears in an Answer and \(Y\) is the Total number of terms in the Answers

\[
IDF(k) = \log_e \left(\frac{A}{B}\right)
\]

\textit{eqn (3.2)}

Where \(A\) is the Total number of Answers and \(B\) is the Number of Answers with keyword ‘k’ in it

From equation (1) and (2), TF- IDF can be calculated as

\[
TF\text{-IDF} = TF(k) \times IDF(k)
\]

\textit{eqn (3.3)}

Where \(TF(k)\) is the frequency of a term in the given document, \(IDF(k)\) is Inverse Document Frequency.

Based on the keyword, the answer is retrieved from the ONTO-KBQA with the support of features of questions. If the answer is available in the ONTO-KBQA, then it will be exchanged to the user. If not, the question is passed through the collaborative forum.

The keyword retrieval algorithm is illustrated in figure 3.18 which consists of three sets, \(Q\) be the set of Questions \(Q=\{q_1, q_2, q_3, \ldots, q_n\}\), the Answer set, \(A\) be the set of Answers \(A=\{a_1, a_2, a_3, \ldots, a_n\}\) and \(K\) be the set of
terms \( K = \{ k_1, k_2, k_3 \ldots k_n \} \) \( r \) be the relationship between the two sets \( A \) and \( B \). The Questions and Answers for the domain “Data Mining” is identified and their Subject-Predicate-Object are assigned. The posted question is searched based on the keyword in the repository. If the keyword is found, then the matching answer will be retrieved otherwise the KBQA system depends on the collaborative forum to get the answers.

\[
//\text{Algorithm for Keyword Retrieval} \\
\text{Procedure KeywordRetrieval()}
\]

\text{Input: Question from the user}

\text{Output: Keyword from the Question}

\text{Begin}

\text{Let} \( Q = \{ q_1, q_2, q_3, \ldots q_n \} \) \text{be the generalised Domain of Question}

\( A = \{ a_1, a_2, a_3, \ldots a_n \} \) \text{be the Answer, } \( K = \{ k_1, k_2, k_3 \ldots k_n \} \) \text{be the Keyword and } \( r \) \text{be the Relationship between } Q \text{ and } A

\text{Step 1: Get the Input Query from the User}

\text{Step 2: Find the Keyword from the User}

\text{Step 3: If (the Keyword exists) Then}

\text{Step 4: If } A \subseteq Q \text{ Then}

\text{Step 5: } r = A \cap Q.

\text{Step 6: Check the ONTO-KBQA for matching the Keyword.}

\text{Step 11: End if}

\text{Step 12: Else}

\text{Step 13: Return}

\text{Step 14: Endif}

\text{End}

Figure 3.18: Algorithm for Keyword Retrieval
Table 3.3: TF - IDF for sample set of Questions

<table>
<thead>
<tr>
<th>Question</th>
<th>Syntactic Analysis</th>
<th>Semantic Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>1.289</td>
<td>2.577</td>
</tr>
<tr>
<td>Q2</td>
<td>5.155</td>
<td>5.155</td>
</tr>
<tr>
<td>Q3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Q4</td>
<td>3.433</td>
<td>3.433</td>
</tr>
<tr>
<td>Q5</td>
<td>2.577</td>
<td>2.577</td>
</tr>
</tbody>
</table>

The sample set of questions and answers are has been analyzed both syntactically and semantically. The results of TF - IDF for the analyzed question and the answers were illustrated in the table 3.3 and 3.4

Table 3.4: TF - IDF for sample set of Answers

<table>
<thead>
<tr>
<th>Question</th>
<th>Syntactic Analysis</th>
<th>Semantic Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>0.0006</td>
<td>0.011</td>
</tr>
<tr>
<td>A2</td>
<td>0.0014</td>
<td>0.0082</td>
</tr>
<tr>
<td>A3</td>
<td>0.0019</td>
<td>0.0023</td>
</tr>
<tr>
<td>A4</td>
<td>0</td>
<td>0.014</td>
</tr>
<tr>
<td>A5</td>
<td>0.0038</td>
<td>0.076</td>
</tr>
<tr>
<td>A6</td>
<td>0</td>
<td>0.0074</td>
</tr>
<tr>
<td>A7</td>
<td>0.0014</td>
<td>0.0095</td>
</tr>
<tr>
<td>A8</td>
<td>0.0095</td>
<td>0.019</td>
</tr>
<tr>
<td>A9</td>
<td>0</td>
<td>0.0094</td>
</tr>
<tr>
<td>A10</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
3.6 PERFORMANCE EVALUATION

The KBQA system is taken 500 questions from the dataset of yahoo answers, SPSS and the results were analyzed in different performance metrics. The ONTO-KBQA was constructed and the sample of “Wh” and “Non Wh” questions was preprocessed and stored in the repository. From the keywords, the answer is retrieved from the ONTO-KBQA repository with the support of case based reasoning. From the analysis, the inference was made where semantic analysis gives the better result than the syntactic analysis and also semantic analysis improves performance of the KBQA system.

Table 3.5: Keyword Analysis for Questions and Answers

<table>
<thead>
<tr>
<th>Keywords</th>
<th>Questions</th>
<th>Answers</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Syntactic Mapping</td>
<td>Semantic Mapping</td>
</tr>
<tr>
<td>Computer</td>
<td>290</td>
<td>311</td>
</tr>
<tr>
<td>Data Mining</td>
<td>430</td>
<td>472</td>
</tr>
<tr>
<td>Knowledge</td>
<td>72</td>
<td>86</td>
</tr>
<tr>
<td>Java</td>
<td>122</td>
<td>122</td>
</tr>
<tr>
<td>Information</td>
<td>40</td>
<td>54</td>
</tr>
<tr>
<td>Cluster</td>
<td>257</td>
<td>301</td>
</tr>
<tr>
<td>Data Cleaning</td>
<td>78</td>
<td>91</td>
</tr>
<tr>
<td>Outlier</td>
<td>108</td>
<td>127</td>
</tr>
<tr>
<td>Spatial Database</td>
<td>114</td>
<td>131</td>
</tr>
<tr>
<td>Association Rule</td>
<td>243</td>
<td>267</td>
</tr>
</tbody>
</table>
The questions and answers were analyzed both semantically and syntactically and the numerical results for answers were depicted in the table 3.5 and the graphical representation for the numerical value is shown in the figure 3.19. The sample of 2000 question keywords was analyzed both semantically and syntactically. From the results, the keywords were analyzed and it has been enhanced in semantic analysis with the support of ONTO-KBQA.

From the results, the inference was made was that semantic mapping of the question and answer analysis helps to improve the mapping of keywords than syntactic analysis. Hence, KBQA system concentrates more on the semanticness of the questions and answers posted in the system.

![Figure 3.19: Keyword Mapping of Answers](image)

The result for the KBQA system is discussed briefly in the different aspect to produce the better performance for the system. The interface was created for the KBQA system and the users are allowed to post the questions
in the system and also they can search for the question with the support of KBQA system. The sample answer pool for the searched question is shown in the figure 3.22. Consequently, the answers were analyzed both semantically and syntactically and the results are depicted in the table 3.5 and figure 3.23 which inferred that semantic mapping helps to improve the mapping of keywords than syntactic mapping.

Figure 3.20: Posting of Questions in KBQA

Figure 3.21: Searching for Questions in KBQA
To illustrate the performance of the KBQA System, the evaluation system employs a set of performance metrics, which are depicted in the table 3.6.

Table 3.6: Performance Metrics

<table>
<thead>
<tr>
<th>Performance Metrics</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>$Precision = \frac{TP}{TP + FP}$</td>
</tr>
<tr>
<td>Recall</td>
<td>$Recall = \frac{TP}{TP + FN}$</td>
</tr>
<tr>
<td>F Score</td>
<td>$F Score = \frac{2(Recall \times Precision)}{Recall + Precision}$</td>
</tr>
<tr>
<td>Accuracy</td>
<td>$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$</td>
</tr>
</tbody>
</table>

Where TP - True Positive, TN - True Negative, FP - False Positive, FT - False Negative.
The Performance Evaluation for the question analysis is increased semantically than syntactic analysis. The KBQA system obtains better accuracy due to the analysis of semantic information. By utilizing the information from the dynamic knowledge sources, the proposed ontology based approach creates a greater impact on the semantic relatedness measurement, which facilitates to determine the inherent contextual information about each question and the answers. Hence, KBQA system preserves the semanticness of the question as well as the answers.

The performance metrics were applied to both wh and nonwh question types. From the analysis, wh questions are giving more accurate result than the nonwh questions. Each question type was analyzed by both syntactically and semantically and the results were compared. Semantic analysis always gives a better result than the syntactic analysis and the results were shown in the figure 3.23. The contextual meaning of the question keyword is preserved in the semantic analysis whereas syntactic analysis is matching only the exact keyword which is extracted from the user’s question.
Table 3.8: Performance Evaluation for TF-IDF

<table>
<thead>
<tr>
<th>Metrics</th>
<th>Syntactic Analysis(%)</th>
<th>Semantic Analysis(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>74.1</td>
<td>82.2</td>
</tr>
<tr>
<td>Precision</td>
<td>72.6</td>
<td>75.3</td>
</tr>
<tr>
<td>Recall</td>
<td>73.8</td>
<td>74.4</td>
</tr>
<tr>
<td>F-Measure</td>
<td>72.6</td>
<td>76.8</td>
</tr>
</tbody>
</table>

Consequently, the performance metrics were applied to TF-IDF methodology. Each question’s keyword was analyzed by both syntactically and semantically and the results were compared. Semantic analysis always gives a better result than the syntactic analysis.

Figure 3.23: Performance Evaluation for TF-IDF
3.7 SUMMARY

In this chapter, the framework for KBQA system, its phases were deliberated clearly. Consequently, it has focused on the ONTO-KBQA domain ontology construction which acts as the repository of the KBQA system. The semantic analysis of the questions as well as answers was processed by both syntactically and semantically. The keyword was extracted with the support of TF-IDF and the performance was evaluated by the performance metrics. The accuracy was improved in semantic analysis than syntactic analysis. The processed questions and answers were retrieved based on case based reasoning based on their relevance.