Dense Semantic Graph and its Application in Single Document Summarisation

Monika Joshi[1], Hui Wang[1] and Sally McClean[2]

[1] University of Ulster, Co. Antrim, BT37 0QB, UK
joshi-m@email.ulster.ac.uk , H.Wang@ulster.ac.uk
[2] University of Ulster, Co. Londonderry, BT52 1SA, UK
sally@infc.ulst.ac.uk

Abstract Semantic graph representation of text is an important part of natural language processing applications such as text summarisation. We have studied two ways of constructing the semantic graph of a document from dependency parsing of its sentences. The first graph is derived from the subject-object-verb representation of sentence, and the second graph is derived from considering more dependency relations in the sentence by a shortest distance dependency path calculation, resulting in a dense semantic graph. We have shown through experiments that dense semantic graphs gives better performance in semantic graph based unsupervised extractive text summarisation.

1 Introduction

Information can be categorized into many forms -- numerical, visual, text, and audio. Text is abundantly present in online resources. Online blogs, Wikipedia knowledge base, patent documents and customer reviews are potential information sources for different user requirements. One of these requirements is to present a short summary of the originally larger document. The summary is expected to include important information from the original text documents. This is usually achieved by keeping the informative parts of the document and reducing repetitive information.

There are two types of text summarization: multiple document summarisation and single document summarization. The former is aimed at removing repetitive content in a collection of documents. The latter is aimed at shortening a single document whilst keeping the important information. Single document summarisation is particularly useful because large documents are common especially in the digital age, and shortening them without losing important information is certain to save time for the users/readers. The focus of our research is on single document summarisation. In order to process a text document, it should be broken down into parts and then represented in a suitable form to facilitate analysis. Various text representation schemes have been studied, including n-gram, bag of words, and graphs. In our research we use graphs to represent a text document. The graph is constructed by utilising semantic relations such as dependency relations between words within the sentence.

We propose a novel graph generation approach, which is an extension of an existing semantic graph generation approach [4] by including more dependencies from dependency parsing of the sentence. This results in dense semantic graph. We evaluated both graphs in a text summarisation task through experiments. Results show that our dense semantic graph outperformed the original semantic graph for unsupervised extractive text summarization.

The next section gives a short literature review of the earlier graph based approaches to text summarisation. In section 3, a detailed description is provided concerning the construction of two different semantic graphs that were used in our study. Section 4 discusses extractive summarisation based on these semantic graphs and section 5 describes the experiments and results. After that conclusion of the analysis follows.

2 Previous Work on Graph based Text Summarisation

Earlier researchers have used graph representation of documents and properties of graphs to extract important sentences from documents to create a short summary. Graph based text summarisation methods such as LexRank [1], TextRank [2] and Opinosis [3] have shown good performance. There are two types of graph that are constructed and used to represent text. Lexical graph uses the lexical properties of text to construct a graph. LexRank and Text Rank are lexical graph based approaches. They construct graphs by connecting two sentences/smaller text units as nodes in the graph based on the degree of content overlap between them.

On the other hand, semantic graph is based on semantic properties of text. Semantic properties are: Ontological relationship between two words such as synonymy, hyponymy; relationship among set of words representing the syntactic structure of sentence such as dependency tree and syntactic trees. A set of words along with the way they are arranged provides meaning. The same set of words connected in different ways gives different meaning.

According to the semantic properties utilised for graph construction, various representations have been reported in literature for semantic graphs [4, 5]. Some of the approaches utilize the lexical database WordNet to generate ontological relations based semantic graph. In this sentences are broken into terms, mapped to WordNet synsets and connected over WordNet relations [6]. In one of the approaches called semantic Rank [7], sentences are connected as nodes and the weight of the edges between them is the similarity score calculated by WordNet and Wikipedia based similarity measures. Other approaches to generate semantic graphs try to utilize the dependency relations of words in a sentence along with the ontological relations between words. Utilizing this particular order of connection also forms the basis of research work done on semantic graphs in our study. In this area of semantic graph generation most of the work has been concentrated on identifying logical triples (subject-object-predicate) from a document and then connecting these triples based on various semantic similarity measures [4]. Predicate (or verb) is the central part of any sentence, which signifies the main event happening within the sentence. Thus it was

mostly agreed to consider the verb and its main arguments (subject and object) as the main information presented in the sentence, and use this as a basic semantic unit of the semantic graph. Various researches have been done on this graph in the field of supervised text summarisation.

We have evaluated two semantic graphs which are based on the dependency structure of words in a sentence. The first graph is triple(subject-object-verb) based semantic graph proposed by Leskovec et al [4]. The second graph is a novel approach of semantic graph generation proposed in this paper, based on the dependency path length between nodes. Our hypothesis is that moving to a dense semantic graph, as we have defined it, is worthwhile. The principle idea behind this new graph has been used in earlier research in kernel based relation identification [8]. However it has not been used for construction of a semantic graph for the complete document. The next section describes more details about this graph.

3 Semantic Graphs

In the research carried out in this paper, we have analysed the difference between performances when more dependency relations than just subject-object-verb are considered to construct a semantic graph of the document. In this direction, we have developed a methodology to select the dependencies and nodes within a shortest distance path of dependency tree to construct the semantic graph. First we will describe the previous use of graphs and then we will introduce the graph generated by our methodology.

3.1 Semantic graph derived from a triplet (Subject-Object-verb)

Leskovec et al. [4] has described this graph generation approach for their supervised text summarization, where they train a classifier to learn the important relations between the semantic graph of a summary and the semantic graph of an original text document. In this graph the basic text unit is a triple extracted from sentence: subject-verb-object. This is called triple as there are three connected nodes. Information such as adjectives of subject/object nodes and prepositional information (time, location) are kept as extra information within the nodes. After extracting triples from every sentence of the text document two further steps are taken: i. co-reference and anaphora resolution: all references to named entities (Person, Location etc.) and pronoun refeences are resolved. ii. Triples are connected if their subject or object nodes are synoymous or referring to the same named entity. Thus a connected semantic graph is generated.

3.2 Dense Semantic graphs generated from shortest dependency paths between Nouns/Adjectives

We have observed that various named entities such as location/time which are important information, are not covered in the subject-predicate-object relations. As this

information is often added through prepositional dependency relations, it gets added to nodes as extra information in the semantic graph generated by previous approaches. However these named entities hold significant information to influence ranking of the sentences for summary generation and to connect nodes in the semantic graph. This has formed the basis of our research into new way of semantic graph generation. First we elaborate the gaps observed in previous approach of semantic graph generation and then give the details of the new semantic graph.

Gaps identified in triple (subject-object-verb) based semantic graph.

The kind of information loss observed in the previous semantic graphs has been described below:

- Loss of links between words in sentence
  Some connections between named entities are not considered because they do not come into the subject/object category. This information is associated with subject/object, but does not get connected in the semantic graph, as they are not directly linked through a predicate. For example consider the sentence below:

  President Obama's arrival in London created a joyful atmosphere.
  The triple extracted from this sentence is:
  Arrival->create->atmosphere

  Here the information *London*, *Obama* is added as extra information to node *Arrival*, and *Joyful* is added to node *Atmosphere*. However a direct link between *London* and *atmosphere* is missing, whereas a reader can clearly see this is atmosphere of London. This connection can be identified in our shortest dependency path graph as shown below:

  *London-**prep-in**->Arrival-**nsubj**->created-**dobj**->atmosphere*

- Loss of inter-sentence links between words

  Some named entities which are not subject/object in one sentence are subject/object of another sentence. When creating a semantic graph of complete document, these entities are the connecting words between these sentences. In the previous graph these connections are lost as shown below by two sentences.

  He went to church in Long valley.
  One of the explosions happened in Long Valley.

  The triple extracted from these sentences is:

  He->went>church
  Explosion->happened->long valley

In the semantic graph derived from triples of the above 2 sentences, we do not have both these sentences connected, because the common link *Long Valley* is hidden as extra information in one semantic graph.

- Identification of subject is not clear

  In a few cases, identification of a subject for the predicate is not very accurate with current dependency parsers. This case occurs in the clausal complement of verb phrase or adjectival phrases called dependency relation "xcomp". Here the determination of subject for clausal complement is not very accurate, as the subject is external.

Construction of shortest distance dependency path based semantic graph

To overcome these gaps, we construct the semantic graph by connecting all noun and adjectives which are connected within a shortest path distance in the dependency tree of that sentence. From the literature review it has been identified that nouns are the most important entities to be considered for ranking sentences. So we have decided to include nouns as nodes in the semantic graph. We also considered adjectives, as they modify nouns and may present significant information. The length of the shortest path is varied from 2-5 to analyse its effect on the efficiency of the PageRank score calculation. The following steps are followed to construct the semantic graph

- Co-reference resolution of named entities
  The text document is preprocessed to resolve all co-references of named entities. We replace the references with the main named Entity for Person, Location, and organization.

- Pronominal resolution
  After co-reference resolution, text is preprocessed for pronominal resolution. All reference (he, she, it, who) are resolved to referring named entities and replaced them in text.

- Identifying nodes and edges of the semantic graph

  The shortest path distance based Semantic graph is defined as G= (V, E), Where

$$V = \left\{ \bigcup_{word_i \in document} Word_i : \text{pos}(Word_i) \in \{\text{JJ}*, \text{NN}*\} \right\} \quad (1)$$

In (1) $\text{pos}(Word_i)$ provides part of the speech tag of $Word_i$. According to Penn tag set for part of speech tags, "JJ" signifies Adjectives and "NN" signifies Noun.

$$\text{Edge set } E = \{\bigcup_{u,v \in V}(u, v) : SD(u, v) \leq limit\} \quad (2)$$

In (2) $SD(u, v)$ is the shortest distance from u to v in the dependency tree of that sentence and *limit* is the maximum allowed shortest path distance, which is varied from 2-5 in our experiments.

We have used Stanford CoreNLP package for co-reference resolution, identification of named entities and dependency parse tree generation[9] [10]. To develop the graphs and calculate the page rank scores of nodes we use the JUNG software package[1]. First we extract dependency relations for each sentence. Then we generate a temporary graph for the dependency tree of that sentence in JUNG .Then Dijkstra's shortest path algorithm is applied to find the shortest distance between nodes. From this temporary graph we find vertices and edges based on equations (1) and (2) to construct the semantic graph.

Fig. 1 and 2 show two graphs, triple based semantic graph and shortest distance dependency path based semantic graphs for the given excerpt of 2 sentences below, taken from the *Long Valley* document of DUC2002 data.

A text excerpt taken from DUC 2002 data.

The resort town's 4,700 permanent residents live in Long Valley, a 19-mile-long, 9-mile-wide volcanic crater known as a caldera. Eruptions somewhat smaller than Mount St. Helens' happened 550 years ago at the Inyo craters, which span Long Valley's north rim, and 650 years ago at the Mono craters, several miles north of the caldera.

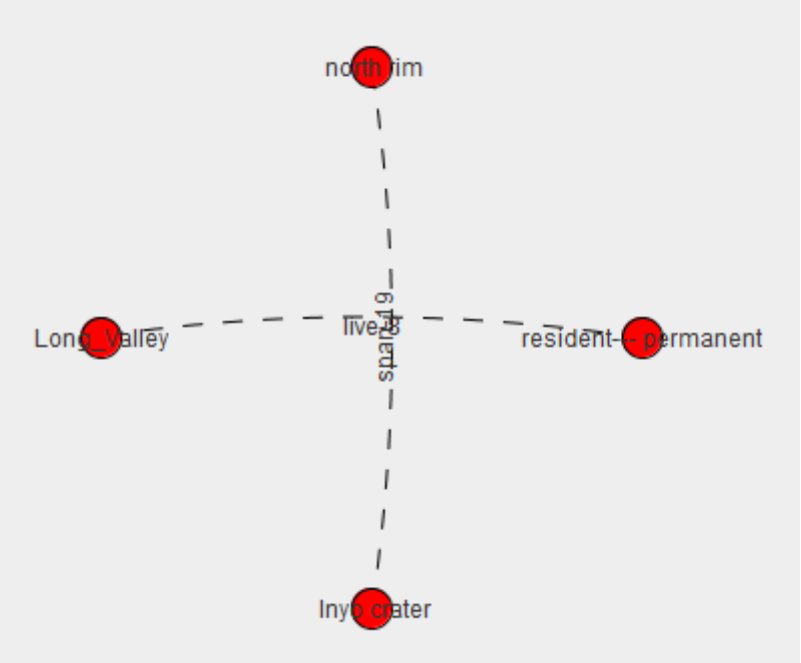

Fig. 1. The triple based semantic graph for the text excerpt taken from DUC 2002 data

[1] http://jung.sourceforge.net/

The next section describes the methodology to rank sentences based on the semantic graph described in this section.

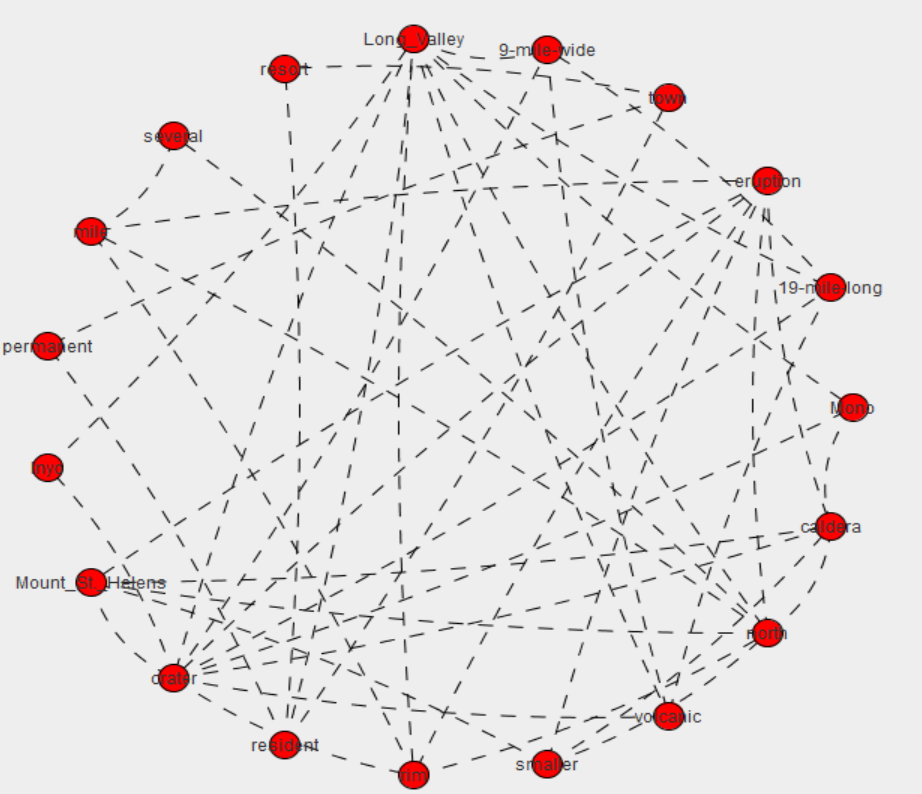

Fig. 2. Sematic graph based on the shortest dependency path between nouns/adjectives (shortest distance=2) for the text excerpt taken from the DUC 2002 data

4 Extraction of Sentences for Summary Generation

In this paper we want to analyse the impact of dense semantic graphs on text summarisation and provide a comparison with the summarisation results of earlier triple based semantic graphs. To achieve this, first we rank the semantic graph by one of the graph ranking algorithm. We have used PageRank method to rank the semantic graph nodes.

The PageRank score of $node_i$ is calculated as:

$$PageRank(node_i) = (1-d) + d * \sum_{node_j \in In(node_i)} \frac{PageRank(node_j)}{|Out(node_j)|} \quad (3)$$

Where *d* is the probability of jumping from $node_i$ to any random node in the graph, typically set between 0.1-0.2. $In(node_i)$ is the set of incoming edges to $node_i$ and $Out(node_j)$ is the set of outgoing edges of $node_j$. Initially PageRank of all nodes is intialised with arbitrary values, as it does not affect the final values after convergence. In this paper semantic graphs are undirected graphs so incoming edges of a node are equal to outgoing edges.

After calculating PageRank score of the nodes in the semantic graph, the score of sentence S_i in the text document is calculated by following equation:

$$Score_{S_i} = \sum_{node_j \in graph \cap S_i} PageRank(node_j) \tag{4}$$

where $node_j$ is the stemmed word/phrase in the graph representation. Scores are normalised after dividing by the maximum score of sentences. After calculating normalized scores of all sentences in the text document, sentences are ordered according to their scores. As per the summary length, higher scoring sentences are taken as summary sentences.

In addition to this summary generation method, we have also tried to analyze impact of including additional features together with PageRank scores on semantic graph based text summarisation. This was done in a separate experimental run where we have included sentence position as an additional feature for scoring of sentences. Since the data we have experimented with is news data, a higher score is given to early sentences of the document. So the score of a sentence S_i after including sentence position, *i* as a feature is given by:

$$newScore_{S_i} = 0.1 \times (Count_{sentences} - i) \Big/ Count_{sentences} + 0.9 \times Score_{S_i} \tag{5}$$

After calculating the new score of the sentences, higher scoring sentences are extracted as the summary as in previous summarisation method. The next section describes the experimental setup.

5 Experiments

We have experimented on two single document summarisation corpuses from Document Understanding Conference (DUC), DUC-01 and DUC-02.

DUC-01 contains 308 text documents and DUC-02 contains 567 text documents. Both sets have 2 human written summaries per document for evaluation purposes. We have used the ROUGE toolkit to evaluate system generated summaries with reference summaries, that are 2 human generated summaries per document [11]. The ROUGE toolkit has been used for DUC evaluations since the year 2004. It is a recall oriented evaluation metric which matches n-grams between a system generated summary and reference summaries.

$$\text{Rouge} - \text{N} = \frac{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{ReferenceSummaries\}} \sum_{gram_n \in S} Count(gram_n)} \tag{6}$$

Rouge-1 is 1-gram metric. Rouge-2 is 2-gram metric. Rouge-W is the longest weighted sequence metric, which gives weight to consecutive longest sequence matches.

ROUGE scores were calculated for different summarisation runs on triple based semantic graphs and shortest dependency distance path based semantic graphs. On triple based graphs two summarisation tasks were run for DUC01 and DUC-02 data. The first considered PageRank only and the second used PageRank, sentence position (Triple based, Triple + position). On the Shortest distance dependency path based semantic graph, 6 summarisation tasks were run for both datasets. The first 4 runs are based on PageRank scores alone by varying shortest distance from 2-5: shortest distance 2 (SD-2), shortest distance 3 (SD-3), shortest distance 4 (SD-4) and shortest distance 5 (SD-5). The fifth and sixth run include sentence position as feature with SD-4 and SD-5(SD-4 + position, SD-5+ position). We have also compared our results with the results of the text summarisation software *Open Text Summarizer(OTS)* [12], which is freely available and has been reported to perform best between other available open source summarizers.

6 Results and Analysis

Figure 3 shows the ROUGE-1, ROUGE-2 and ROUGE-W scores for DUC-01 data achieved by different experimental runs described in section 5. The Rouge evaluation setting was a 100 words summary, 95% confidence, stemmed words and no stop words included during summary comparison.

In figure 3, we have observed that the lowest Rouge scores are reported with the triple based experiment. By including position, results for triple based experiment are improved. Rouge-1 scores for SD-2, SD-3, SD-4, and SD-5 improves systematically and are better than triple based and triple based + position. This shows that as the shortest length of dependency path was increased from 2 to 5, the Rouge score has improved due to better ranking of the nodes in the semantic graph. This better ranking can be attributed to more connections found after increasing the path distance to find links in the dependency tree. A similar trend of increase in ROUGE-2 and ROUGE-W scores are observed for experiments on DUC-02 data in SD-2, SD-3, SD-4, SD-5, SD-4+position, and SD-5+position.

Although benchmark OTS results are always higher than best results achieved by our approach, it is useful to observe that our results are comparable to the benchmark results, as the main purpose of our research is to analyse the impact of dense semantic graphs on text summarisation compared to previous semantic graph. Table I gives results in a numerical form for the DUC-01 experiments. Figure 4 and Table II shows the scores for the DUC 02 dataset. For both corpuses the ROUGE scores improves on shortest dependency based graph, until distance 5. During results analysis we have observed that the ROUGE score decreases or becomes approximately constant if we increase distance after 5.

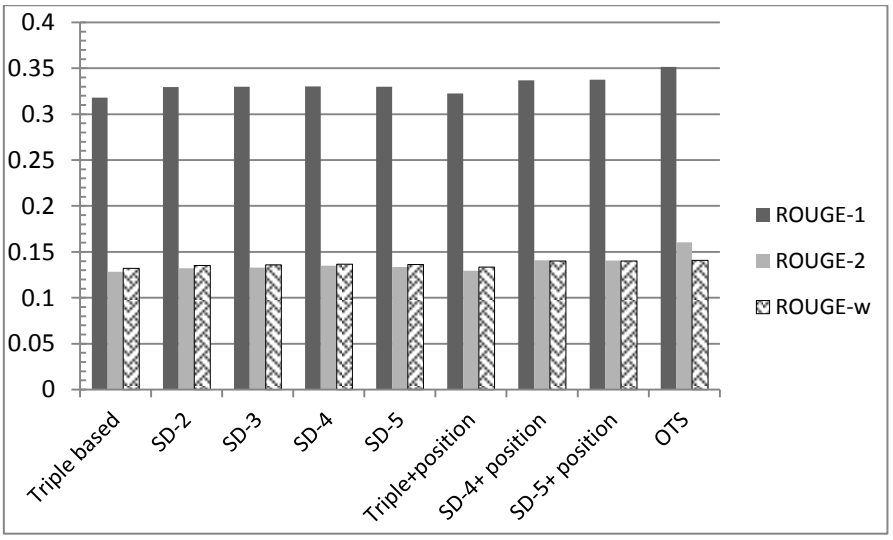

Fig. 3. ROUGE scores obtained for a summarisation test on DUC-01 data

Table 1. ROUGE scores for a summarisation test on DUC-01 data

System	Rouge-1	Rouge-2	Rouge-W
Triplet based	0.31793	0.12829	0.13214
SD-2	0.32964	0.13229	0.1354
SD-3	0.3298	0.13301	0.1359
SD-4	0.33037	0.1351	0.13671
SD-5	0.32974	0.13365	0.13621
Triple + position	0.3224	0.12923	0.13355
SD-4+ position	0.33676	0.14106	0.14017
SD-5+ position	**0.33753**	**0.14049**	**0.14023**
OTS	**0.35134**	**0.16039**	**0.14093**

Including sentence position as a feature, improves the summarisation results on both triple based graph and shortest distance dependency path based semantic graph. Also in this case, ROUGE scores for summarisation run on shortest distance dependency path based semantic graph are higher than for triple based semantic graphs. This also indicates that we can include more features to improve the results further. Overall results indicate that shortest distance based semantic graphs performs better in ranking the sentences and are comparable to benchmark system OTS.

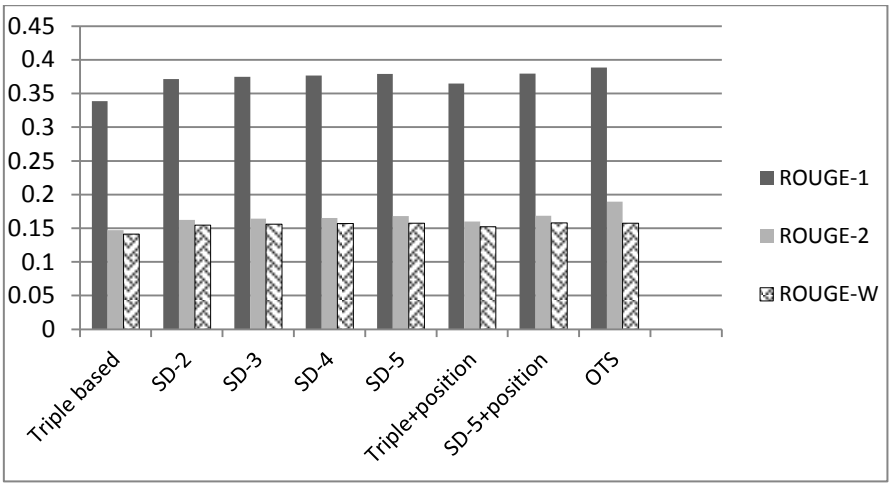

Fig. 4. ROUGE scores obtained for summarisation test on DUC-02 data

Table 2. ROUGE scores for summarisation test on DUC-02 data

System	Rouge-1	Rouge-2	Rouge-W
Triplet based	0.33864	0.14714	0.14143
SD-2	0.37154	0.16221	0.15465
SD-3	0.37494	0.16409	0.1563
SD-4	0.37666	0.16498	0.15694
SD-5	0.37919	0.168	0.15778
Triple + position	0.36465	0.16016	0.15231
SD-4+ position	0.37666	0.16498	0.15694
SD-5+position	**0.37937**	**0.16846**	**0.15793**
OTS	**0. 0.38864**	**0.18966**	0.15766

7 Conclusion

PageRank based summarisation is a novel approach for both our approaches. Earlier for triple based semantic graph, PageRank node score was considered as a feature for supervised text summarisation. In this paper we have looked at unsupervised single document summarisation. In the evaluation, we have seen that only PageRank based summarisation results do not exceed the benchmark results, but are comparable. Benchmark OTS system utilises a language specific lexicon for identifying synonymous words and cue terms. In future work, we can include a similar lexicon to identify more relation between words to improve the performance. In this paper we have hypothesised that if more dependency relations are considered for semantic graph generation it gives better PageRank scores and thus improves the ranking accuracy for

extraction of summary sentences. Although triple based graphs are more visually understandable they can be enhanced by adding more dependencies. When sentence position was included as an extra feature, it improved the Rouge scores. Also it is noticeable that summarisation results for shortest distance dependency path based semantic graph are similar to results after including the additional feature *sentence position*. This makes this graph equally useful in domains where sentence position does not have an effect on importance.

In future work we will apply semantic similarity and word sense disambiguation to improve the connectivity of the graph and identify more relations between nodes.

References

1. G. Erkan and D. R. Radev, "LexRank: graph-based lexical centrality as salience in text summarization," *Journal of Artificial Intelligence Research*, vol. 22, no. 1, pp. 457–479, Jul. 2004.
2. R. Mihalcea and P. Tarau, "TextRank: Bringing Order into Texts," in *Proceedings of Empirical Methods in Natural Language Processing*, 2004.
3. K. Ganesan, C. Zhai, and J. Han, "Opinosis : A Graph-Based Approach to Abstractive Summarization of Highly Redundant Opinions," in *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010, no. August, pp. 340–348.
4. J. Leskovec, N. Milic-Frayling, and M. Grobelnik, "Extracting Summary Sentences Based on the Document Semantic Graph," *Microsoft Technical Report TR-2005-07*, 2005.
5. D. Rusu, B. Fortuna, M. Grobelnik, and D. Mladenić, "Semantic Graphs Derived From Triplets With Application in Document Summarization," *Informatica Journal*, 2009.
6. L. Plaza and A. Díaz, "Using Semantic Graphs and Word Sense Disambiguation Techniques to Improve Text Summarization," *Procesamiento del Lenguaje Natural Revista*, vol. 47, pp. 97–105, 2011.
7. G. Tsatsaronis, I. Varlamis, and K. Nørvåg, "SemanticRank: ranking keywords and sentences using semantic graphs," *COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics*, no. August, pp. 1074–1082, 2010.
8. R. C. Bunescu and R. J. Mooney, "A Shortest Path Dependency Kernel for Relation Extraction," in *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, 2005, no. October, pp. 724–731.
9. K. Toutanova, D. Klein, C. D. Manning, and Y. Singer, "Feature-rich part-of-speech tagging with a cyclic dependency network," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - NAACL '03*, 2003, vol. 1, pp. 173–180.
10. H. Lee, Y. Peirsman, A. Chang, N. Chambers, M. Surdeanu, and D. Jurafsky, "Stanford's multi-pass sieve coreference resolution system at the CoNLL-2011 shared task,", Jun. 2011.
11. C. Lin and M. Rey, "R OUGE : A Package for Automatic Evaluation of Summaries," in *Proceedings of the ACL-04 Workshop: Text Summarization Branches Out*, 2004, pp. 74–81.
12. V. a. Yatsko and T. N. Vishnyakov, "A method for evaluating modern systems of automatic text summarization," *Automatic Documentation and Mathematical Linguistics*, vol. 41, no. 3, pp. 93–103, Jun. 2007.