Chapter 2

TEXT MINING BACKGROUND AND THE USE OF GRAPH THEORY

Text Mining\(^1\) is an exciting area of research which uses techniques borrowed from data mining, machine learning, information retrieval, natural-language understanding, case-based reasoning, statistics, and knowledge management, to help the people to gain rapid insight into large quantities of semi-structured or unstructured text. In the earlier chapter, we have introduced the core contributions related to the thesis. Now, in the current chapter, we cover the basics of text mining, use of graph based techniques on text mining and important research issues which overlaps with the thesis.

Figure 2.1: Basic Text Mining Applications

The typical tasks of text mining can be given as (included, but not limited to) (1) Document Clustering, (2) Document Classification, (3) Document organization and (3) Information extraction { i.e. Document summarization, Automatic Summarization Evaluation, Keyphrase extraction, Automatic question answering and automatic evaluation of descriptive answers, essays etc.}. In this thesis, we consider unsupervised text mining applications, explore the impact of graph based techniques on these applications and introduced the social graph based effective core techniques for these applications. As, unsupervised approaches do not require learning or training or user involvements in the entire execution.

of the system, so we consider only unsupervised approaches in this thesis. The brief description some core text-mining techniques (which are used in the thesis) is given below:

**Chapter Organization**: We start this chapter with the graph based representation of text and future research directions (Section 2.1). In Section 2.2, we present the brief description of document summarization (single and multi-document summarization), current research status, and important research issues. Similarly, in Section 2.3, we discuss the automatic evaluation of machine generated summary and text answers. We also discuss about current research status and important research issues. In Section 2.4, we discuss about keyphrase extraction (including current research status, and important research issues). In Section 2.5, we discuss about document clustering (including current research status and important research issues). Finally, in Section 2.6, we discuss about automatic question answering (including current research status and important research issues)

### 2.1 Graph Based Representation of Text

Most of the graph based unsupervised techniques either uses words/tokens (sometimes filtered by some linguistic constraints) and sentences as nodes of the graph.

#### 2.1.1 Using words/tokens as nodes of the graph

Mihalcea and Tarau, (2004), introduce the unsupervised way to build a graph that represents the text, and interconnects words or other text entities with meaningful relations. The main aim was to enable the application of graph-based ranking algorithms to natural language texts. To add the link between words/tokens, (Mihalcea and Tarau, 2004) are using a co-occurrence relation, controlled by the distance between word occurrences. According to them, two vertices are connected if their corresponding lexical units co-occur within a window of maximum words, where can be set anywhere from 2 to 10 words. Liu et al. (2010) used the same type of graph and apply Topical PageRank (TPR) to measure word importance with respect to different topics. Wan and Xiao, (2008) also used the same type of word graph of text for automatic keyphrase extraction. The following Table 2.1 contains the sample sentences from INSPEC abstract and Figure 2.2 contains the sample graph.
Table 2.1 Sentences from Inspec abstract

| S1. | Compatibility of systems of linear constraints over the set of natural numbers. |
| S2. | Criteria of compatibility of a system of linear Diophantine equations, strict inequations, and nonstrict inequations are considered. |
| S3. | Upper bounds for components of a minimal set of solutions and algorithms of construction of minimal generating sets of solutions for all types of systems are given. |
| S4. | These criteria and the corresponding algorithms for constructing a minimal supporting set of solutions can be used in solving all the considered types systems and systems of mixed types. |

Figure 2.2: Word graph, for the text given in Table 2.1 (Mihalcea and Tarau, 2004)

Other approaches: Litvak and Last, (2008), introduced the use of forward directed word graph of text for keyphrase extraction and document summarization.

2.1.2 Using sentences as nodes of the graph.

Mihalcea and Tarau, (2004), introduced the use of sentences as nodes of the graph. Next, to draw the link between any two nodes, it introduces the use of content overlap between two sentences (vertices). The overlap of two sentences can be determined simply as the number of common tokens between the lexical
representations of the two sentences, or it can be run through syntactic filters, which only count words of a certain syntactic category, e.g. all open class words, nouns and verbs, etc.

According to Mihalcea and Tarau, (2004), other sentence similarity measures, such as string kernels, cosine similarity, longest common subsequence, etc. are also possible.

**Example:** We use the sentences given in Table 2.1 to prepare the undirected text graph. In this case, we add the link between any two nodes (sentences), if both sentences show the cosine similarity value greater than 0.3. We use the cosine similarity score as the edge weight for the given two nodes.

![Figure 2.3 Graph by using sentences as the node of the graph.](image)

**Other approaches:** Erkan and Radev, (2004), use cosine similarity to identify the undirected link between two sentences (treated as vertex). However, instead of the traditional cosine similarity score, it uses IDF (inverse document frequency) induced cosine similarity score. The calculated weight is used as the link weight, between the two given sentences (nodes). Wan and Yang, (2006) present an improved affinity graph based multi-document summarization. Formally, we can say that, for a given a sentence collection \( S = \{s_i \mid 1 \leq i \leq n\} \), the affinity weight \( \text{aff}(s_i, s_j) \) between a sentence pair of \( s_i \) and \( s_j \) is calculated using the cosine measure. The weight associated with term \( t \) is calculated with the \( \text{tf}_t \times \text{isf}_t \) formula, where \( \text{tf}_t \) is the frequency of term \( t \) in the corresponding sentence and \( \text{isf}_t \) is the inverse sentence frequency of term \( t \), i.e. \( 1 + \log \left( \frac{N}{n_t} \right) \), where \( N \) is the total number of sentences and \( n_t \) is the number of sentences containing term \( t \). If sentences are considered as nodes, the sentence collection can be modeled as an undirected graph by generating the link between two sentences if their affinity weight exceeds 0, i.e. an undirected link between \( s_i \) and \( s_j \) \((i \neq j)\) with affinity weight \( \text{aff}(s_i, s_j) \) is constructed if \( \text{aff}(s_i, s_j) > 0 \); otherwise no link is constructed.
2.1.3 Important Research Issues

From the above discussion, it is clear that the way of forming graph and use of supporting information in ranking, are a main part of majority of page rank based ranking schemes. Similarly to successfully apply other graph based techniques, we need to populate the graph with other information. The following contains the brief summary of some research points (which we have identified):

I. Due to the relatively higher occurrence frequency, lower length N-grams, show higher statistical influence in keyphrase extraction. A latest survey of state-of-the-art unsupervised keyphrase extraction techniques (Hasan and Ng, 2010), also concluded that the way used to form phrases affects the robustness of the most of the graph based techniques (even they are not as robust as TF-IDF based schemes). So, we need to concentrate on different types of graph formation also.

II. Our study shows that to effectively answer any question, we need the information related to the semantic correlation and relatedness between (a) terms of the given question and candidate answer and (b) terms of the candidate answer (if any). We effectively explore this relation in the development of the automatic answering system for Why-based questions.

III. Similarly, as graph based systems do not preserve the information related to the position of words/sentences, etc. So, we need to explore the possibilities of utilizing the statistical and semantic relation between terms/words/sentences.

2.2 Document Summarization (Single and Multi-document Summarization)

Document Summarization gives the information about main topics covered in a document or theme of document in a very limited number of words or sentences. The document summarization techniques can be categorized into (1) Query focused summarization or guided summarization, (2) Generic summarization and (3) Hybrid summarization.

Query focused summarization helps user in searching relevant information (user specific information) in a condensed form. Generic summarization techniques give the information about main topics covered in a document or theme of the document without any additional clue. Such summaries are very useful in browsing tasks, as it gives the brief information about entire text. The hybrid approach is the mixture of both techniques.

However, based on the document length, it can be categorized as (1) Single document summarization and (2) Multi-document summarization.
Single document summarization techniques generally present top 100, 200 words or fewer top extracted sentences from any given document, as a summary. Next, multi-document summarization techniques generally present 100-200 words or fewer top extracted sentences as summary for a given set of documents.

Similarly, based on the nature of presentation, we can categorize it as (1) extractive summarization (contains extracts summary) and (2) Abstract summarization (contains an abstract summary). Extract summarization technique presents, top extracted sentences as a summary. Most of the summarization technique presents extract summaries. Abstract summarization technique, generally combines the information spread in different sentences into a few sentences and present the compact sentences as summary (Kumar et al, 2013).

2.2.1 Literature Survey on Single Document Summarization (Graph based and Other Approaches)

A lot of efforts have already been applied to improve the effectiveness of the single document summary. Some of them are related to graph based, some others are non-graph based. A brief literature survey of some latest related work in this area is given below:

2.2.1.1 Graph based approaches

Most of the graph-based unsupervised methods use either sentence or word as node of the graph. For example: Mihalcea and Tarau, (2004) use every sentence as node of graph and apply Page Rank’s “random surfer model” to rank the sentences. Wan and Yang, (2007), incorporate the cross-document relationships between sentences in a cluster and finally apply graph ranking based summarization algorithm.

Some recent approach on abstract summarization methods like: Filippova, K. (2010), introduce the sentence abstraction scheme, by using a word graph of text. The algorithm uses k-shortest path algorithm and filter all those paths, which are shorter than 8 words or do not contain a verb. It uses offset position of words and occurrence frequency in the calculation of edge weight and includes salient words (selected based on occurrence frequency) in shortest path. Loret and Palomar, (2011), propose a method which uses a compendium to decide, which sentences are suitable to include in abstract sentence(s). However, it depends upon the frequency and page rank score of words.
2.2.1.2 Other novel approaches

CollabSum (Wan and Yang, 2007): For a given cluster of documents, it designed three summarization methods based on the use of cross-document relationships between sentences in the cluster of documents.


2.2.2 Literature Survey on Multi-document Summarization

2.2.2.1 Graph based approaches

A lot of methods have been proposed for multi-document summarization. The most frequently used techniques among all proposed methods are the use of sentence vector representation (where each row represents a sentence and each column represents a term) and graphs based methods (where each node is a sentence and each edge represents the pairwise relationship among corresponding sentences). Finally, all these methods rank the sentences according to their scores calculated from a set of predefined features, such as term frequency inverse sentence frequency (TF-ISF) (Radev et al. 2004); (Lin and Hovy, 2002), sentence or term position (Yih et al. 2007), and number of keywords (Yih et al. 2007).

2.2.2.2 Other novel approaches

Some state of the art methods with key features are: centroid-based methods (e.g., MEAD (Radev et al. 2004)), graph-ranking based methods (e.g., LexPageRank (Erkan and Radev, 2004)), non-negative matrix factorization (NMF) based methods (e.g., (Gong and Liu, 2001)), Conditional random field (CRF) based summarization (Shen et al. 2007), and LSA based methods (Gong and Liu, 2001).
2.2.3 Important Research issues

From above discussion, it is clear that an appropriate combination of statistical and local importance of words can be very effective in the summarization process (both single and multi-document summarization). As, single document contains less information w.r.t., set of documents used for multi-document summarization, so both can be treated separately. Similarly, in the case of multi-document summarization, we need to handle high volume of noisy entries w.r.t., single document summarization. Some common algorithm, works well with single document summarization, but does not work well with multi-document summarization. Additionally, extract based summary contains less volume of information w.r.t., abstract summary, but contains very few works.

2.3 Automatic Evaluation of Machine Generated Summary and Text Answers

Evaluation of machine generated summaries and text answers is a very important research area. The main aim behind the evaluation of machine generated summary is to judge the quality on the basis of some strict evaluation metrics like: (1) Correlation with the manual metric, (2) Discriminative Power compared with the manual metric and (3) Readability.

Similarly, the evaluation of text answers is related to the scoring of descriptive answers and essays on the basis of single or multiple model answers. We basically focus on the strength of topics covered in test answers and essays, relevance and correctness of information etc., in entire scoring.

2.3.1 Literature Survey on Automatic Summarization Evaluation and Important Research Issues

Human evaluation for text summarization is time consuming, costly, and prone to human variability (Nenkova et al. 2007). Thus, the importance of Automatic evaluation of text summaries increases. The brief introduction of some novel works of this area can be given as:

Current state-of-the-art techniques such as manual pyramid scores (Nenkova et al. 2007) or automatic ROUGE metric (considers lexical N-grams as the unit for comparing the overlap between summaries (Lin and Hovy, 2003)) use human summaries as reference. Hovy et al. (2005) and Hovy et al. (2006) proposed basic elements based methods (BE), it facilitates matching of expressive variants of syntactically well-formed units called Basic Elements (BEs). The
ROUGE/BE toolkit has become the standard automatic method for evaluating the content of machine-generated summaries.

2.3.2 Important Research issues

The ROUGE/BE toolkit has become the standard automatic method for evaluating the content of machine-generated summaries, but the correlation of these automatic scores with human evaluation metrics has not always been consistent and tested only for fixed length human and machine generated summaries. Still, there is a significant gap in quality between human evaluation and these automated metrics. Other issue is the role and sense of matching word(s), used by such techniques, which may not be same.

2.4 Keyphrase Extraction

Keyphrases are sequences of words, which captures main topics or theme of the document. Keyphrases can be considered as a highly condensed summary of documents and it is a core task for several important information retrieval and information extraction tasks, like: summarization, document clustering and indexing etc. (Witten et al. 1999), (Turney P., 1999).

Based on the approaches applied, keyphrase extraction can be divided into three categories:

I. **Free indexing based approach:** In this approach selection of keyphrases do not depend on vocabulary support of related areas. On the basis of the use of AI techniques, it can be further divided into (a) Learning based Witten et al. (1999), Turney P., (1999) and (b) Non-Learning based approach (Mihalcea and Tarau, 2004), (Grineva et al. 2009), (Wan and Xiao, 2008). Non-Learning based approaches generally use statistical information, grammatical facts, heuristics and lexical information of a given language in keyphrase extraction.

II. **Controlled indexing based approach:** In this approach keyphrases are chosen from a controlled vocabulary (a dictionary, thesaurus, or a list of terms). For example (Medelyan and Witten, 2005) and (Medelyan and Witten, 2006).

III. **Knowledgebase supported approach:** In this approach (1) Ontology (Nguyen and Phan, 2009) and (2) semantic relatedness score calculated by using Wikipedia or other knowledge resources are used in ranking (Li et al. 2010). Additionally, some approaches use Wikipedia to create a semantic graph for ranking (Tsatsaronis et al. 2010) and (Li and Li, 2011).
A latest survey of generic unsupervised keyphrase extraction techniques (Hasan and Ng, 2010), divides the keyphrase extraction scheme into three operational stages. Actually, most of the generic unsupervised keyphrase extraction approaches come under, either (1) free indexing based approach or (2) knowledge base supported approach. The stages are:

I. **Candidate lexical unit selection**: This is the first operational stage and used to filter out unnecessary word tokens from the input document and generate a list of potential keywords using heuristics. (Mihalcea and Tarau, 2004), (Grineva et al. 2009) and (Wan and Xiao, 2008).

II. **The lexical unit ranking**: At this operational stage, depending on the underlying approach, each candidate word is represented by its syntactic and/or semantic relationship with other candidate words. The relationships were defined by using co-occurrence statistics, external resources (e.g., neighbourhood documents, Wikipedia), or other syntactic clues.

III. **Keyphrase formation**: In the final step, the ranked list of candidate words is used to form keyphrases. A candidate phrase, typically a sequence of nouns and adjectives, is selected as a keyphrase if (1) it includes one or more of the top-ranked candidate words (Mihalcea and Tarau, 2004), (Grineva et al. 2009), or (2) the sum of the ranking scores of its constituent words makes it a top scoring phrase (Wan and Xiao, 2008).

### 2.4.1 Literature Survey on Keyphrase Extraction (Graph based and Other Approaches)

The brief description of techniques used in automatic keyphrase extraction can be given as:

#### 2.4.1.1 Graph based approach

The following contains the graph based approaches for automatic key phrase extraction:

Liu et al. (2009), propose to cluster candidate words based on their semantic relationship to ensure that the extracted keyphrases cover the entire document. The objective is to have each cluster represent a unique aspect of the document and take a representative word from each cluster.

In the TextRank algorithm, Mihalcea and Tarau, (2004), a text is represented by a graph. Each vertex corresponds to a word type. A weight, $w_{ij}$, is assigned to the edge connecting the two vertices, $v_i$ and $v_j$, and its value is the number of times the corresponding word types co-occur within a window of W words in the associated text. The goal is to (1) compute the score of each vertex, which reflects its importance, and then (2) use the word types that correspond to the highest scored vertices to form keyphrases for the text. Page rank scheme is used to calculate the rank score of each node/word.
SingleRank (Wan and Xiao, 2008) is essentially a TextRank approach with three major differences. First, SingleRank graph has a weight equal to the number of times the two corresponding word types co-occur. Second, while in TextRank only the word types that correspond to the top-ranked vertices can be used to form keyphrases, in SingleRank, they do not filter out any low-scored vertices. Rather, it (1) scores each candidate keyphrase, which can be any longest-matching sequence of nouns and adjectives in the text under consideration, by summing the scores of its constituent word types obtained from the SingleRank graph, and (2) outputs the N highest-scored candidates as the keyphrases for the text. Finally, SingleRank employs a window size of 10 rather than 2.

ExpandRank (Wan and Xiao, 2008) is a TextRank extension that exploits neighborhood knowledge for keyphrase extraction. For a given document ‘d’, the approach first finds its ‘k’ nearest neighboring documents from the accompanying document collection using a similarity measure (e.g., cosine similarity). Then, the graph for ‘d’ is built using the co-occurrence statistics of the candidate words collected from the document itself and its k nearest neighbors.

Liu et al. (2010), considers that both documents and words can be represented by a mixture of semantic topics. It builds a Topical PageRank (TPR) on the word graph to measure word importance with respect to different topics. After that, given the topic distribution of the document, it further calculates the ranking scores of words and extracts the top ranked ones as keyphrases.

Tsatsaronis et al. (2010), presents a semantic graph creation technique by using WordNet and Wikipedia, called a semantic graph. Finally, it applies different variations of PageRank and HITS to rank the words for keyphrase extraction, called semantic ranking. It also applies semantic ranking techniques for document summarization.

2.4.1.2 Other approaches

The following contains the other novel unsupervised approaches for keyphrase extraction.

Kea (Witten et al 1999), another remarkable effort in this area, identifies candidate keyphrases using lexical methods calculates feature values for each candidate, and uses a machine learning algorithm to predict which candidates are good keyphrases. The machine learning scheme first builds a prediction model using training documents with known keyphrases, and then uses the model to find keyphrases in new documents.

Kumar and Srinathan, (2008), use LZ78 based dictionary preparation step and statistical pattern filtration step to identify the candidate keyphrases (also denoted as candidate N-grams). Finally, it uses simple heuristics, grammatical and statistical features, to rank the identified candidate keyphrases.
Tomokiyo and Hurst, (2003), uses point-wise KL-divergence between multiple language models for scoring both phraseness and informativeness, which can be unified into a single score to rank.

Technique, like: (Zha, H. 2002) use mutual reinforcement principle and sentence clustering.

2.4.2 Important Research issues

From the above discussion, it is clear that, we can divide the entire keyphrase extraction technique into two important steps, i.e., (1) Candidate keyphrase identification and (2) ranking of the identified candidate keyphrases. Due to the variation of document topics, size and presence of a lot of noisy entries, identification of correct candidate keyphrases has become a challenging task.

2.5 Document Clustering

Document clustering is a knowledge discovery technique, which categorizes the collection of documents into meaningful groups or sets. It is very important for several text mining techniques (e.g., efficient organization and document summarization, etc.) and allows user to browse the target quickly.

According to Filippone et al. (2008), clustering techniques can be roughly divided into two categories:

- **Hierarchical Clustering**
- **Partitioning Clustering**

Hierarchical clustering techniques are able to find structures which can be further divided in substructures and so on recursively. The result is a hierarchical structure of groups known as a dendrogram.

Partition based clustering methods try to obtain a single partition of data without any other sub-partition like hierarchical algorithms do and are often based on the optimization of an appropriate objective function.

Based on the way to represent the document, the document clustering algorithms can be divided into the following three categories:

1. **Document vector based model for document clustering**: Document vector model is an algebraic model used to represent the document in the form of vector of identifiers. In this model, we use rows to represent documents and columns to represent terms. Here, we represent each document through a single row, where each column represents a distinct word from the given document. However, to calculate the weight of words we use features like: (1) occurrence frequency of words in the document, (2) binary representation (i.e., presence or absence or words
in the document) and (3) tf-idf score of words (Aggarwal and Zhai, 2012) (Zheng et al. 2009). Ontology and semantic enhancement techniques are used to enhance the performance of document vector model (Aggarwal and Zhai, 2012), (Zheng et al. 2009). The, next less popular model is, probabilistic model. In this model documents are represented by means of a probability distribution of terms $P(t_1, ..., t_m)$. This model specifies the probability $P(d_j | L_c)$ that the document $d_j$ belongs to the category $C$. (Aggarwal and Zhai, 2012). Finally, we apply various clustering algorithms, like Partitional clustering algorithms (e.g., K-means), Hierarchical clustering algorithms (e.g., single link, complete link and group average agglomerative clustering algorithms) etc.

2. **A graph based model for document clustering:** The most of the graph based clustering algorithms, treats either document or sentences as node of the graph and then use features related to common words in deciding link weight. Finally, it applies graph cut algorithms (Aggarwal and Zhai, 2012) (Filippone et al. 2008). However, in the case of kernel and spectral way of document clustering, we again depend upon partitional clustering algorithm to get the final result (Filippone et al. 2008).

3. **Using a knowledge base to represent document:** such methods use Wikipedia or other knowledge base (Saramaki et al. 2005), (Huang et al. 2008), (Huang et al. 2009) or prior classes (document categories) of knowledge base in document clustering (Hu et al. 2009). Some uses word-Net or ontology to enrich the document representation. However, most of such methods enrich the quality of document representation achieved by using previously discussed methods.

### 2.5.1 Literature Survey on Document Clustering and Important Research Issues

Generally, we use data clustering algorithms (Tan, Steinbach and Kumar, 2006) to cluster the documents. For this we use either vector space or graph representation of documents. For a past few years, there has been an increase in the use of additional knowledge resources like WordNet and Wikipedia, as external knowledge base. So, we go through such techniques only. Most of these techniques map the documents to Wikipedia, before applying traditional document clustering approaches (Kaufman and Rousseeuw, 2009) and (Tan et al. 2006).

Huang et al. (2009), create a concept-based document representation by mapping terms and phrases within documents to their corresponding articles (or concepts) in Wikipedia. They also developed a similarity measure that evaluates the semantic relatedness between concept sets for two documents.
Huang et al. (2008), applies supervision, using active learning. First, it utilizes Wikipedia to create a concept-based representation of a text document, with each concept associated with a Wikipedia article. It then exploits the semantic relatedness between Wikipedia concepts to find pairwise instance-level constraints for supervised clustering, guiding clustering towards the direction indicated by the constraints. Banerjee et al. (2007), proposed a method for improving the accuracy of clustering short texts by enriching their representation with additional features from Wikipedia.

(Hu et al. 2009), addresses two issues: enriching document representation with Wikipedia concept and category information for document clustering. Next, it applies agglomerative and partitional clustering algorithm to cluster the documents.

2.5.2 Important Research issues

Most of the work discussed above uses either bag of word based model or bi-gram based model. If we depend upon bag of words based approach, then two phrases having different meaning will treat common words as match count. For example, “Taj Mahal Hotel” and “Taj Mahal Tea” will show two matches (i.e., “Taj” and “Mahal”). To solve this issue we can use the phrase based approach. But due to differences in writing strategy, people may use different phrases to represent the same topic or concept. Next, most of the above discussed clustering algorithms use the word-based matching to calculate the similarity score. However, it is important to note that several matching words may have different levels of importance.

2.6 Automatic Question Answering

Automatic question answering is the task of automatically answering questions posed in natural language. All such tasks use some reference(s), either in the form of text document, web page or answer database etc. to answer the questions.

Based on the type of questions, we can divide the entire question answering system into two parts, i.e. (1) Factoid based question answering and (2) Non-Factoid based question answering. Hovy et al. (2002) Classified non-factoid type questions into three types: why type, definition type and how type.

Most of the factoid based question answering contains usually only one or at most a few correct answers to a given question, and the answer in most cases is a single word token or a short noun phrase. The system returns one or more ranked answer candidates for each question, and they are judged manually for correctness.
While, different from factoid based question answering, the answers to non-factoid based questions vary from a single sentence to a paragraph. Answers to these questions contain a reason or a cause and in the majority of the cases, answers may not contain a high density of overlapping phrases with the question. Additionally, finding the optimal length of answers of such question is an open challenge.

### 2.6.1 Literature Survey on Automatic Question Answering

The task of Question Answering has mainly focussed on answering factoid questions, where the answers are usually short phrases such as, named entities. Focus has recently moved towards the task of non-factoid question answering, such as, “why questions”, “how-to questions” and definition based questions, etc. The brief introduction to some of the novel works of this area can be given as:

Higashinaka and Isozaki, (2008), rank candidate answer paragraphs for answering why-questions in Japanese. It uses Support Vector Machines and features like: content similarity, causal expressions, and causal relations from two annotated corpora and a dictionary were extracted. Oh et al. (2012), also presented a supervised technique, which used sentiment analysis and word classes to answer why-questions in Japanese.

Verberne et al. (2010), proposed a three-step model for why-QA: (1) a question-processing module that transforms the input question to a query; (2) an off the-shelf retrieval module that retrieves and ranks passages of text that share content with the input query; and (3) a re-ranking module that adapts the scores of the retrieved passages using structural information from the input question and the retrieved passages.

Although significant work has not been done towards answering why-questions, a large number of approaches have been suggested in the past for the task of open domain question answering. (Berger et al. 2000); (Jeon et al. 2005); (Riezler et al. 2007) bridge the lexical chasm between the question and the answer using a machine translation model. Such data may not always be available and in order to obtain a reasonable performance, these techniques require large amounts of such data. Tellex et al. (2003), presents a BOW-based model, which uses statistical weights based on term frequency, document frequency, passage length, and term density. Quarteroni et al. (2007), consider the problem of answering definition questions. They use predicate–argument structures (PAS) for improved answer ranking. Verberne et al. (2009), compared a number of machine learning techniques in their performance for the task of ranking answers that are described by TF-IDF, a set of 36 linguistically-motivated overlap features and a binary label representing their correctness.
2.6.2 Important Research issues

From the discussion till now, it is clear that (1) dependency on matching terms, or (2) the dependency on semantic or ontology based technique, which expands the concept without properly capturing its relation with other terms in the question, may not give effective improvements in this area. The automatic identification of the optimal size of the answer is also an important research issues, as till now all the available techniques either consider passage (paragraph) or set of 5-6 consecutive sentences as the answer to any question. However, the answer to such (non-factoid based) question may be of any length.