CHAPTER 7

GRAPH BASED SUMMARIZATION

In this chapter, we describe two summarization tasks: single document summarization and query focused multi-document summarization based on graph based approaches. These tasks make use of the UNL semantic graphs of the document/documents from which the compressed summary graph is generated. We focus on the abstractive summarization method where the original texts rephrased and produces a compressed form of text. In this chapter, we describe a bootstrapping approach for the identification of important components of the fully connected document semantic graph for summarization. The proposed bootstrapping approach learns the graph operations such as selection, insertion and modification, that can be performed on the nodes and edges of the semantic graph. This bootstrapping approach is applied to two summarization tasks: single document summarization and query focused multi-document summarization which we will describe in this chapter.

7.1 INTRODUCTION

With a wide variety of documents available in the web, text summarization is one of the important tasks, which effectively compresses the information in a document(s). Text Summarization can be classified as extractive and abstractive methods. An extractive summarization method consists of selecting important sentences, paragraphs etc. from the original document to produce a compressed form of the original text. The importance
of the sentences is decided based on the statistical and linguistic features of sentences. In contrast, an abstractive summarization method consists of understanding the original text and rephrasing it into different forms without changing the meaning conveyed in the original text, but in a compressed form of a summary. When compared with an extractive summarization, abstractive summarization is a difficult and challenging task, which requires the semantic representation of the text, inference rules and natural language generation (Erkan & Radev 2004).

In general, the text summarization task is performed at various levels, such as the surface, entity and discourse levels (Hahn & Mani 2000). Surface-level approaches tend to represent information in terms of shallow parsers which can then be selectively combined to yield a selection function used to extract important information. Entity-level approaches (Mani & Maybury 1999) build an internal representation of the text, modeling text entities and their relationships. Text entities are units of texts, such as words, phrases, sentences or even paragraphs. These approaches tend to represent patterns of connectivity in the text to help determine what is salient. Discourse-level approaches (Mann & Thompson 1988) model the structure of the text and its relation to communicate goals.

In general, highly salient sentences are extracted from the document set, based on syntactic and/or statistical features (Erkan & Radev 2004), whether the approach to summarization is rule based or machine learning based. Machine learning approaches to summarization can be categorized into unsupervised and supervised. The unsupervised method is mainly based on scoring sentences in the documents by combining a set of predefined features (Mani & Bloedorn 1998; Conroy et al 2006). Contrastingly, in the supervised method, summarization is treated as a classification or a sequence labeling problem and the task is formalized as
identifying whether a sentence should be included in the summary or not (Shen et al 2007). However, the method requires a large set of training examples which are not available for resource-constrained languages.

In this chapter, we propose an abstractive summarization task, where the summaries are generated from a document semantic graphs. The proposed approach, initially starts the process with the UNL semantic graphs constructed from a Tamil text (described in section 3.4). The summary can be generated in any language from the compressed summary graph, where the language specific natural language generation rules are required for the formation of summary sentences. In this chapter, two summarization tasks are described: single-document and query focused multi-document summarizations.

7.2 RELATED WORK

In this section, we will discuss the existing approaches as available for single-document summarization, multi-document summarization and methodologies we have adopted for the summarization – bootstrapping and spreading activation.

7.2.1 Single-Document Summarization

Although the traditional non-graph based approaches are successful in locating salient text units in documents, research based on the graph theory can help people to better understand the relations of the text units (i.e. sentences). Our primary objective of this study is to generate a summary from semantic graphs, using a bootstrapping approach where spreading activation takes place for the selection of significant components (nodes and links). In this section, we investigate the existing approaches related to graph based summarization, bootstrapping and spreading activation.
Hendrickx & Bosma (2008) extended an approach for graph based summarization, where coreference relations have been utilized for sentence compression. The relevancy and redundancy are expressed in the links connected with the content units represented as nodes. The sentence compression module outputs full sentences, that are simply added to the summary without validating the sentences (Hendrickx & Bosma 2008). Erkan & Radev (2004) presented an extractive summarization procedure, where the salient measures have been determined using the graph-based centrality measures of sentences. Another approach presented by Leskovec et al (2004) performs a deep syntactic analysis to extract the subject-predicate-object triplets. The pronouns and coreference links are connected and the triplets are normalized to form a complete semantic graph. However, while the coreference links are considered for the construction of fully connected semantic graphs, the author extracts only the triplets for the summary.

### 7.2.2 Multi-Document Summarization

In this section, we will discuss the various approaches to query based multi-document summarization, where we will discuss the redundancy elimination, and the selection of sentences for the summary. Most extractive summarization approaches normally extract highly ranked sentences from the documents, in which the sentences may contain redundant information. In many approaches, sentences are scored using a statistical approach, and then the redundancy is avoided by ensuring that the cosine similarity of selected sentences to already included sentences is below a learned threshold (Darling & Song 2011).

ALJa’am et al (2009) proposed a new approach to the text summarization using conceptual data classification. The most interacting sentences are extracted to generate the summary and it doesn’t use any semantic or grammatical concepts. A polynomial approximate algorithm is
then applied to find the minimal number of optimal concepts for summary generation (ALJa’am et al 2009).

An unsupervised approach has been attempted for the query based summary, where the EM algorithm is used for clustering the documents using various lexical, lexico-semantic and cosine similarity features, and ranking the sentences using the K-means algorithm to generate summaries (Chali & Joty 2008).

A query based multi-document summary has been generated by extracting the frequent items from a clustering of a document set based on the weighted tfidf. The irrelevant documents for a query are filtered out and the important sentences are extracted from the selected documents for the summary (Rosner & Camilleri 2008). MEAD, proposed by Radev et al (2004) is a centroid based multi-document summarizer, which generates summaries using cluster centroids produced by the topic detection and tracking system.

A probabilistic approach has been proposed for the query-oriented topics-based summarization task, in which two strategies have been presented for simultaneously modeling the query and the document cluster. A query specific generative process has been introduced, to integrate the query information with the topic model and the topic distribution has been regularized to avoid the “too many parameter tuning” problem (Tang et al 2009).

The existing approaches on query focused multi-document summarization, cluster the documents based on cosine similarity which basically uses term based similarity, and thus do not obtain semantically similar documents for a summary. Moreover, extractive summaries use scoring functions for the selection of sentences based on the n-gram subsequence. However, the existing graph based approaches resolve the
issues in statistical approaches, the sentences are represented as text units (nodes), and are connected by the relations represented as edges.

A document graph based query focused multi-document summarization system has been described by Paladhi & Bandyopadhyay (2008), in which the basic unit of clustering is a paragraph, and the minimum spanning tree over the document graph is identified as the summary. A similar clustering approach has been considered by Bhaskar & Bandyopadhyay (2010) where the sentences have been considered as the basic clustering unit. The top ranked sentences have been parsed and the sentences are compressed by removing the unimportant or irrelevant phrases of the sentence.

Another approach to summarization uses a directed document semantic graph, where entities/concepts are represented as nodes and relations between entities/concepts, such as “is-a” and “related_to” are represented as edges. While entities/concepts have been extracted from the parsed sentences, the relations among them have been identified, using a set of heuristics (Mohamed & Sanguthevar 2006). However, while the document centric graphs are merged to form a global graph, the handling of redundant subgraphs has not been tackled, and this results in the summary containing repetitive information.

Yeh et al (2008) proposed a graph based ranking method for extracting the important sentences, using weighted sentence specific scores, where spreading the activation has been performed on the weighted sentences over the entire network. However, the spreading activation uses only node information (i.e. weighted score of the sentences) for traversal and selection of sentences; the summary may contain unimportant sentences, and the sentences may also contain anaphoric concepts which may have an impact on sentence scoring; but all these have not been discussed.
Though the existing graph based approaches work well in extracting the important sentences for a summary, the redundant sentences to avoid the repetitive information in the extracted summary have not been tackled. Furthermore, domain-specific semantic relations, and relevancy and redundancy relations have been utilized to identify the link between the sentences. However, the semantic relationships between the concepts and/or sentences have not been captured, which results in a non-cohesive and sometimes, irrelevant summary. In the proposed approach, we make use of sentence semantic graphs with various anaphora tackled, in which common concepts across the sentences and/or documents are then connected to form a global semantic graph. While building the global semantic graph, the redundancies are eliminated, using the graph matching procedure, where the semantic properties of the nodes and edges are used.

The selection of prominent components is carried out using a query graph integrated spreading activation theory to obtain the query relevant summary graph, which we will discuss in section 7.5. The direction of traversal is decided based on the nodes and edges along with the query graph, which narrow down the selection of components relevant to the query.

The extractive methods discussed above, focused on extracting important sentences, without examining the relevance ratio of the sentences with respect to the summary. Moreover, extractive summaries have the possibility of obtaining irrelevant sentences, that do not have accurate information. The salience and centrality measures that have been used for important sentence extraction, only extract triplets where complex and compound sentences which are important in the summary are ignored.

In contrast, an abstractive summarization technique is proposed to overcome the difficulties in extractive methods. In our approach, instead of creating the graphs of triplets, we construct a semantic graph which tackles
complex structures of sentences, which can then be indicated using the identifiers. In this study, we propose a semi-supervised bootstrapping approach for the identification of important components of a fully connected semantic graph for abstractive summarization. The bootstrapping approach is applied while performing a spreading activation process in a fully connected directed semantic graph.

7.2.3 Spreading Activation

Spreading Activation (Anderson 1983) is a graph based algorithm which initially starts with a node with weights in the semantic network, and iteratively spreading that activation out to other nodes connected to the source node in the semantic network (Crestani 1997). In general, Spreading activation algorithms iteratively propagate the activation from the initial set of nodes referred to as the seed, to the other nodes in a network through outward links (Trousov et al 2009).

Spreading activation has been utilized for various NLP tasks; however, we restrict the discussion to summarization alone. Spreading activation has been performed for summarization (Nastase 2008) to select the important nodes for the summary. Thiel & Berthold (2012) proposed two methods for the identification of structural and spatial node similarities through spreading activation. One is the overlap of direct and indirect neighbors, and another is the comparison of distant neighborhood. Since the authors used the cosine similarity measure over directed graphs, they have focused only on node information and not on the link or edge information. The spreading activation algorithm activates the spreading by computing the semantic relatedness measure, based on the properties of the terms in the query, the type of rdfs relation associated with instances, and the instances associated in the linked data web. The adaptive threshold has been determined
for the activation function to explore the nodes in the linked data (Freitas et al 2011).

The next section describes the basic definitions and notations that we use throughout this chapter.

7.3 DEFINITIONS AND NOTATIONS

In this section, we describe some basic definitions and notations of standard graphs. Graph isomorphism is generally used for the matching and merging of subgraphs of two graphs. The same procedure is utilized in the proposed approach for removing the redundant subgraphs and building the global semantic graph.

**Labeled Graph:** A labeled graph can be represented as G (V, E, LV, LE, φ), where V is a set of vertexes, E ⊆ V × V is a set of edges; LV and LE are sets of vertex and edge labels respectively; and φ is a label function that defines the mappings V → LV and E → LE.

**Subgraph:** Given two graphs G1(V1, E1, LV1, LE1, φ1) and G2(V2, E2, LV2, LE2, φ2), G1 is a subgraph of G2, if G1 satisfies: (i) V1 ⊆ V2, and ∀v ∈ V1, φ1(v) = φ2(v), (ii) E1 ⊆ E2, and ∀(u, v) ∈ E1, φ1(u, v) = φ2(u, v). G1 is an induced subgraph of G2, if G1 further satisfies: ∀u, v ∈ V1, (u, v) ∈ E1 ⇒ (u, v) ∈ E2, in addition to the above conditions. G2 is also a supergraph of G1 (Inokuchi et al 2002; Huan et al. 2003).

**Graph Isomorphism:** A graph G1(V1, E1, LV1, LE1, φ1) is isomorphic to another graph G2(V2, E2, LV2, LE2, φ2), if and only if a bijection f : V1 → V2 exists such that: (i) ∀u ∈ V1, φ1(u) = φ2(f(u)), (ii) ∀(u, v) ∈ E1 ⇔ (f(u), f(v)) ∈ E2, (iii) ∀(u, v) ∈ E1, φ1(u, v) = φ2(f(u), f(v)). The bijection f is an isomorphism between G1 and G2. A graph G1 is a subgraph isomorphic to a
graph $G_2$, if and only if there exists a subgraph $g \subseteq G_2$ such that $G_1$ is isomorphic to $g$ (Huan et al 2003). In this case $g$ is called an embedding of $G_1$ in $G_2$.

**Query Graph ($G_Q$):** A query $Q$ consists of multiple words, denoted as a semantic graph where concepts are represented as nodes, and the relations between the concepts are represented as edges.

**Sentence Semantic Graph ($G_{SENT}$):** Initially, the semantic graph is constructed for each sentence in a document, where concepts and the relations between the concepts in a sentence are represented as a graph.

**Document Semantic Graph ($G_{DOC}$):** The coreference and synonyms concepts (nodes) of the sentence semantic graphs ($G_{SENT}$) are connected to form a fully connected document semantic graph.

**Global Semantic Graph ($G_{GLOBAL}$):** The common nodes and/or subgraphs of each document semantic graph ($G_{DOC}$) are connected, to form a global semantic graph. The elimination of redundant information is discussed in section 7.5.2.1., which can occur while integrating semantic graphs of multiple documents.

### 7.4 SINGLE DOCUMENT SUMMARIZATION

In this section, we describe the single document abstractive summarization task in which the learning of graph operations and the identification of important components using the modified spreading activation procedure are performed. The input to this task is a fully connected document semantic graph ($G_{DOC}$). First, the important components of $G_{DOC}$ are identified using a single stage pattern based bootstrapping approach described in section 3.6.1. This bootstrapping approach learns the possible
graph operations that can be assigned to the nodes and edges of the document semantic graph (G_DOC). Then, the modified spreading activation procedure (the edge information is integrated while the original spreading activation starts the process with only node information) starts the activation of spreading on the flags assigned nodes and edges, to obtain the compressed summary semantic graph. The following sections describe the bootstrapping for learning graph operations and edge integrated spreading activation procedure for single document summarization task.

7.4.1 Bootstrapping for Learning Graph Operations for Summarization

In this work, we use bootstrapping for learning the graph operations to be performed on the document semantic graph in order to obtain the summary semantic graph. In the bootstrapping process, we use pattern specification which is important in determining the various graph operations. In order to design this pattern for the purpose of summarization, we need to identify the features that decide which components of the semantic graph are to be selected or modified for building the summary.

Before we explain the bootstrapping procedure, we will explain the graph operations we learn using this procedure, and the context (or features) which decides the learning.

7.4.1.1 Graph operations and summarization

The input to our bootstrapping approach is a fully connected semantic graph, and therefore, we use the information available in the graph as features for representing the patterns. We have introduced a new feature in the form of a graph operation flag, which determines what type of graph operation we need to execute on the nodes/edges of the original document
semantic graph. The graph operations include deleting nodes/edges or merging nodes/edges, or in some cases, replacing nodes/edges or even creating new nodes/edges. The bootstrapping algorithm learns the value of this flag (delete, insert, replace or modify) depending on the context of the node/edge in the semantic graph. In the next section, we will discuss the heuristics that decide, how important components are selected through appropriate graph operations, for building the summary. Normally, extractive summarization involves the selection of important sentences; in other words the deletion of unimportant sentences. In the proposed work, this translates to the deletion of unimportant components of the semantic graph. However, since we are dealing with the construction of an abstractive summary, additional operations such as modification, creation and merging of nodes/edges are also carried out. In the reduction of lengthy sentences into a short form without losing its complete meaning, certain components of the semantic graph need modification (i.e. sentences of the summary could be different from their original form); tackling of the dangling nodes in the semantic graph involves the creation of new nodes to connect the dangling nodes, and merging of nodes is performed to obtain a compressed form of summary. Based on the properties discussed above, we present a set of heuristics for each graph operation.

7.4.1.1.1 Deletion of nodes/edges

Certain components are not important for a summary and thus can be eliminated. The non-essential parts of the semantic graph are removed based on the semantic relations existing between the nodes. Martin & Rino (2002) defined that the non-essential components of the sentences are signaled by the semantic relations “mod” and “aof”. Similarly, we define a set of heuristics, where the semantic relations are utilized in the removal of non-essential parts of semantic graph.
The deletion operation can be divided to two:- deletion of a node and deletion of an edge.

The heuristics used during the deletion of a node are:

- Deletion of a node $C_i$ independently is based on its POS tag and the frequency of the occurrence of $C_i$ in the document
- Deletion of a node $C_i$ along with the associated edge $R$ is based on the relation labels connecting $C_i$ with the other nodes.

These deletions may lead to several difficulties.

1. If the deletion operation is performed on a leaf node $C_i$ with edge $R$; then,
   - if there is no further graph operation associated with edge $R$
     - edge $R$ is deleted.
   - Else
     - The edge is considered to be a dangling edge and may be used to connect to another node in the graph, which is semantically similar to the deleted node $C_i$ through further graph operations.

2. else
   - the edge may be used to connect to another node in the graph, which is semantically similar to the deleted node $C_i$ through further graph operations.
3. The deletion of the node $C_i$ may result in a dangling edge $R$, that is, the edge $R$ is connected to a single node $C_j$. Now $R$ has to connect $C_j$ to another new node $C_k$ which is semantically similar to $C_i$. But, it may be necessary to change the relation label of edge $R$ connecting $C_j$ and $C_k$.

### 7.4.1.1.2 Insertion of nodes/sub-graphs

The introduction of a new node is carried out, to overcome the dangling node problem, existing in the original semantic graph or as the consequence of a deletion operation. For instance, the terms in a document can occur in a list format, where relations do not occur between them. Such concepts create dangling nodes in the original semantic graph. These dangling nodes may be important for summarization. The dangling nodes are handled by connecting semantically similar concepts with a newly created common node, based on a common upper level concept obtained through semantic abstraction from the UNL Ontology. While inserting a new node $C_{new}$, it is necessary to introduce an appropriate edge $R_{new}$ between the newly inserted node and the dangling nodes. The edges are labeled with appropriate semantic relations, based on the semantic constraints assigned to the newly inserted node and the semantic constraints of the dangling nodes.

### 7.4.1.1.3 Merging the components of a graph

To reduce sentential redundancy, Sornlertlamvanich (2001) has proposed merging nodes with the same UW. In this work, the merging operation replaces edges that connect nodes with different UWs but labeled with the same semantic relation by a single edge, connecting to a subgraph, formed by grouping nodes with edges labeled with the “conjunctive” semantic relation.
Let SG be the semantic graph, and $C_i$ and $C_j$ a pair of its components which forms a relation $R$. The merging operation is performed based on the following assumptions

**Level1:** Node $C_i$ is connected to the nodes $C_j$ and $C_k$ with the same relation $R$; then the nodes $C_j$ and $C_k$ are grouped and connected with the “and” relation, and node $C_i$ is connected to the group $(C_j, C_k)$ with the same relation $R$.

**Level2:** Node $C_i$ is connected to the node $C_a$ with the relation $R$; node $C_j$ is connected to node $C_b$ with relation $R$, and if the nodes $C_i$ and $C_j$ are synonyms, then keep any one node (based on the frequency of occurrence or the strength of the node) $C_i$ or $C_j$ and apply level1.

**Level3:** Nodes $C_i$ and $C_j$ are synonyms and are connected to the nodes $C_k$ and $C_l$ with the relations R1 and R2 respectively. If the relations R1 and R2 are dependent relations, then apply level1 and keep the more appropriate relation.

Section 7.4.2 explains the features that describe the context which define the above graph operations that need to be performed.

### 7.4.2 Identification of Features for Describing Context

While linguistically analyzing the Tamil text, certain features associated with the context are involved in the selection of important nodes for summarization. Feature selection plays an important role in the specification of patterns used in the identification of important components, from the document semantic graph through appropriate graph operations, necessary for summarization. In the proposed work, these important components are selected, based on the semantic properties, and therefore, we need to identify features of the semantic graph representation, and other contextual and statistical features that decide the graph operations needed to build the summary. The features are classified, based on whether they are associated with the nodes or edges of the semantic graph.
Features associated with the nodes include word based features, such as POS tags, lexical features and semantic constraints associated with a word/ concept, concept based features including the frequency of concepts, and type of relation between an edge and a particular concept. Edge based features consider the relation associated with an edge. In essence the use of the word based semantic constraints and the edge based semantic relations results in a semantic based rather than a sentence or text unit based graph summarization.

In addition to the node and edge based features, we also have features to represent the boundaries of semantic sub-graphs, and sentence identifiers. In our work, features are used to create a summary semantic graph from the original document semantic graph.

**Table 7.1 Features for Learning Graph Operations**

<table>
<thead>
<tr>
<th>Features associated with nodes</th>
<th>Word based features</th>
</tr>
</thead>
<tbody>
<tr>
<td>POS tag of a word (POS)</td>
<td>Part of Speech information</td>
</tr>
<tr>
<td>Semantic constraint associated with a word (SC)</td>
<td>Semantic class feature to obtain the synonyms of the concepts</td>
</tr>
<tr>
<td>Attributes associated with a word (Attr)</td>
<td>Used to represent number, gender, aspect, mood etc.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Classical Features</th>
<th>Statistical Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concept identifier</td>
<td>Frequency of a concept</td>
</tr>
<tr>
<td>Multi-word Expression Identifier</td>
<td>Frequency of a concept in a document</td>
</tr>
<tr>
<td></td>
<td>Scoring function for the selection of nodes</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Graph based features</th>
<th>Context based features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Semantic relations associated with a word (Relation)</td>
<td>Semantic relations of interest while performing spreading activation</td>
</tr>
<tr>
<td>Nested graph identifier (NG_id)</td>
<td>Identifier to detect the boundary of the sub-graphs</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Flag features</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Flags {Create [CF], Delete [DF], Replace [RF], Merge [MF]}</td>
<td>Graph operation to be performed on the nodes and/or links in the graph</td>
</tr>
</tbody>
</table>
7.4.3 Bootstrapping for Learning Graph Operations

The first step in this single stage pattern based bootstrapping procedure, is to represent the tuples of the pattern, which describes the context of the semantic graph. During the learning procedure, the patterns with graph operation flags as labels are utilized, to assign the flags in a fully connected semantic graph through matching. The documents of the test corpus have already been tagged with the features identified, except for the graph operation flag which is set as a result of the matching procedure. These labeled examples (where the graph operation flag is set) are then used for the tuple value confidence scoring scheme, to decide which tuple of the pattern is to be masked for partial matching. Again the similarity scoring scheme is also applied to generate new patterns based on partial matching, and the iterations are continued until all the documents have been labeled, or no new patterns are generated. This bootstrapping procedure is similar to the approach proposed in Section 3.6.1 for learning relation labeling, except that in the previous work, learning of relation labels is performed. In this work, we utilize the procedure to set the graph operation flags associated with nodes/edges of the semantic graph. In the following sections, we describe the pattern representation and the scoring methodology used during the bootstrapping procedure.

7.4.3.1 Pattern Representation

The next step in the bootstrapping procedure is the representation of the pattern using the features identified. In this work, we consider <concept node – relation – concept node> as the basic sub-component of the semantic graph, and use this as a basis for the representation of the patterns. The pattern is represented using the context based and classical features; the pattern for learning is defined as follows
<Component 1 – Relation – Component 2>\textit{NG\_id+S\_id}

where components 1 & 2 consist of features, such as the morphological suffix (MS), POS tag (POS), Semantic constraint (SC), Multi-Word tag (MW) and the Flag (\textit{Flag}) representing which operation to perform. These features are tagged for each component in the pattern as \textit{MS+POS+SC+MW+Flag}. In addition, the pattern may also consider the overall graph features, such as sentence identifiers and/or sub-graph identifiers. The features used are given in Table 7.1. The detailed pattern representation is given below.

\textit{<MS+POS+SC+MW+Flag - Relation+Flag - MS+POS+SC+MW+Flag>NG\_id+S\_id}

The tuple \textit{MW} and \textit{NG\_id} defined in the pattern are optional, while the rest of the tuples are necessary features of the pattern. If the component in the pattern is a multi-word expression, the MW flag is set, and likewise, only if the semantic graph consists of the nested subgraph, which essentially are sub-graphs that act as a node, then the \textit{NG\_id} is used. Based on the flag, the operations are performed during the spreading activation. The rest of the features listed in Table 7.1 are used during the selection of nodes and subgraphs.

7.4.3.2 Matching

In the bootstrapping procedure, matching is carried out through exact matching and partial matching. Here, we make use of two probabilistic scoring schemes (described in section 3.6.1.2 for the semantic relation extraction task) for the selection of patterns for modification, and the tuples to be considered for masking.
7.4.3.2.1 Scoring

The selection of the pattern for modification and the selection of the tuples for masking are carried out, using probabilistic scoring schemes. In the proposed approach, two scoring functions are introduced.

1. Scoring for pattern selection for modification (described in section 3.6.1.3)
2. Scoring for tuple selection for masking

Scoring - Tuple Value Selection for Masking

The pattern for modification is selected based on the pattern selection score discussed above. In order to perform a partial match, the tuples are masked in an iterative manner, to obtain partially matched instances that are then modified to generate new patterns. We use tuple value selection scores, to decide which set of instances would yield patterns for modification. In our approach, the tuple value selection score is determined by the probability of the number of samples, with the Tuple Value (TV) of all the patterns of a particular Flag (F) among all samples with the same TV. The tuple value selection score is given in Equation (7.1).

\[
Select(TV_i) = \arg \max_{i=1}^{m} \sum_{j=1}^{n} p([TV_i/P_j,F]/TV) \tag{7.1}
\]

The scoring for tuple values selection is based on the occurrence of this tuple value in patterns for setting a particular flag. While the pattern selection score is used to select the pattern for masking, the tuple value selection score selects specific tuple values, which can be abstracted to obtain more generic patterns.
7.4.3.3 Generation of New Patterns

The node, which is an entity with high frequency in the document, is selected as a starting node to start the bootstrapping process. If the selected node ($N_{\text{START}}$) is a location (which is determined by the semantic constraint associated with $N_{\text{START}}$), then the bootstrapping process starts applying the graph operations based on the following initial conditions.

1. The starting node $N_{\text{START}}$ is connected to another node $N_i$ by the semantic relations representing locations ($R_{\text{LOC}}$) such as $plc$, $plf$, $plt$, and then the nodes ($N_j$) connecting $N_{\text{START}}$ either directly or indirectly are marked with the selection flag.

$$R_{\text{LOC}}(N_{\text{START}}, N_i) \land R_{\text{MISC}}(N_i, N_j)$$

$$\implies \text{Set Flag}_{SELECT}(N_{\text{START}}, R_{\text{LOC}}, N_i, R_{\text{MISC}}, N_j)$$

where $i = 1, j > 1$

2. The starting node $N_{\text{START}}$ is connected to another node $N_i$ by the semantic relations representing participants ($R_{\text{OBJ}}$), and then the node ($N_i$) is connected with node ($N_i$) by another participant relation ($R_{\text{AGT}}$); then the nodes and edges ($N_{\text{START}}, R_{\text{OBJ}}, N_i, R_{\text{AGT}}, N_j$) are marked with the selection flag.

$$R_{\text{OBJ}}(N_{\text{START}}, N_i) \land R_{\text{AGT}}(N_i, N_j)$$

$$\implies \text{Set Flag}_{SELECT}(N_{\text{START}}, R_{\text{OBJ}}, N_i, R_{\text{AGT}}, N_j)$$

where $i = 1, j > 1$

3. The starting node $N_{\text{START}}$ is connected to another node $N_i$ by the semantic relations representing locations ($R_{\text{LOC}}$) such as
\( plc, plf, \) and \( plt, \) and then the node \( (N_j) \) are connected with \( N_i \) by semantic relations representing time \( (R_{TIME}) \) such as \( tim, tmf, \) and \( tmt; \) then, the nodes and edges \( (N_{START}, R_{LOC}, N_i, R_{TIME}, N_j) \) are marked with the selection flag.

\[
R_{LOC} (N_{START}, N_i) \land R_{TIME} (N_i, N_j) \\
\implies \text{Set Flag}_{SELECT} (N_{START}, R_{LOC}, N_i, R_{TIME}, N_j)
\]

\( where \ i = 1, j > 1 \)

4. The starting node \( N_{START} \) is connected to another node \( N_i \) by the semantic relations representing locations \( (R_{LOC}) \) such as \( plc, plf, plt; \) then the nodes \( (N_j) \) connected with \( N_i \) by semantic relations representing quantity, and the nodes and edges \( (N_{START}, R_{LOC}, N_i, R_{QUA}, N_j) \) are marked with the selection flag.

\[
R_{LOC} (N_{START}, N_i) \land R_{QUA} (N_i, N_j) \\
\implies \text{Set Flag}_{SELECT} (N_{START}, R_{LOC}, N_i, R_{QUA}, N_j)
\]

\( where \ i = 1, j > 1 \)

If the selected node is a person, then the nodes connected with the participant relations and attributive relations are marked for the selection of the important components. Certain subgraphs are important, where the condition does not depend on the specific property of the starting node.

The node \( N_i \) is connected to a set of nodes \( N_j \) by the attributive semantic relations \( (R_{ATTR}) \) such as \( iof \) and \( nam; \) then the nodes \( (N_j) \) connected with \( N_i \) \( (N_i, R_{ATTR}, N_j) \) are marked with the selection flag.
\[ R_{ATTR} (N_i, N_j) \implies \text{Set Flag}_{SELECT} (N_i, R_{ATTR}, N_j) \quad \text{where } i = 1, j > 1 \quad (7.6) \]

Semantic relations such as *man*, *mod*, and *ben* imply modifier relationships, and removing the nodes having modifier relations would not affect the meaning of the sentence. Therefore, the nodes and edges satisfying these properties are marked for deletion.

The creation of new nodes and edges is performed for the dangling nodes \((N_{D,i})\) which are semantically similar. With the use of the UNL Ontology, the semantic abstraction of the semantically similar concepts is identified and the new node is created. The new node \((N_{NEW})\) is then connected to the dangling nodes having the same or similar semantic constraints.

### 7.4.4 Spreading Activation for Graph Summarization

The existing Spreading Activation search algorithm described by Suchal (2008) has been modified to execute the graph operation flags. The modified spreading activation algorithm is described below.

Modified Spreading activation performs as follows:

- The activation of the starting node is based on the frequency of occurrence and POS information associated with the node.

- The activation of spreading is performed iteratively by considering the strength of the node, which includes the frequency of occurrence, POS tag and the semantic relations (links) connecting the main node (starting node) either directly or indirectly.
• Instead of measuring the distance between the starting node and the current node, the similarity between the starting node and the current node is measured in terms of constraint similarity, attribute similarity and edge similarity.

• Edges labeled with relations are used for the selection of nodes in a graph.

• Instead of checking the termination, we traverse the graph fully for selecting the important nodes.

• Subgraphs are also selected as important components through the nested graph identifier. The nested graph identifier is useful in the identification of the boundary of the components of a graph, so that sub-graphs can be selected easily.

• Graph operations are carried out during the spreading activation of the semantic graphs for the selection of the important nodes/sub-graphs.

The algorithm given below shows the modified spreading activation procedure for the traversal of nodes.

**Algorithm:** Spread-Activation (v, e, c≤0, r)

Require: Starting vertex v.

Require: Activation energy e > 0.

Require: Energy c accumulated on graph vertices.

Require: Relation r accumulated on graph edges.

Require: Weight w assigned on graph edges
1. \( c_v \leftarrow c_v + e + r_w \)
2. \( e' \leftarrow e / \text{Vertex-degree (v)} \)
3. if \( e' > \theta \) then
4. for all vertices \( t \) such as, there exists an edge from \( v \) to \( t \) (represented as a pattern) do
5. \( c \leftarrow \text{Spread-activation (v, e', c, w)} \)
6. end for
7. end if
8. return \( c \)

**Input:** Completely connected semantic graph which has been tagged. The graph operation flags of the nodes and edges of this tagged document graph have been set, using the bootstrapping learning procedure.

The first step in the spreading activation procedure, is deciding from which node to start. This is done by the energy of the node \( C_i \) (concept) or the concept with the maximum frequency of occurrence in the document semantic graph. The graph operation associated with node \( C_i \) is first carried out. The details of the graph operations are described in section 7.4.1.1. The direction of the spreading activation, in other words, the edge to follow, has to be decided. The constraint based spread activation model described by Crestani (1997), discussed the distance constraints where the relation strength had been measured based on the relation existing between two nodes, either directly or indirectly. In the same vein, in our approach, we use semantic relations connecting nodes whose strengths are measured based on whether the relations are coordinating or subordinating relations (described in
section 6.7.2.1.1). The process is continued till the spreading activation cannot continue any further.

The compressed summary semantic graph is obtained using the modified edge integrated spreading activation algorithm from document semantic graph. The important components of the document semantic graph are assigned a set of flags using a pattern based bootstrapping approach. The modified spreading activation procedure is then applied over the flags assigned document semantic graph widely to extract the important components such as nodes and edges. The next section describes the query focused multi-document summarization task where the query graph is integrated with the edge integrated spreading activation algorithm. The redundancy elimination procedure is carried out to avoid duplications and the query graph integrated spreading activation procedure is applied to obtain the query relevant summary. To obtain the query oriented summary graph, the spreading activation narrow down the search of important components in the global semantic graph with respect to the user query.

7.5 QUERY FOCUSED MULTI-DOCUMENT SUMMARIZATION

Multi-document summarization is a task of identifying the important common themes and/or aspects of multiple documents. A multi-document summary has several advantages over the single-document summary. It provides an integrated overview of a document set, indicating common information across many documents, unique information in each document, and cross-document relationships, and can allow users to zoom in on details of different aspects of interest. One of the main tasks in multi-document summarization is the identification of similarities and differences between documents (Wan & Yang 2008).
One of the challenges of multi-document summarization is that a set of documents might contain diverse information, which is either related or unrelated to the particular topic. Therefore, effective methods are needed to analyze the information stored in different documents, and abstract the globally important information to reflect the main topic. While in single-document summarization, the sentences conveyed in a document are unique and may not have redundant information that affect the quality of the summary, in the case of multi-document summarization an important challenge is that, the information contained in different documents inevitably overlaps, and hence, we need effective methods to merge and reproduce information with minimum redundancy. Moreover, multiple documents from a News corpus, for example, can convey the same event with different sentence structures, without conveying additional information. While summarizing the text from such multiple documents, the redundant information needs to be eliminated to produce a generic distinctive summary. Redundancy of information is especially an issue, when we use ranked documents obtained from a search engine, using a search query to produce a query focused summary.

Most existing summarization approaches focus on an extractive summary, where the important sentences are extracted using the salient information from different documents. They use a shallow analysis, without paying much attention to the rich semantic features associated with words, and semantic relations expressed within and across sentences as well as documents. Yet another problem is that different users have different information needs. Thus, an ideal multi-document summarization should provide different levels of details for a specific topic, according to the user’s interest (Chali & Joty 2008). This can be achieved by exploiting the lexical, syntactic, semantic, pragmatic and discourse information of multiple documents.
One promising approach to multi-document summarization is to outline the overall structure of a set of related documents to give users an overview of a specific topic, and subsequently allow them to zoom into different areas according to their interest. This is performed by semantically analyzing multiple documents rather than a single document (described in section 7.4), using the semantic relations between concepts in the sentences, between sentences, etc. Multi-document summarization is also carried out using graph-based approaches, such as LexRank (Erkan & Radev 2004). LexRank has been applied a random walk to a fully connected undirected graph, to redistribute the node weights where text units (i.e. sentences) are represented as nodes, and similarities between text units are represented as edges. However these approaches result in redundant information in the summaries.

In fixed-word order languages like English, the roles are conveyed by prepositions, and syntactic parsing is needed to preserve the structure and meaning of the sentence. Therefore, parsing is a difficult process; we used the morpho-semantic features of a morphologically rich language to identify the semantic relations. In general, coreference information is helpful for the task of automatic summarization (Baldwin & Morton 1998; Steinberger et al 2007). Important entities in a text are mentioned multiple times and are spread over the text. Tracking these entities results in identifying what the text is about. Furthermore, the presence of an important entity in a particular sentence may indicate that this might be an important sentence that should be extracted. From a linguistic viewpoint, coreference information can help to maintain the coherence and readability in a text (Bergler et al 2003; Witte et al 2006). Instead of extracting the important sentences based on the occurrence of entities, the proposed approach uses the coreference information for narrowing down the searching and tracking of the important components of the global semantic graph.
In this work, we focus on two aspects of multi-document summarization – redundancy elimination and query focusing. The graph matching redundancy elimination approach that we have used, is based on the structural and semantic properties of the semantic graph, representing the set of documents to be summarized. While focusing on a query oriented summary from multiple documents, relevance checking needs to be performed, using various similarities scoring measures, such as concept similarity (synonym, hyponym and hypernym concepts) and graph similarity (Concept-Relation-Concept (CRC) and semantic relation), between the query and the sentences, to narrow down the process of obtaining a query relevant summary. The spreading activation algorithm is modified to decide the direction of traversal among the nodes and edges of the semantic graph, based on the given query. In this work, in addition to the edge information described in section 7.4.4., the query graph is also incorporated during the selection of the important components of the global graph.

7.5.1 Graph Compression

The compression of a text is an important part of summarization. The summarization compression ratio is the ratio of the size of the compressed data to the size of the source data. The compression method may apply at the level of sentences, words or characters. Graph compression of the semantic representation of a single document is performed, while building a summary semantic graph by learning the graph operations to be executed on nodes and edges during the spreading activation. In this section, we focus on multi-document summarization, where the global multi-document semantic graph is subjected to a first level of compression through redundancy elimination. A query focused modified spreading activation technique is then used to obtain the query focused multi-document summary. Here, in addition to the query graph, we assume that the semi-supervised learning algorithm
described in our earlier work, is used to flag the graph operations which is to be carried out at each node and edge of the global semantic graph during the spreading activation. A detailed design diagram with the contributions highlighted is shown in Figure 7.1. The tasks shown in Figure 7.1 are explained in the following sections.

![Semantic Graph Construction Diagram](image)

**Figure 7.1 Detailed Design Diagram for Query Focused Multi-Document Summarization**

The UNL - language independent representation has been utilized for the construction of the sentence semantic graphs of each document which are then connected to form a global semantic graph of multiple documents.

### 7.5.2 Building Global Semantic Graph

The input to our multi-document summarization is a set of semantic graphs initially constructed for the sentences within a document. The common nodes across sentences and documents with high frequencies are connected to form a complete semantic graph of multiple documents. While grouping the semantic graphs of multiple documents, the repetition of the same node/subgraphs may lead to redundant information.
7.5.2.1 Graph compression through redundancy elimination

Redundant subgraphs are eliminated in two stages. During the first stage, while building a fully connected semantic graph of multiple documents, common nodes/subgraphs across documents that overlap with other are merged to avoid redundancy. During stage 2, after the construction of the global graph, the subgraphs of this global graph are assigned a unique label called the canonical label, and the redundant subgraphs having identical canonical labels are eliminated.

Stage 1: Elimination of overlapping nodes/subgraphs

To eliminate the redundant information while grouping multiple document semantic graphs, the following actions take place.

1. If node A is connected to node B with edge R1 and node C with edge R2 in a document semantic graph $SG_{DOC}$, and the node A is connected to node C with edge R2 and node D with edge R5 in another document semantic graph $SG'_{DOC}$, then the subgraph A-R2-C of $SG'_{DOC}$ is merged with the subgraph of $SG_{DOC}$ and the other nodes connected with nodes A and C of $SG'_{DOC}$ are merged with the nodes A and C of $SG_{DOC}$.

![Graph Diagram](image_url)

**Figure 7.2** (a) subgraph of Document Semantic Graph $SG_{DOC}$
(b) Subgraph of Document Semantic Graph $SG'_{DOC}$
(c) Merged Subgraph of $SG_{DOC}$ and $SG'_{DOC}$
2. If node A is connected to node C with edge R1 in a document semantic graph $SG_{DOC}$, and the node A’ (which is a synonym concept of node A) is connected to node C with edge R1 in another document semantic graph $SG’_{DOC}$, then the subgraph R1-C alone of $SG’_{DOC}$ is merged with the subgraph of $SG_{DOC}$, and node A’ of $SG’_{DOC}$ is connected to node A of $SG_{DOC}$ with “equ” semantic relation, only when the nodes A and A’ are non-verbal concepts. The semantic relation “equ” shows that A and A’ are semantically equivalent concepts.

![Diagram of subgraphs](image)

**Figure 7.3**

(a) Subgraph of Document Semantic Graph $SG_{DOC}$
(b) Subgraph of Document Semantic Graph $SG’_{DOC}$
(c) Merged Subgraph of $SG_{DOC}$ and $SG’_{DOC}$ with Synonym Concept Connected

Using the above two conditions, the semantic graphs of multiple documents are grouped together to form a global semantic graph. The fully connected complete semantic graph across documents is shown in Figure 7.4. As already mentioned above, the nodes in the semantic graph contain the word-based features and the edges contain the semantic relations that exist between the nodes. While merging the nodes/subgraphs, various parameters such as document identifiers, term frequency and concept frequency in each document, and sentence identifiers are utilized, to know in which sentences in a document the term/concept occurs. Initially, documents are selected for summarization based on term and concept frequencies, and the semantic
relations connecting the concepts under consideration. The document identifiers are used to determine which documents are closely relevant, and the sentence identifiers are utilized to maintain the sentence ordering at the time of summary generation.

**Figure 7.4 Sample Complete Fully Connected Semantic Graph of Multiple Documents**

The above two conditions help in identifying and eliminating the redundant subgraphs; however, not all semantically similar subgraphs are captured. For instance, sentences can be represented in the active voice as well as in passive voice. In this case, the sentence structures are different though the meanings conveyed by the sentences are the same. Such redundancies are detected during stage 2, which is performed after the construction of the fully connected semantic graph across documents.
**Stage 2: Labeling of redundant subgraphs**

The elimination of redundant subgraphs requires an operation to check whether two subgraphs are identical or not. One such operation is to perform the graph isomorphism. However, in certain cases where many such checks are required among the same set of subgraphs, a better way of performing this task is to assign a unique label to each subgraph, that is invariant on the ordering of the vertices and edges in the graph. Such a procedure is referred to as the canonical label of the graph $G = (V, E)$ and is denoted by $\text{cl}(G)$ (Kuramochi & Karypis 2005). Using canonical labels, the subgraphs having identical canonical labels are identified as redundant subgraphs.

In practice, the complexity of finding a canonical labeling of a graph can be reduced using various heuristics to narrow down the search space. As part of our summarization process, we have used the patterns (used for learning graph operations) that make use of the edge and node labels, to reduce the complexity of determining the canonical label of a graph.

The canonical labeling of a graph is performed, based on various conditions explained below. This labeling is then used to eliminate the redundant subgraphs, to obtain a complete non-redundant global semantic graph. The following are the conditions for the canonical labeling of the subgraphs.

1. Given two subgraphs $G = \{G_1, G_2\}$, they are identical when the nodes of the $G_1$ are connected with an active verb by edges $E$ and the nodes of $G_2$ are connected with the passive verb by the same edges $E$. 

Example 7.1

Active sentence: Raman Ravanana konRaan
Raman killed Ravan

Passive sentence: Ravan Ramanaaal kollappaattaan
Ravan was killed by Raman

Both the sentences convey the same meaning, but with different sentence structures. The equivalent semantic graphs are shown in Figure 7.5.

![UNL Semantic Graphs of (a) “Raman killed Ravan” (b) “Ravan was killed by Raman”](image)

Figure 7.5 UNL Semantic Graphs of (a) “Raman killed Ravan” (b) “Ravan was killed by Raman”

Figure 7.5 (a) shows the UNL semantic graph of the active sentence and (b) shows the semantic graph of the passive sentences. Both graph representations are equivalent, constructed from two different sentence structures (active and passive respectively) conveying the same meaning. The variations are represented as attributes associated with the nodes. These variations are handled while constructing the semantic graphs, using morpho-semantic features. By default, the nodes indicating verbs without the attribute “@active” are considered as active verbs. While looking for redundant subgraphs, these graphs could not be captured, because the head node is a passive verb in (b). Therefore, masking the tuples associated with the nodes is
done to determine the redundant subgraphs. The masking process is decided based on the maximum match of the subgraph (b) with the subgraph (a). The canonical label is then assigned to the identical subgraphs.

2. Assume that Graph G1 is constructed for a verbal phrase and Graph G2 is constructed for an adjectival phrase. The subgraphs of G1 and G2 are said to be identical, when the verbal node \( (N_v) \) is connected to the node \( (N_i) \) with “aoj” relation, and the same node \( (N_v) \) acts as a modifier \( (N_M) \) of the graph G2, which is connected to the node \( (N_i) \) with the “mod” relation. The canonical label is then assigned to the subgraph \( (N_M - \text{mod} - N_i) \) of G2. The rest of the nodes connecting \( N_M \) are transformed to the verbal node \( (N_v) \) of the G1 with their corresponding edges (semantic relations). The example below shows the sentences with the verbal phrase and adjectival phrase, and Figure 7.6 shows the process of matching identical subgraphs.

**Example 7.2**

Verbal Phrase: திருச்சிராப்பள்ளி, இந்து மாபெரும் அமினை பதிவு செய்யப்பட்டது.

Thiruchirapalli indiyaavin tamilnadu maanilaththil amainthuLLathu.

*Tiruchirapalli is located in Tamil Nadu state of India.*

Adjectival Phrase: காவிரி ஆர்ரங்கரையில் அமினைக்கான திருச்சிராப்பள்ளி, காவிரிக்கும் உட்பட தர்க்கப்பட்ட பெப்லன்றாந் நகராலாக வந்துள்ளன.

Kaaaviri aARangaraiyil amainthuLLa thiruchirapaLLi, tamilakaththil uLLa naangu mukkiyamaana nakarankaLLi ondraagum.
Tiruchirapalli is one of the four most important cities in Tamil Nadu, which is located on the banks of the Cauvery.

In some cases, certain components of the verbal phrases act similar to the components of the adjectival phrases, i.e. the meaning conveyed by the components of the verbal and adjectival phrases are the same. Such identical phrases are identified using the above condition, and each component is assigned a canonical label.

The semantic graphs of the sentences are shown in Figure 7.6. The subgraphs that are identical, are marked inside a circle in Figure 7.6 (a) and (b). Figure 7.6 (c) shows the complete graph with the redundant subgraphs removed and the other nodes/subgraphs merged.

![Figure 7.6](image-url)
The canonical labels are then processed to remove the redundant subgraphs. After removing the redundancies in the semantic graphs, a complete semantic graph of the multiple documents is generated.

7.5.3 Learning of Graph Operations

In this section, we use the single-stage pattern based bootstrapping approach described in section 3.6.1 for learning and flagging various graph operations such as the deletion / selection, insertion and modification, of the nodes and edges of the global semantic graph. While learning the graph operations for single-document summarization, the decision on setting the flags follows a set of heuristics (described in section 7.4.1.1), which was based on the semantic property of the nodes, that had a high frequency of occurrence in the document. These flags are now associated with the global semantic graph formed from the semantic graphs of single documents and are used to identify the important components for a given query. The flagged graph operations on the nodes and edges are then actually carried out, during the query graph integrated spreading activation technique.

7.5.4 Query Focused Summarization

In a single or multi-document summarization, the summary is meant to give an overview of the information in the documents. Contrastingly, when the summary is produced in response to a user query, the query determines what information is appropriate for inclusion in the summary, making the task potentially more challenging. In this section, we focus on the selection of important components among the flag assigned nodes and edges of the global semantic graph.

In this context, we use the query graph (\(G_0\)), document semantic graph (\(G_{DOC}\)) and global semantic graph (\(G_{GLOBAL}\)). The query given by the users consists of a single term or multiple terms. Here, both the single term
and multiple term queries are converted into their semantic representations, viz, the query graph \((G_Q)\).

Searching for a query graph \((G_Q)\) in the global semantic graph is carried out in two ways. One is to search for the exact match in which all the nodes and edges in the query graph are precisely matched with the subgraph of \(G_{GLOBAL}\). The existing query focused techniques (Bhaskar & Bandyopadhyay 2010; Chali & Joty 2008; Berger & Mittal 2000) failed to handle the situations, where the query words are not exactly as presented in the document. In the proposed work, we tackle such situations by using various levels of matching proposed by Umamaheswari et al (2011), in which they introduced various levels of matching the query words, the concept of the query words, and relations between the query words. We use the matching procedure used by Umamaheswari et al (2011), where it searches for a query graph match in the global graph \((G_{GLOBAL})\). The matched subgraph is marked as a starting node, and proceeds over the entire global semantic graph \((G_{GLOBAL})\), where the direction of traversal is decided based on the user query, the node in which the activation is performed, and the edges connecting that node. While the activation signals the node for traversal, the flagged graph operations are carried out to identify the essential components of the summary. The traversal among the nodes of the global graph is performed by the spreading activation algorithm, which we will discuss in section 7.5.4.1.

7.5.4.1 Graph compression through spreading activation

The spreading activation theory has been applied in many other research fields, such as information retrieval (Bollen et al 1999), hypertext structure analysis (Pirolli et al 1996), Web trust management (Ziegler & Lausen 2004) and collaborative recommendation (Huang et al 2003). To obtain a summary relevant to the given query, tracking the user query is necessary to narrow down the selection process. This section takes the spreading activation theory one step further, by incorporating the query graph
along with the node and edge information, to estimate the strength of the nodes and determining the direction of spreading over the nodes for selection. Thus we integrate the query graph along with the node and edge information, for the spreading activation. We initially start spreading the activation from the query nodes of the global semantic graph, and recursively find the important components by spreading over the nodes, keeping track of the query graph which is based on the node and edge-specific features throughout the global semantic graph. To start the activation of spreading towards the nodes in the graph, the query graph is signaled as the starting node, and while spreading over the graph it concurrently tracks the query graph, during the selection of each node and edge for spreading. As already discussed, the query graph consists of single or multiple nodes, where the activation also starts at multiple points, when the query graph is scattered in the global graph. The multiple points to start the activation process, are decided using various distance and similarity measures.

The similarity measure is performed at the word and context levels. To achieve the word level similarity, we use the UNL Ontology described in section 1.4.3, to obtain semantically similar concepts which get highlighted, while the spreading activation is performed on the global graph. The semantically related concepts of the query words are identified, using semantic similarity scoring. This similarity scoring is measured between the semantic constraints of the query words and the concepts of the global graph, with the semantic constraints of UNL Ontology. The similarity measure is defined in Equation (7.7).

\[
SIM(G_Q(C_i), G_{GLOBAL}(C_j)) = DIST(\text{Parent}_{sc}(G_Q(C_i)_{sc}), G_{GLOBAL}(C_j)_{sc})
\]  

(7.7)

where

\[C_i\] – concepts in the query, where \(i \geq 1\)

\(\text{Parent}_{sc}\) – Parent UNL Semantic constraint in UNL Ontology,
SC – UNL semantic constraint, and
C_j – concepts in the global graph

To obtain the context level similarity, we use the dependent relations (described in section 6.9.6), in which the similarities between various semantic relations are identified.

To find the components related to the given query graph, we spread an activation signal starting from the query words/concepts and their expansions (in a manner similar to (Mani & Bloedorn 1999). As we traverse the graph starting from these nodes, the signal is propagated by assigning a weight to each edge and each node traversed, based on the signal strength. The signal strength is computed, based on the nodes and their connected edges, and the query graph. The modified query graph integrated spreading activation algorithm is given below.

Algorithm: Spread-Activation \((G_Q, e, c \leftarrow 0, r)\)

Require: Starting vertex \(G_Q\) (consists of nodes and edges).
Require: Activation energy \(e > 0\).
Require: Energy \(c\) accumulated on graph vertices.
Require: Relation \(r\) accumulated on graph edges.
Require: Weight \(w\) assigned on graph edges

\[
c_v \leftarrow c_v + e + r_w
\]

\[e' \leftarrow e/\text{Vertex-degree} (G_0)\]

if \(e' > 0\) then

For all vertices \(t\) such as, there exists an edge from \(v\) to \(t\) (represented as a pattern) do

\[c \leftarrow \text{Spread-activation} (G_Q, e', c, w)\]

end for

end if

return \(c\)
7.5.4.2 Termination check

One of the important aspects of summarization using spreading activation is where to stop the activation of spreading in the global semantic graph. For the single document summarization, the activation spreads over the entire document semantic graph for the selection of important components. However, to obtain a summary relevant to a user query, the termination check is an essential task for the selection of important components. The constrained spreading activation algorithm (Crestani 1997) uses different constraints to control the spreading activation. In the proposed approach, certain constraints discussed by Crestani (1997) are modified and utilized to control the spreading activation.

Distance constraints: As discussed by Crestani (1997) the spreading of activation should stop the process, when it reaches nodes that are too far away from the initially activated ones. This may fail in cases, where the termination check is performed in the global semantic graph (multiple documents connected). Thus, in the proposed approach, the distance is measured between the nodes and edges (concepts and relations) of the global semantic graph and the query graph. This constraint can be used to stop the process of spreading.

Fan-out constraints: As described by Crestani (1997), the spreading of activation should stop at nodes with very high connectivity (fan-out) to avoid a spreading that would be too wide. In the proposed approach, this constraint may fail because an entity can connect with other nodes of different sentences of multiple documents, and thus, it is strongly connected. If that strongly connected node is used as a signal for stopping, it may result in obtaining an incomplete summary. Thus, these fan-out constraints contradict the process of the proposed approach, and are not used.
**Path constraints:** Crestani (1997) suggested a set of inference rules, to decide which path to choose by determining the signal strength for the activation of spreading. In the proposed approach, the direction of traversal among the nodes is decided, based on the semantic information (both word and relation based) associated with the given query graph, and the nodes and edges of the global semantic graph under consideration.

**Activation constraints:** A threshold function has been suggested by Crestani (1997) to activate the nodes for spreading. The signal strength obtained between the nodes above the threshold value gets higher priority over the low signal strength below the threshold value and gets activated. The signal strength has been used to activate the nodes for spreading alone. In the proposed approach, the activation of the nodes as well as the edges is decided using query graphs, and the nodes and edges of the global semantic graph. Instead of using the threshold function used by Crestani (1997), the nodes and edges are activated by the information (word and context (relation) based) associated with the query graph and the global graph.

From the analysis, these three constraints are utilized to decide which nodes to select for activation and where to stop the activation process. To obtain the word based semantic similarities, we use the UNL Ontology, and the similarity is computed by Equation (7.7). The relation based similarity is computed using the initial conditions based on the semantic relations obtained, between the concepts.

**7.6 EVALUATION**

The evaluation of the summary is carried out from different perspectives. They are

- Human versus machine generated summary
- Extractive versus abstractive summary
There are also automatic methods for summary evaluation, such as ROUGE (Lin 2004), which gives a score based on the similarity in the sequences of words, between a human-written model summary and the machine summary.

ROUGE (Lin 2004) is an automatic evaluation metric that computes an n-gram similarity score between the model summary and the summary to be evaluated. Several types of ROUGE measures exist, and the one with the highest correlation with manual scores is ROUGE-2 recall – the recall of model summary bigrams. Very high correlations between manual metrics and ROUGE have been observed (Hoa & Karolina 2009).

The performance of the proposed approach is evaluated by analyzing the compressed graph with the original semantic graph, in terms of the number of nodes before and after compression, the number of newly created nodes and modified nodes, number of nodes before and after merging. We investigated and compared both the original document semantic graph and the compressed summary graph automatically. Furthermore, the summary generated by spreading activation is evaluated, and compared with the human generated summary.

The evaluation of summaries is performed, using intrinsic and extrinsic methods. While the intrinsic method of evaluation is based on user judgments with both precision and ROUGE scores, the extrinsic method is based on the information retrieval task, with queries and evaluation of the FIRE task (FIRE 2010).

To evaluate the precision of the summaries generated, we have compared our system with a template based summary (Subalalitha et al 2011) and extractive summary, using a rule-based approach. The template-based summary proposed by Subalalitha et al (2011) introduced seven templates,
specific to the Tourism domain for generating the summary. The concepts of
the natural language words matching with the UNL semantic constraints of
each template, are considered for the summary. However, while this template
based information is language independent, they are domain specific.
Adapting these templates to the generic domain requires different or more
templates. In fact, we have added new template information to adapt to
various domains, such as health, news and sports. For obtaining the extractive
summary, we have designed a rule-based extractive summarization approach,
which extracts important sentences from multiple documents. First, the
sentence constituents are identified and extracted, using a set of rules which
was designed with the word based morphological features. The semantic
constraints of each word are obtained using the UNL list which consists of the
root form of the natural language word, its equivalent English translation and
the UNL semantic constraint obtained from the UNL Knowledge Base (UNL
KB). The important sentences are then extracted, using the term and concept
frequencies, and co-occurring concepts in a sentence.

One of the widely used measures for evaluating summaries, is the
ROUGE score (Lin 2004). ROUGE stands for Recall-Oriented Understudy
for Gisting Evaluation. It includes measures to automatically determine the
quality of a summary, by comparing it with other (ideal) summaries created
by humans. The measures count the number of overlapping units, such as n-
gram, word sequences, and word pairs between the computer-generated
summary to be evaluated and the ideal summaries created by humans. The
proposed summarization approach is evaluated using the ROUGE scores. For
each run, four scores, such as ROUGE-1, ROUGE-2, ROUGE-L, and
ROUGE-SU4 were computed. In addition, we introduce a new score for
graph based summarization called ROUGE-G, to evaluate the summary in
terms of nodes and edges. The quality of the summary is investigated and
measured, by counting the number of overlapping units (nodes and edges),
between a predicted summary graph and the generated summary graph obtained by the proposed approach.

We have also investigated the summaries obtained using different methods, with the reference summaries by ROUGE scores. We have compared the proposed approach with three different methods, such as the summary obtained using term frequency, concept frequency and original spreading activation. From the analysis, the edge integrated spreading activation technique performs better when compared to all other methods. This is because, instead of considering the node information alone for determining the strength, the proposed approach integrated the edge information to find the strength of the nodes for activation, which results in obtaining the good quality of the summary.

Since the evaluation is carried out on the compressed semantic graph, which consists of nodes and edges, it is difficult to evaluate the common subsequence, co-occurring concepts etc. Thus, to evaluate the summary graphs, we introduce a new ROUGE score called ROUGE-G (i.e. ROUGE- Graphs). ROUGE-G computes the number of nodes and edges in the candidate summary graph, that overlaps those of the reference summary graph.

7.6.1 Single Document Summarization

7.6.1.1 Comparison with template-based and extractive summaries

The summary obtained using these three approaches, were given to three experts for the evaluation of precision, and the results given by them are shown in Table 7.2 below.
Table 7.2 Comparison of the Different Summarization Methods

<table>
<thead>
<tr>
<th>Methods</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User 1</td>
</tr>
<tr>
<td>Template based Summary</td>
<td>0.48</td>
</tr>
<tr>
<td>Extractive Summary</td>
<td>0.53</td>
</tr>
<tr>
<td>Proposed Approach (Abstractive Summary)</td>
<td>0.71</td>
</tr>
</tbody>
</table>

From Table 7.2, the summaries obtained from the three different methods are compared with a reference summary. When compared to the template based and extractive methods, the proposed graph based approach, generates a summary which covers most information contained in the document. In the template based method, the different sets of templates need not cater to the all the information, and filling the template information need not necessarily explore all the information widely. This may result in a decrease in precision. In rule based extractive summarization, the sentence units are extracted based on the frequency of the occurrence of the concepts and co-occurring concepts in a sentence, which may not extract the important concepts with low frequency. In contrast, the summary obtained using the proposed graph based approach, has information similar to that of the reference summary, and the precision is shown in Table 7.2.

7.6.1.2 Evaluation using ROUGE Scores

Table 7.3 shows various ROUGE scores for different summarization methods. Among the methods, the Concept based Summary (CF) is better than the Term based Summary (TF). Instead of searching for terms, the CF method searches for concepts, and thus increases the performance. While evaluating the spreading activation, we found that integrating the edge information results in producing a better summary, when
compared to the spreading activation proposed by Quillian (1968), because, the original spreading activation starts with a single node, and does not consider the edge information for estimating the signal strength. Instead in the proposed approach, the signal strength is estimated using the information associated with the nodes, and their connected edges of the global semantic graph. Thus the performance increases when compared to other methods. The newly introduced score is ROUGE-G, which can be used to evaluate the semantic summary graphs. The existing ROUGE scores focused on the n-gram, co-occurrence, subsequence etc. The ROUGE scores except 1-gram, investigate the structural information to maintain the coherence and readability of the summary. However, these scores are used to evaluate the extractive summaries. To evaluate the summary graphs, we introduce ROUGE-G where the overlapping nodes and edges of the candidate summary graphs with the reference summary graphs are estimated. Thus, in the proposed graph based approach, the number of nodes and edges of the candidate summary graphs overlapping the reference summary graph gives 65% accuracy.

**Table 7.3 ROUGE Scores of Summary obtained using Various Methods**

<table>
<thead>
<tr>
<th>Methods</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-L</th>
<th>ROUGE -SU4</th>
<th>ROUGE-G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Term Frequency (TF)</td>
<td>0.31717</td>
<td>0.2432</td>
<td>0.2063</td>
<td>0.1632</td>
<td>0.1505</td>
</tr>
<tr>
<td>Concept Frequency (CF)</td>
<td>0.35550</td>
<td>0.3015</td>
<td>0.2635</td>
<td>0.1873</td>
<td>0.2013</td>
</tr>
<tr>
<td>Spreading Activation</td>
<td>0.39810</td>
<td>0.3210</td>
<td>0.2803</td>
<td>0.2235</td>
<td>0.1865</td>
</tr>
<tr>
<td>(Quillian, 1968)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Edge Integrated</td>
<td>0.7023</td>
<td>0.4861</td>
<td>0.4603</td>
<td>0.3980</td>
<td>0.6593</td>
</tr>
<tr>
<td>Spreading Activation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Concepts – Relations</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(nodes and edges)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**7.6.1.3 Evaluation of edge integrated spreading activation**

This section discusses the performance of the proposed edge integrated spreading activation algorithm for obtaining the important
components for abstractive summarization. We have compared our approach to the original spreading activation theory (Quillian 1967). The spreading activation starts with a node, and spreads over the graph to extract the important components for the summary. The signal strength is calculated based on the weight of each node under consideration, and the number of connected edges. Instead, in the proposed graph based approach, the information associated with each node and edge is utilized to calculate the signal strength.

![Original Vs Edge Integrated Spreading Activation](image)

**Figure 7.7 Comparison of Original Spreading Activation and Edge Integrated Spreading Activation (Impact of signal decay in Spreading Activation on Summarization Performance)**

Figure 7.7 shows the impact of signal decay while performing spreading activation. Similar to the evaluation methods discussed by Nastase (2008), the results shown in Figure 7.7 for decay values of 0.1, 0.5, 0.95, 0.99, 0.999, 0.9999, 1 – indicate that faster decay (reflected through a higher decay value) keeps the summary focused initially and slowly spreads out in a wide manner. The ROUGE-G score is computed to evaluate the edge integrated spreading activation theory. When compared to the baseline
approach, the proposed approach, which considers the edge information for spreading, produces better results.

7.6.2 Query Focused Multi-Document Summarization

The evaluation is two fold: evaluation of the redundancy elimination and evaluation of query focused summaries. The main contributions of this work are:

1. Redundancy elimination of semantic graphs is carried out in two stages: during the construction of the global graph and after the construction of global graphs. We use various semantic measures, such as word based and context based ones, to eliminate the redundant subgraphs.

2. Query focused summary generation, by selecting the important components through the query graph integrated spreading activation procedure.

In this section, we first discuss the evaluation on the elimination of the redundant subgraphs, and then describe the various measures used for evaluating the query relevant summaries and query integrated spreading activation.

7.6.2.1 Evaluation of the redundancy elimination in semantic graphs

We first analyzed the extent to which the proposed approach eliminated the redundant subgraphs during the two stages as discussed in Section 7.5.2. Here, we use the entropy measure to evaluate redundancy elimination. The entropy of a source represents how much it can be compressed by an optimal lossless algorithm. It is a measure of information and lower entropy values indicate better redundancy elimination without loss of information.
The entropy $H$ of a given source $X$ is defined as

$$H(X) = - \sum p(x) \log(p(x))$$

where the source $X$ is modeled as a random variable, and $p(x)$ is the probability of an element $x \in X$ occurring.

The entropy of a given source is affected by the number of elements in $X$. Thus a normalized measure, redundancy, is better for comparing multiple sources. Redundancy compares the actual entropy of a source to its theoretical maximum entropy.

$$R(X) = 1 - \frac{H_{\text{actual}}(X)}{H_{\text{max}}(X)}$$

In this context, the entropy is affected by the number of nodes and edges in the global semantic graph ($G_{\text{GLOBAL}}$). The redundancy elimination compares the actual entropy of the global semantic graph to its compressed semantic graph (after redundancy elimination). Table 7.4 shows the entropy measures that is obtained using topic driven approach (Nastase 2008), where no explicit redundancy elimination is carried out and the entropy values obtained during each stage of redundancy elimination discussed in this paper.

### Table 7.4 Entropy for the Elimination of Redundant Subgraphs

<table>
<thead>
<tr>
<th>Approaches</th>
<th>Entropy Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline Approach</td>
<td>0.53</td>
</tr>
<tr>
<td>Our approach</td>
<td>0.316</td>
</tr>
</tbody>
</table>

We have compared the redundancy elimination with the baseline approach proposed by Nastase (2008). From Table 7.4 we can prove that lower entropy value leads to higher redundancy elimination.
7.6.2.2 Evaluation of the query oriented summary

The evaluation of summaries is performed, using intrinsic and extrinsic methods. While the intrinsic method of evaluation is based on user judgments with both precision and ROUGE scores, the extrinsic method is based on the information retrieval task, with queries and evaluation of the FIRE task (FIRE 2010).

7.6.2.2.1 Comparison with template-based and extractive summaries

Table 7.5 shows the summaries obtained from the three different methods are compared with a reference summary. When compared to the template based and extractive methods, the proposed graph based approach, generates a summary more relevant to the user query. In the template based method, the different sets of templates need not cater to the different user queries, and filling the template information need not necessarily satisfy the user’s need. This may result in a decrease in precision. In rule based extractive summarization, the sentence units are extracted based on the frequency of the occurrence of the concepts and co-occurring concepts in a sentence, which may not be relevant to the query. In contrast, the summary obtained using the proposed graph based approach, has information similar to that of the reference summary, and the precision is shown in Table 7.2.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>User 1</td>
</tr>
<tr>
<td>Template based Summary</td>
<td>0.248</td>
</tr>
<tr>
<td>Extractive Summary</td>
<td>0.224</td>
</tr>
<tr>
<td>Proposed Approach (Abstractive Summary)</td>
<td>0.592</td>
</tr>
</tbody>
</table>
7.6.2.2.2 Evaluation using ROUGE Scores

Table 7.6 shows various ROUGE scores for different summarization methods. Among the methods, the concept based summary (CF) is better than the term based summary (TF). Instead of searching for terms, the CF method searches for concepts, and thus increases the performance. While evaluating the spreading activation, we found that integrating the edge information and query graphs results in producing a better summary, when compared to the spreading activation proposed by Quillian (1968), because, the original spreading activation starts with a single node, and does not consider the edge information for estimating the signal strength. Instead in the proposed approach, the signal strength is estimated using the information associated with the nodes, and their connected edges of the global semantic graph, and the query graph. Thus the performance increases when compared to other methods.

The newly introduced score is ROUGE-G, which can be used to evaluate the semantic summary graphs. The existing ROUGE scores focused on the n-gram, co-occurrence, subsequence etc. The ROUGE scores except 1-gram, investigate the structural information to maintain the coherence and readability of the summary. However, these scores are used to evaluate the extractive summaries. To evaluate the summary graphs, we introduce ROUGE-G where the overlapping nodes and edges of the candidate summary graphs with the reference summary graphs are estimated. Thus, in the proposed graph based approach, the number of nodes and edges of the candidate summary graphs overlapping the reference summary graph gives 53% accuracy.
Table 7.6 ROUGE Scores of Summary obtained using Various Methods

<table>
<thead>
<tr>
<th>Methods</th>
<th>ROUGE-1</th>
<th>ROUGE-2</th>
<th>ROUGE-L</th>
<th>ROUGE-SU4</th>
<th>ROUGE-G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Term Frequency (TF)</td>
<td>0.31717</td>
<td>0.2432</td>
<td>0.2063</td>
<td>0.1632</td>
<td>0.1505</td>
</tr>
<tr>
<td>Concept Frequency (CF)</td>
<td>0.35550</td>
<td>0.3015</td>
<td>0.2635</td>
<td>0.1873</td>
<td>0.2013</td>
</tr>
<tr>
<td>Spreading Activation (Quillian 1968)</td>
<td>0.39810</td>
<td>0.3210</td>
<td>0.2803</td>
<td>0.2235</td>
<td>0.1865</td>
</tr>
<tr>
<td>Query Integrated Spreading Activation</td>
<td>0.6258</td>
<td>0.4797</td>
<td>0.4656</td>
<td>0.3685</td>
<td>0.5329</td>
</tr>
</tbody>
</table>

7.6.2.2.3 Extrinsic evaluation – FIRE task

We have evaluated our system using the query set of FIRE tasks (FIRE 2010) and investigated the generated summary, using the guidelines provided by the FIRE evaluation task. Each query is tagged with <title> and <title>, and the guidelines are given in between the tags <narr> and </narr>. An example of a Tamil query and its equivalent English query, is given below.

<Top lang="ta">
<title>பொற்றுக்கொள்ளுபட்டர்களின் காலத்துருக்கள்</title>
<desc>பொற்றுக்கொள்ளுபட்டர்கள் காலத்துருக்கள்</desc>
<narr>சுத்தமான அவைதிகள், புதுப்பெற்ற பொற்றுக்கொள்ளுபட்டர்கள் மையாகச் சுத்தமான திருட்சிகளின் செயல்பாடுகளைக் காண்பதற்கு மிகச் சுத்தமான அவைதிகள் செயல்பாடுகளைக் காண்பதற்குத் திருத்தமான பொற்றுக்கொள்ளுபட்டர் மையாகச் சுத்தமான திருட்சிகளின் செயல்பாடுகளைத் தெரியுமோ விளக்கம்</narr>
</Top>

<Top lang="en">
<title>Shoe throwing at persons of eminence</title>
<desc>Incidents involving throwing of shoes at persons of eminence</desc>
<narr>In the recent past, an Iraqi journalist threw a shoe at the American President George W. Bush. In a similar incident, Jarnail Singh, a journalist, threw a shoe at the Indian Home Minister, P. Chidambaram. Relevant documents should contain information about such incidents in which a shoe was thrown at some well-known person. Information about legal steps taken against the offenders is irrelevant.</narr>
</Top>
The FIRE evaluation task consists of 2,000,000 documents. First, the sentences in each document were converted into a UNL semantic graph (referred to as the sentence semantic graph $G_{SENT}$) representation, using a rule-based approach described in section 3.4. The common concepts, anaphoric nodes and coreference nodes of the sentence semantic graphs are connected, to form a document semantic graph ($G_{DOC}$) and the global semantic graph ($G_{GLOBAL}$). While building the global semantic graph, various redundant information is removed, and the possible graph operations are assigned to the nodes and edges of $G_{GLOBAL}$. The query integrated spreading activation algorithm is applied to obtain the query specific summary graph.

The proposed query focused summaries are tested using a set of FIRE task queries. Given a query, the important components are selected using a query integrated spreading activation theory. These summaries are evaluated with the reference summary, using three experts with the guidelines given in the FIRE task. The queries are classified into three – candidate summaries that are very much closer to the reference summary and the given query ($>0.75$), candidate summaries that are closer ($<0.75$, $>0.50$), and candidate summaries that are less relevant ($<0.50$). In Table 7.4, 15 queries are produced, five for each category.

The query relevant summary graph is compared with the three reference summary graphs. The query graph is given for various levels of matching (Umamaheswari et al 2011) in the global semantic graph, where the semantically similar concepts and relations have been considered. The edges (relations) connecting the nodes (concepts) also play an important role in the activation of spreading of the nodes and edges for a summary. While analyzing the query graphs ($>0.75$) and summary graphs, though the information is scattered over the global graph, the information relevant to the
given query graph is obtained by activating the signals at multiple nodes of the global graph. However, in cases where the spreading activation is performed in a wide manner, the query graph does not contain any relationship among the concepts, and thus the concepts are said to be dangling nodes. In such cases, the activation starts the process in a wide manner which results in a less relevant summary.

For instance, consider an example query (11 & 12) given in Table 7.7 – “இந்த படங்கள் ஓரை நடிகை ஓரை படத்தின் மூல்நிலைக்கு (intiya aninešan tūrai tiraippatānkal) - Indian animation industry films” and “இந்த படங்கள் ஓரை படத்தின் (intiya pādu) ஓரை நடிகை ஓரை படத்தின் மூல்நிலைக்கு (Ātars vīṭtu vacati vāriyam mōcați rājiṇāmā) - Adarsh Housing Society scam resignation”. Query 11 has the average rating of 0.36. This is because the activation searches for the list of films instead of the technical details of animation, and thus decreases the relevancy. Query 12 has dangling concepts and the term “resignation” does not connect to any concepts in the query. Thus the spreading activation search for the issues related to “Adarsh Housing Society scam” ignores the information regarding the “resignation”. Similarly, for Query 15, it searches for the person involved. Initially, the information obtained is relevant to the query. While activation spreads over the global graph, it focuses towards the biography of the person, which results in a less relevant summary, when compared to the reference summary.
Table 7.7 Evaluation of Query Focused Summaries

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Queries</th>
<th>Summarization rating</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Queries (&gt;0.75)</td>
<td>User1</td>
</tr>
<tr>
<td>1</td>
<td>செயலாக்கச் சுற்றுச்சூழல் பாதுகாக்க (எரிவு மரணம்) - Steve Irwin death</td>
<td>0.78</td>
</tr>
<tr>
<td>2</td>
<td>ஆசிரியர்களின் விளக்கம் நல்லூர பல்கலைக் கழகம் கைத்து பத்தைகள் (Astridiyal intiya maqavarka mitu tikkutal) - Attacks on Indian students in Australia</td>
<td>0.80</td>
</tr>
<tr>
<td>3</td>
<td>பிரபலம் மற்றும் முக்கியத்துவம் மைத்திட்டு (Pirapala maqitarka mitu manjani vīcēru) - Shoe throwing at persons of eminence</td>
<td>0.82</td>
</tr>
<tr>
<td>4</td>
<td>2001 ஈன்றையார்களுக்கு வழங்கப்பட்ட பெரிய பெரிய புராண (2001 Ilakkiyattūkkā npal paricu veppavar) - 2001 Nobel Prize winner for Literature</td>
<td>0.76</td>
</tr>
<tr>
<td>5</td>
<td>பாக்திஸ்தானில் புலக்குக் காண்பிட்டு டீசனின் குழுக்கள் கைத்து பத்தைகள் (pakistanil sri lanka tekkaya kikket kulu mitu tikkutal) - Attack on Sri Lankan national cricket team in Pakistan</td>
<td>0.79</td>
</tr>
<tr>
<td></td>
<td>Queries (&gt;0.50, &lt;0.75)</td>
<td>----------------------</td>
</tr>
<tr>
<td>6</td>
<td>பிராப்னா பெண் கைத்து பத்தைகள் (aiṟpū appṟu totakka vaavam pirapalam) - iPhone iPad launch design popularity</td>
<td>0.70</td>
</tr>
<tr>
<td>7</td>
<td>பெபுலும் விளக்கம் நல்லூர பல்கலைக் கழகம் என்பதின் என்னது (ppulum melaam mēkkal appakāyppai ceyalppuka intiyē) - Bill and Melinda Gates Foundation philanthropic activities India</td>
<td>0.6</td>
</tr>
<tr>
<td>8</td>
<td>கீழ்ப்பட்ட மொழியில் விளக்கம் பல்கலைக் கழகம் என்னது (karppappai vēy purunçy vilippuraru tutppu cikiccai) - Cervical cancer awareness treatment vaccine</td>
<td>0.58</td>
</tr>
<tr>
<td>9</td>
<td>இமயன் காண்பிட்டு முன்னோட்டத்தில் போல்வோட்டம் செய்யுங்கள் (Immay kā purunçy maruttuva maqai pākistānum) - Imran Khan cancer hospital Pakistan</td>
<td>0.55</td>
</tr>
<tr>
<td>10</td>
<td>இதுவோட்ட மொழியில் பல்கலைக் கழகம் என்னது (Intiya kutimakān pākistānum utālam) - Indian citizen Pakistani spy</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>Queries (&lt;0.50)</td>
<td>----------------------</td>
</tr>
<tr>
<td>11</td>
<td>இன்னொரு அறிவியல் என்ன எடுக்கியுங்கள் (intiya aqīmcān thārāurai tiraippaṭukākā) - Indian animation industry films</td>
<td>0.37</td>
</tr>
<tr>
<td>12</td>
<td>ஆட்சி காட்சியில் கருப்பு குடிக்குச் சேர்ந்து பயணக் கால (Ams 300 vacati wariyam mēça mōji mūrīn) - Adarsh Housing Society scam resignation</td>
<td>0.35</td>
</tr>
<tr>
<td>13</td>
<td>சுருக்குகள் கைத்து ரத்நகர ரகு சர்க்கார் (cūrīnik verscas cārcē) - Satanic Verses controversy</td>
<td>0.33</td>
</tr>
<tr>
<td>14</td>
<td>பிரபலம் பிரபலம் கைத்துத் தொடும் 1 கைத்து பத்தைகள் (intiyē intiyē mutal narumā 1 cārpappātāi) - First F1 circuit India</td>
<td>0.28</td>
</tr>
<tr>
<td>15</td>
<td>பிரபலம் வெளியிலோ பிரபலம் வெளியிலோ (tiraippaṭat tārjikka kūļuvā peṉ talaivar) - Film Censor Board woman Chairperson</td>
<td>0.23</td>
</tr>
</tbody>
</table>
7.6.2.2.4 Evaluation of query integrated spreading activation

This section discusses the performance of the proposed query integrated spreading activation algorithm for obtaining the important components for abstractive summarization. We have compared our approach with the original spreading activation theory (Quillian 1967), and the edge integrated modified spreading activation algorithm (described earlier) used for single document summarization. We have also compared our approach with the baseline approach proposed by Nastase (2008), where the spreading activation is carried out to obtain a topic driven multi-document extractive summarization. The spreading activation starts with a node, and spreads over the graph to extract the important components for the summary. In the proposed approach, instead of starting with one node, the activation starts with a query graph (consisting of multiple nodes) and spreads over the global graph. As discussed by Nastase (2008), in their method, the signal strength is calculated based on the weight of each node under consideration, and the number of connected edges. Instead, in the proposed graph based approach, the information associated with each node and edge is utilized to calculate the signal strength.

![Original Vs. Query Integrated Spreading Activation](image)

**Figure 7.8** Comparison of Original Spreading Activation and Query Integrated Spreading Activation (Impact of Signal Decay in Spreading Activation on Summarization Performance)
Figure 7.8 shows the impact of signal decay while performing spreading activation. Similar to the evaluation methods discussed by Nastase (2008), the results shown in Figure 7.8, for decay values of 0.1, 0.5, 0.95, 0.99, 0.999, 0.9999, 1 – indicate that faster decay (reflected through a higher decay value) keeps the summary more focused around the user query graph. The ROUGE-G score is computed to evaluate the query integrated spreading activation theory. When compared to the baseline approach, the proposed approach, which considers the edge information and the query graphs for spreading, produces better results.

Table 7.8 shows the summaries obtained, using different methods such as Template based, extractive and the proposed graph based approach. While investigating the summaries manually, the summary obtained using templates, has irrelevant and noisy concepts, which are not relevant to the user query, the extractive summary has information irrelevant to the user query, and the abstractive summary obtained using the proposed approach, has several advantages over the other two methods. Though the abstractive summaries require a natural language generation task, the summaries are meaningful and are relevant to the user query. The noisy concepts and irrelevant sentences are highlighted in Table 7.8.
Table 7.8  Sample Summary obtained by Different Summarization Methods for a user Query “திருச்சிராப்பள்ளி கோயில்கள் கொண்டாட்டம்” and its Equivalent English Query “Tourist places in Tiruchirapalli”

<table>
<thead>
<tr>
<th>Methods</th>
<th>Tamil Summary</th>
<th>Equivalent English Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Template based Summary</td>
<td>சிகுதி மகாராஜ பேசிக்குவிப்பூச்சி, பல்லும்பூச்சி, வலாச்சூச்சி, திருத்தாலைநாகார், நீர்த்தாலைநாகார், பெருநீர்த்தாலைநாகார், கோயில்கள் கொண்டாட்டம், மலைகளின் தோற்றம், வனத்தோற்றம், ஆரியாலைநாகார், பாலாமகாரியார், காளைனிலையார், காவர்கியார், கோலைசிறார், காலைசிறார், குளுநிலையார், தோற்றம் தன்னை செலுத்துவதாக கோயில்கள் கொண்டாட்டம்.</td>
<td>This page talks about the places such as Tiruchirappalli, Tamil Nadu, India, tiruvanaikka, Kyle Hill, tiruvanarka Island, Rockfort, ponmalai, apisekapuram, Srirangam, Aiyamangalam, manacanallur, Nagapattinam, Chennai, Theni, Tuticorin, kantalpet, Ramanathapuram, Chidambaram, melainai, cemetery, Central Bus Station , tavern bus Stand, Cauvery river, and kollita river.</td>
</tr>
</tbody>
</table>

Tayumana Swami Temple, atop Ganesha Temple, Mariamman Temple, arankanata Swami Temple, vekkali Amman Temple, campukecuvarar Temple, Subramanya Swamy Temple, pacevarna Swami Temple, amranesvarar Swami Temple, uttamar Temple, Prasanna Venkatachalapathy Temple, nilivanesvarar temple and shrines can be found here.
Table 7.8  (Continued)

<table>
<thead>
<tr>
<th>Methods</th>
<th>Tamil Summary</th>
<th>Equivalent English Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extractive</td>
<td>திருச்சிராப்பள்ளி, தமிழ்நாடு மாநிலத்தில் வருமாறு மாநிலத்தில் அமைந்துள்ளது. கோயம் பட்டனாட்சியும் பயணத் தொழில்நுட்பத்துடன் கூடுதலாக இருந்தது. கோயம் பட்டனாட்சியும் மேற்குத் தொழில்நுட்பத்துடன் கூடுதலாக இருந்தது. திருச்சிராப்பள்ளி என்னும் இடத்தின் வருந்தை தொடர்பான பொறியியல்வாய்ப்பு மற்றும் தொழில்நுட்பத்துடன் கூடுதலாக இருந்தது. மாத்திரமே மேற்குத் தொழில்நுட்பத்துடன் கூடுதலாக இருந்தது. திருச்சிராப்பள்ளி என்னும் இடத்தின் வருந்தை தொடர்பான பொறியியல்வாய்ப்பு மற்றும் தொழில்நுட்பத்துடன் கூடுதலாக இருந்தது.</td>
<td>Tiruchirappalli, Tamil Nadu state of India situated in. Located on the banks of Cauvery in Tiruchirappalli in Tamil Nadu is one of the four most important cities. Trichy Rockfort found in the cave 'Chira' the name of the Jain monk in the eleventh century inscription says akkukai stay and to be meditated. Tiruchirappalli, Tamil Nadu is one of the people who lived in the most ancient cities. 6th-century cave temples in South India Rockfort ruled Mahendravarma first built. The fourth largest metropolitan area in Tiruchirappalli, Tamil Nadu, who is ranked 47 in intiyala, Vayalpakkutkal located on the Kaveri. Kaveri araiyottiya the rich alluvial soil with six of its branch river kollitam. Trichy is the terminus of the engineering equipment manufacturer in Tamil Nadu. Trichy in Tamil Nadu, is the center, close to the population, the city bus stops and changes were made to reduce traffic congestion.</td>
</tr>
<tr>
<td>Summary</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Methods</td>
<td>Tamil Summary</td>
<td>Equivalent English Summary</td>
</tr>
<tr>
<td>---------</td>
<td>--------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>Proposed Approach (Abstractive Summary)</td>
<td>திறந்துசெய்யப்பட்டன, அருகியர்கள் அவ்வாறும் பாதுகாப்பு கற்றுள்ளன அம்பதிகளுக்கு, திறந்துசெய்யப்பட்டிட்டு. திறந்துசெய்யப்பட்டிட்டு கற்றுள்ளன அம்பதிகளுக்கு</td>
<td>Tiruchirappalli, Tamil Nadu state of India located on the banks of the Cauvery. Tiruchirappalli is also called as Trichy. The kallanai cemetery was built by the Chola is in Tiruchirappalli. There are temples such as Tayumana Swami Temple, utchi Ganesh Temple, Marianman Temple, arankanata Swami Temple, vekkali Amman Temple, campuccevarar Temple, Subramanya Swamy Temple, pancavarna Swami Temple, anranesvara Swami Temple, uttamar Temple, Prasanna Venkatachalapathy Temple, nilivanesvar temple and shrines. Rockfort, Srirangam, Thiruvanaikovil, mukkompu, stone, Vayalur Murugan Temple, Ganga konda Cholapuram tourist attractions which can be found. He goes to Chennai from 16 buses daily.</td>
</tr>
</tbody>
</table>

279
7.7 CONCLUSION

In this work, we have proposed graph based, single-document and query focused multi-document summarization task. In a single-document summarization, the bootstrapping technique is first applied to determine the important components for the summary, where various graph operations have been assigned to each of the nodes and edges. Then, the edge integrated spreading activation algorithm is applied over the entire document semantic graph to execute the graph operations assigned by the bootstrapping. The system is evaluated using the FIRE-2010 dataset and ROUGE scores. The proposed system produces 67% accuracy when compared against the summarization systems based on term frequency, concept frequency and original spreading activation.

In a query focused multi-document summarization, various redundancies have been eliminated at two stages. During the first stage, the redundant information is eliminated, while building the global semantic graph and during the second stage, the redundancies are eliminated after the construction of the global semantic graph. The elimination process has been carried out, using various graph matching procedures, with the word based and context based semantics. We then used the pattern-based bootstrapping approach for learning the graph operations that can be assigned to the nodes and edges of the global semantic graph. To obtain the summary based on the user’s interest, we modify the spreading activation procedure, by integrating the query graph while spreading starts the activation on multiple nodes and edges of the global semantic graph. The proposed approach is evaluated, investigated and compared with different summarization methods, such as the template based summary and extractive summary. We have also compared our approach with the baseline system and measure the accuracy with various ROUGE scores. From the analysis, our approach produces a better query focused summary when compared to the other systems.