Genetic Semantic Graph Approach for Multi-document Abstractive Summarization

Atif Khan
Faculty of Computing
Universiti Teknologi Malaysia
Johor, Malaysia
atifkhan@icp.edu.pk

Naomie Salim
Faculty of Computing
Universiti Teknologi Malaysia
Johor, Malaysia
naomie@utm.my

Yogan Jaya Kumar
Faculty of Information and
Communication Technology
Universiti Teknikal Malaysia
Melaka, Malaysia
yogan@utem.edu.my

***Abstract*—The aim of automatic multi-document abstractive summarization is to create a compressed version of the source text and preserves the salient information. Existing graph based summarization methods treat sentence as bag of words, rely on content similarity measure and did not consider semantic relationships between sentences. These methods may fail in determining redundant sentences that are semantically equivalent. This paper introduces a genetic semantic graph based approach for multi-document abstractive summarization. Semantic graph from the document set is constructed in such a way that the graph nodes represent the predicate argument structures (PASs), extracted automatically by employing semantic role labeling (SRL); and the edges of graph correspond to semantic similarity weight determined from PAS-to-PAS semantic similarity, and PAS-to-document set relationship. The PAS-to-document set relationship is represented by different features, weighted and optimized by genetic algorithm. The salient graph nodes (PASs) are ranked based on modified graph based ranking algorithm. In order to reduce redundancy, we utilize maximal marginal relevance (MMR) to re-ranks the PASs and use language generation to generate summary sentences from the top ranked PASs. Experiment of this study is carried out using DUC-2002, a standard corpus for text summarization. Experimental results reveal that the proposed approach performs better than other summarization systems.**

Keywords—semantic role labeling; semantic graph; semantic similarity measure; genetic algorithm; abstractive summarization

I. Introduction

Nowadays, users of internet are flooded with the huge amount of textual data due to the explosive growth of information overload over the internet. The current age of information overload demands need of text summarization, which is a significant and timely tool for user to speedily comprehend the massive volume of information. Multi-document text summarization aims to select the most significant information from the given source documents and produce a concise summary that can satisfy user's needs [1]. It assists online users to obtain information efficiently. It has attracted more attention in diverse fields of research such as information retrieval, natural language processing and machine learning [2]. Text summarization approaches can be broadly separated into two groups: extractive summarization and abstractive summarization. Extractive summarization extracts the most important representative sentences from the source documents and group them to produce a summary. However, abstractive summarization requires natural language processing techniques such as semantic representation, natural language generation, and compression techniques. Abstractive summarization aims to interpret and examine the source text and creates a concise summary that usually contain compressed sentences or may contain some novel sentences not present in the original source text [1].

Most of the studies have focused on multi-document extractive summarization using techniques of sentence extraction [3], statistical analysis [4], discourse structures and various machine learning techniques [5, 6]. Different graph-based methods [7-10] have also been investigated for multi-document extractive summarization. However, abstractive summarization is a challenging area for researchers. To date, a few research efforts have been done in this direction. A particular challenge for multi-document summarization is that topically related documents usually contain overlapping information. Thus, suitable summarization methods are required to merge similar information content across several documents [11]. Specifically, the aforementioned graph based methods attempted for multi-document extractive summarization, treat sentence as bag of words and did not take into account the semantic structure of sentence and semantic relationships between sentences. These methods determine sentence similarity by utilizing content similarity measure, which may not able to identify redundant sentences that are semantically equivalent. Thus, the final summary would contain redundant information.

To our knowledge, genetic semantic graph based approach has not been investigated for multi-document abstractive summarization (MDAS). Therefore, this study aims to propose a genetic semantic graph based approach for MDAS, which will automatically merge similar information across the documents, and employs language

ISBN: 978-1-4673-6832-2©2015 IEEE

generation to generate abstractive summary. The approach constructs semantic graph from the document text in such a way graph nodes represent the predicate argument structures (PASs), extracted automatically by employing semantic role labeling (SRL), and the edges of graph correspond to semantic similarity weight determined from PAS-to-PAS semantic similarity, and PAS-to-document set relationship. The PAS-to-document set relationship is represented by different features, weighted and optimized by genetic algorithm. The salient graph nodes (PASs) are ranked based on modified graph based ranking algorithm. In order to reduce redundancy, we apply MMR to re-rank PASs. The top ranked PASs are chosen based on compression rate of summary and are fed to language generation phase to generate summary. Our contributions are summarized as follows:

- Propose a semantic graph approach for multi-document abstractive summarization.
- Modify graph based ranking algorithm to take into account PAS-to-PAS semantic similarity and PAS-to-document set relationship. Different features (weighted and optimized by genetic algorithm) are employed to represent PAS-to-document set relationship.
- Examine Jiang semantic similarity measure to detect redundancy by capturing semantically similar predicate argument structures (PASs).
- To evaluate the proposed semantic graph based approach with Pyramid and ROUGE evaluation measures on DUC 2002 multi-document summarization shared tasks.

The rest of this paper is organized as follows: Section II demonstrates the related work to this research study. Section III outlines the proposed approach. Section IV presents the evaluation results and Section V discusses the results. Finally we end with conclusion in Section VI.

II. RELATED WORK

Limited research studies have dealt with multi-document abstractive summarization. Two mainstream approaches are applied to multi-document abstractive summarization: linguistic and semantic based approaches. Linguistic based approaches proposed for abstractive summarization employ tree based method [11, 12], lead and body phrase method [13] and information item based method [14]. All linguistic based approaches rely on syntactic representation of the source document, and therefore the general limitation of these approaches is the lack of semantic representation of source text. On other hand, different semantic based approaches have also been introduced for abstractive summarization such as template based methods [15, 16] and ontology based methods [17-19]. The obvious drawback of template based methods is that linguistic patterns and extraction rules for template slots are manually created by humans, which is time consuming. Moreover, these methods could not handle similar information across multiple documents. Moreover, the ontology based methods heavily rely on domain expert to build domain ontology, which require more effort and time, and these methods are not applicable to other domains. A series of analysis studies is performed by [20] to compare human-written model summaries with system summaries at semantic level of caseframes. However, these studies did not propose any summarization model. In recent years, different graph based approaches [7-10, 21-24] have been employed for multi-document extractive summarization. These methods use PageRank algorithm [25] or its variants to rank sentences or passages. However, these approaches treat sentence as bag of words and did not consider the semantic structure of sentence i.e. predicate argument structure. Moreover, these approaches rely on content similarity measure and did not consider semantic relationships between sentences while computing the salience score of sentences. These approaches may fail to detect redundant sentences that are semantically equivalent, and therefore the final summary would be inadequate. The only graph based approach introduced for abstractive summarization [19] constructs semantic graph from manually built ontology. This approach heavily relies on human expert and is limited to single document. The next section explains the proposed approach.

III. PROPOSED APPROACH

The architecture of proposed approach is given in Fig.1.

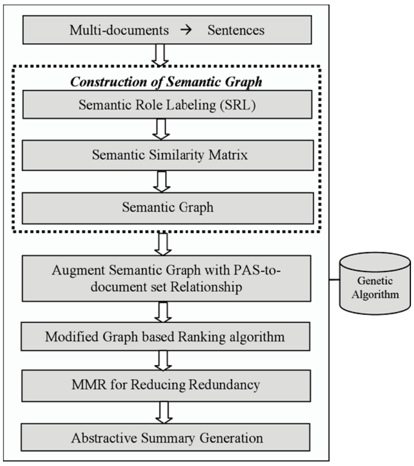
Multi-documents → Sentences
Construction of Semantic Graph
Semantic Role Labeling (SRL)
Semantic Similarity Matrix
Semantic Graph
Augment Semantic Graph with PAS-to-document set Relationship
Genetic Algorithm
Modified Graph based Ranking algorithm
MMR for Reducing Redundancy
Abstractive Summary Generation

Fig. 1. Proposed genetic semantic graph approach

A. Semantic Role Labeling

The goal of this step is to extract predicate argument structure from each sentence in the document set. At first, we split the document set into sentences in such a way that each sentence is preceded by its corresponding document number and sentence position number. Since abstractive summarization requires deep semantic analysis, therefore SENNA semantic role labeler [26] is employed to parse each sentence and properly labels the semantic word phrases. These phrases are referred to as semantic arguments. The semantic arguments can be grouped in two categories: core arguments *(Arg)* and adjunctive arguments *(ArgM)* [1]. In this study, we consider *A0* for subject, *A1* for object, *A2* for indirect object as core arguments, and *ArgM-*

LOC for location, *ArgM-TMP* for time as adjunctive arguments for predicate (Verb) *V*. We consider all the complete predicates associated with the single sentence structure in order to avoid loss of important terms contributing to the meaning of sentence and the actual predicate of the sentence. We assume that predicates are complete if they have at least two semantic arguments. A sentence containing one predicate is represented by simple predicate argument structure, while a sentence containing more than one complete predicate is represented by a composite predicate argument structure.

Example 1: Consider the following two sentences represented by simple predicate argument structures.

S_1: Eventually, a huge cyclone hit the entrance of my house.
S_2: Finally, a massive hurricane attack my home

The corresponding simple predicate argument structures P_1 and P_2 are obtained after applying semantic role labeling to sentences S_1 and S_2:

P_1: [*AM-TMP*: Eventually] [*A0*: a huge cyclone] [*V*: hit] [*A1*: the entrance of my house]

P_2: [*AM-DIS*: Finally] [*A0*: a massive hurricane] [*V*: attack] [*A1*: my home]

Once the predicate argument structures (PASs) are obtained, they are split into meaningful words or tokens, followed by removal of stop words. The remaining words in PASs are stemmed to their base form using porter stemming algorithm [27]. Next, we employ SENNA POS tagger [26] to label each term of semantic arguments (associated with the predicates), with part of speech (POS) tags. The POS tags *NN* stands for noun, *V* for verb, *JJ* for adjective and *RB* for adverb etc. This study compares predicate argument structures based on noun-noun, verb-verb, location-location and time-time arguments. Therefore, we extract only tokens from predicate argument structures, which are labeled as noun, verb, location, and time. All the PASs associated with the sentence will be included in comparison. Once the nouns, verbs, and other arguments (time and location) if exist, are extracted, the predicate argument structures obtained in Example 1 after further processing will become as follows:

P_1: [*A0*: hurricane *NN*] [*V*:attack] [*A1*: home (*NN*)]

P_2: [*AM-TMP*: Eventually (*RB*)] [*A0*: cyclone (*NN*)] [*VBD*: hit] [*A1*: entrance (*NN*), house (*NN*)]

B. *Semantic Similarity Matrix*

The goal of this step is to construct a matrix of semantic similarity scores for each pair of predicate argument structure. The idea of similarity matrix, which is closely related to the concept of "homophily", has also been employed in the social network analysis [28, 29]. In this step, similarity of the predicate argument structures (PASs) is computed pair wise based on acceptable comparisons of noun-noun, verb-verb, location-location and time-time. Based on experimental results in the literature [30], Jiang and Conrath measure has the closest correlation with human judgment amongst all the semantic similarity measures. Therefore, this study exploits Jiang's semantic similarity measure [31] for computing semantic similarity between each pair of PASs. Jiang‘s measure is information content based measure and consider that each concept in the WordNet [32] hold certain information. According to this measure, the similarity of two concepts is dependent on the information that the two concepts share. Jiang's measure [31] calculates the semantic distance to obtain semantic similarity between any two concepts is given as follows:

$$Jiang_{dist}\left(C1,C2\right)=IC\left(C1\right)+IC\left(C2\right)-2xIC\left(lso\left(C1,C2\right)\right) \quad (1)$$

First, Jiang's measure uses WordNet to compute the least common subsumer (**lso**) of two concepts, which is the closest shared parent of the two concepts, then determines $IC(C1), IC(C2), and\ IC(lso\ (C1,\ C2))$. The information content (IC) of concept is achieved by determining the probability of occurrence of a concept in a large text corpus and quantified as follows:

$$IC\left(C\right)=-\log P(C) \quad (2)$$

Where $P(C)$ is the probability of occurrence of concept '*C*' and is computed as follows:

$$P\left(C\right)=\frac{Freq\left(C\right)}{N} \quad (3)$$

Where *Freq*(*C*) is the number of occurrences of concept '*C*' in the taxonomy and *N* is the maximum number of nouns.

Given two sentences $S_i\ and\ S_j$, the semantic similarity between their corresponding predicate argument structures p_i and p_j is represented by $sim_{sem}(p_i, p_j)$ and is determined using (4), where $sim_{verb}(p_i, p_j)$ is the similarity between predicates (verbs), $sim_{arg}(p_i, p_j)$ refers to the sum of similarities between the corresponding arguments of the predicates. We use Jiang's semantic similarity measure for computing similarity between noun terms in the semantic arguments of the predicate argument structures and the verbs of predicate argument structures respectively. $sim_{tmp}(p_i, p_j)$ denotes the similarity between corresponding temporal arguments, and similarity between corresponding location arguments is represented by $sim_{loc}(p_i, p_j)$. Since Jiang's measure is based on WordNet, the temporal and location arguments may not be found in the WordNet, therefore we use edit distance algorithm for computing possible match/similarity between temporal and location arguments of the predicates. The similarity between the two predicate argument structures is computed is as follows:

$$sim_{sem}\left(p_i,p_j\right)=sim_{verb}\left(p_i,p_j\right)+[sim_{\arg}\left(p_i,p_j\right)+sim_{tmp}\left(p_i,p_j\right) \quad (4)$$
$$+sim_{loc}\left(p_i,p_j\right)$$

The example illustrating the Jiang semantic similarity can be found in our previous research study [1]. Once the semantic similarity score for each pair of predicate argument structure is obtained, then semantic similarity matrix is built from the similarity scores of predicate argument structures.

C. Semantic Graph

The goal of this phase is to build semantic graph from the semantic similarity matrix constructed in previous phase. The undirected weighted semantic graph is constructed from similarity matrix (representing similarity scores of predicate argument structures (PASs)) in such a way if the similarity weight $sim(p_i, p_j)$ between two predicate argument structures PASs p_i and p_j ($i \neq j$) is greater than 0 then a link is established between them, otherwise no link is established. We avoid self transition by letting $sim(p_i, p_i) = 0$. In this study, we are interested in only significant semantic similarities and define a similarity threshold that is empirically set to 0.5 [9]. So, a link is added between predicate argument structures (vertices), whose semantic similarity lies in the range of $0 < \beta \leq 0.5$; otherwise no link is established.

Formally, given a document set *D*, let *G* = (*Vs*, *Es*) is an undirected weighted graph that reveals the semantic relationship between predicate argument structures in the document set. Let *Vs* represents the set of vertices and each vertex v_i in *Vs* is the predicate argument structure in the document set. Let *Es* represents the set of edges and each edge e_{ij} in *Es* is labeled with the semantic similarity weight between predicate argument structures v_i and v_j (i≠j). The similarity weight between two predicate argument structures v_i and v_j is computed using (4) and it is written formally as follows:

$$f\left(v_i, v_j\right) = sim_{sem}\left(v_{i,}v_j\right) \qquad (5)$$

D. Augment Semantic Graph with PAS-to-document set Relationship

In order to reflect the impact of document set on predicate argument structures (PASs), this phase additionally augments the edge of semantic graph (representing semantic similarity weight between PASs), with PAS-to-document set relationship. We assume that the PASs which appear early in the document and have close distance to the centroid of the document set will be considered as salient and will get more chances to be selected for summary. Thus, we express the correlation/relationship of PAS to document set by four features discussed in our previous study [1] and are given as follows: PAS to PAS semantic similarity(refers to the average similarity of PAS with other PASs in the document set), position, TF-IDF and frequent semantic term. Since text features are sensitive the quality of summary i.e. not all features have same relevance with respect to summary. Therefore, this study exploits genetic learning algorithm (GA) to obtain optimal feature weights. GA is chosen since it is a robust and adaptive optimization technique [33]. The training and testing of GA is performed on 59 multi-documents (obtained from DUC 2002) using 10-fold cross validation. The initial population chosen for GA contains 50 chromosomes, which is initialized with the real values between 0 and 1 in a random manner. The fitness function is the average recall obtained with each chromosome when summarization process is applied to the training corpus. The fitness function determines the fitness value of each chromosome. Based on fitness value, the best chromosomes from the current population are selected as parents for the next generation. Selected chromosomes are then reproduced using cross over and mutation operations in each generation. In order to let the fitness value to converge, we run 100 maximum generations before terminating the GA process. The individual chromosome that achieves the highest value of fitness is chosen as the optimal feature weights. The scores of these features adjusted by optimized weights are computed, and combined to give the strength of correlation of PAS to document set. The strength of relationship/correlation between PAS v_i and its document set $D_{set}(v_i)$ is defined as follows:

$$w\left(v_i, D_{set}\left(v_i\right)\right) = \sum_{k=1}^{4} v_i\_f_k \qquad (6)$$

We adjust features by their corresponding optimal weights obtained using GA and therefore (6) can be rewritten as follows:

$$w\left(v_i, D_{set}\left(v_i\right)\right) = \sum_{k=1}^{4} w_k \times v_i\_f_k \qquad (7)$$

Where $w(v_i, D_{set}(v_i))$ represents the strength of relationship/correlation between PAS v_i and its document set $D_{set}(v_i)$, $v_i\_f_k$ is score of feature k for PAS v_i and w_k is the weight of feature k.

Once strength of relationship/correlation between PAS v_i and its document set $D_{set}(v_i)$ is defined, so, the new bounded/conditional similarity weight for the edge (PAS-to-PAS similarity) in the semantic graph is represented by $f\left(v_i, v_j \mid D_{set}\left(v_i\right), D_{set}\left(v_j\right)\right)$. It is calculated by linearly merging the similarity weight constrained on the document set containing PAS v_i i.e. $f\left(v_i, v_j \mid D_{set}\left(v_i\right)\right)$ and the similarity weight constrained on the same document set containing PAS v_j i.e. $f\left(v_i, v_j \mid D_{set}\left(v_j\right)\right)$. The conditional similarity weight is formally computed as follows:

$$f\left(v_i, v_j \mid D_{set}\left(v_i\right), D_{set}\left(v_j\right)\right)$$

$$= \mu . f\left(v_i, v_j \mid D_{set}\left(v_i\right)\right) + \left(1-\mu\right) . f\left(v_i, v_j \mid D_{set}\left(v_j\right)\right)$$

$$= \mu . f\left(v_i, v_j\right) . w\left(v_i, D_{set}\left(v_i\right)\right) + (1-\mu) . f\left(v_i, v_j\right) . w\left(v_j, D_{set}\left(v_j\right)\right)$$

$$= f\left(v_i, v_j\right) . [\mu . w\left(v_i, D_{set}\left(v_i\right)\right) + \left(1-\mu\right) . w . \left(v_j, D_{set}\left(v_j\right)\right)]$$

$$= sim_{sem}\left(v_{i,} v_j\right) . [\mu . w\left(v_i, D_{set}\left(v_i\right)\right) + \left(1-\mu\right) w\left(v_j, D_{set}\left(v_j\right)\right)] \quad (8)$$

Where $\mu \in [0,1]$ is a combination weight controlling the contribution from the document set containing v_i, and the same document set containing v_j. In this study, μ is set to 0.5 (optimal value), based on experimental observation.

E. *Modified Graph based Ranking Algorithm*

Conventionally, Google's PageRank [34] and HITS algorithm [35] are graph based ranking algorithms that have been effectively employed in Web-link analysis and social networks. PageRank algorithm applied on undirected graph achieved the best performance in DUC 2002 multi-document extractive summarization task. The PageRank is a graph-based ranking algorithm/model that provides the means for determining the significance of a vertex within a graph by considering global information from the whole graph.

Previous graph based methods exploit relationships/associations between sentences based on content similarity and did not consider semantic relationships between sentences. These methods apply similar procedure like PageRank to choose sentences based on number of "votes", received from their neighbouring sentences. To our knowledge, graph based ranking algorithm has not been considered for multi-document abstractive summarization. This study employs a modified weighted graph based ranking algorithm (MWGRA), which will take into account the edge weights in the vertices (PASs) ranking process (or importance analysis). The edge weight corresponds to PAS-to-PAS semantic similarity, and **PAS-to-document set** relationship. We let denote the salience or importance score of predicate argument structure v_i by $MWGRA(v_i)$. The importance score of predicate argument structure can be inferred from all those predicate argument structures that are connected to it; and we formulate it in a recursive manner as follows:

$$MWGRA\left(v_i\right) = (1-d_p) + d_p . \sum_{v_j \in \, In(v_i)} \frac{MWGRA(v_j) . w_{ji}}{\sum\limits_{v_z \in \, Out(v_j)} w_{zw}} \quad (9)$$

Where d_p denotes the damping factor in the ranking algorithm, and usually assigned a value of 0.85 [34]. $In\left(v_i\right)$ are the number of vertices that are pointing to given vertex v_i, $Out\left(v_j\right)$ are the number of outgoing links from the *vertex* v_j, w_{ji} represents the weight associated with the edge between vertices (PASs) v_i and v_j. w_{zw} represents the weights associated with outgoing links from vertex v_j.

From implementation perspective, the initial scores of all vertices (PASs) are set to 1. The modified weighted graph based ranking algorithm (MWGRA) employs (9), is run on undirected weighted graph to calculate the new salience/ranking scores of the vertices (PASs). The ranking/iteration algorithm keeps on computing the salience scores of the vertices until convergence is achieved. The converge is achieved by the ranking/iteration algorithm, when the difference between the ranking scores determined for any vertices (PASs) at two successive iterations falls below a given threshold (0.0001 in this work) [9]. After the convergence is achieved, the ranking scores obtained for vertices (PASs) of the graph are sorted in reverse order. Fig. 2 depicts the undirected graph for document set d061 in DUC 2002, and the ranking scores of the graph vertices obtained with the graph based ranking algorithm. The ranking scores are enclosed in square brackets, and appear next to each vertex of the graph.

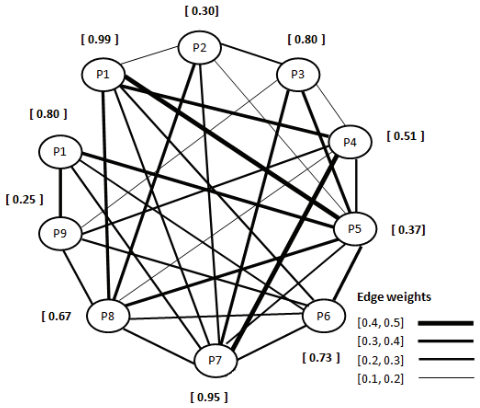

Fig. 2. PageRank score computed for different vertices of the graph, shown in square brackets next to the vertices.

F. *MMR to Reduce Redundancy*

The ranking algorithm may assign same ranking score to the PASs representing the same concept, and therefore the final summary may contain redundant information. Furthermore, the other concepts of the documents which represents group of least similar PASs may not be included in the final summary; and thus significant information may lost. In this study, we employ a modified version of MMR [36] to reduce redundancy by re-ranking the PASs for inclusion in summary. A predicate argument structure is included in the summary generation list if it is not too similar to any existing PAS in the summary generation list. At first, the PAS with the highest salience score is selected and appended into the summary generation list. Then, the subsequent PAS having the higher salience score according to (10) is selected, and added into summary generation list. This process chooses PASs by taking into account both importance and redundancy and keeps on repeating until the compression rate of summary is met.

$$MMR = argmax_{p_i \in R \setminus P}\left[\alpha . RS\left(p_i\right)\right] - \left(1-\alpha\right) . \max_{p_j \in P.} sim\left(p_i, p_j\right) (10)$$

In (10), R is the set of all predicate argument structures to be summarized, P is the set of PASs already selected for inclusion in summary generation, $R\setminus P$ is the set of as yet unselected PASs in R, $RS(p_i)$ is the ranking (salience) score for PAS determined previously, $sim(p_i, p_j)$ refers to the semantic similarity [37] between PASs and α is a tuning parameter between PAS's importance and its relevance to previously chosen predicate argument structures. We set value of α = 0.6 [23] for the optimal performance.

G. Abstractive Summary Generation

This phase takes the top scored predicate argument structures (PASs) from previous phase, employs SimpleNLG [38] and a simple heuristic rule implemented in it to generate summary sentences from PASs. SimpleNLG is an English realisation engine, which provides simple interfaces to produce syntactical structures and transform them into sentences using simple grammar rules. The simple heuristic rule states that if the subjects in the predicate argument structures (PASs) refer to the same entity, then merge the predicate argument structures by removing the subject in all PASs except the first one, separating them by a comma (if there exist more than two PASs) and then combine them using connective such as "and".

For instance, the following source input sentences :

S$_1$: Hurricane Gilbert claimed to be the most intense storm on record in terms of barometric pressure.

S$_2$: Hurricane Gilbert slammed into Kingston on Monday with torrential rains and 115 mph winds.

S$_3$: Hurricane Gilbert ripped roofs off homes and buildings.

After applying SENNA SRL, the corresponding three predicate argument structures *P*$_1$, *P*$_2$ and *P*$_3$ are obtained as follows:

P$_1$: [*A0*: Hurricane Gilbert] [*V*: claimed] [*A1*: to be the most intense storm on record]

P$_2$: [*A0*: Hurricane Gilbert] [*V*: slammed] [*A1*: into Kingston] [*AM-TMP*: on Monday]

P$_3$: [*A0*: Hurricane Gilbert] [*V*: ripped] [*A1*:roofs off homes and buildings]

We assume that *P*$_1$, *P*$_2$ and *P*$_3$ are the top scored predicate argument structures selected from previous step. According to the rule stated above, the subject *A0* is identified as repeated in the above example and is eliminated from all predicate argument structures except the first one. The SimpleNLG applies the heuristic rule on the above three predicate argument structures and form the summary sentence that is compression version of the original source sentences.

Summary Sentence: Hurricane Gilbert claimed to be the most intense storm on record, slammed into Kingston on Monday with torrential rains and ripped roofs off homes and buildings.

IV. EVALUATION RESULTS

The proposed semantic graph based approach for multi-document summarization is evaluated using Document Understanding Conference (DUC) 2002 document sets (DUC, 2002). DUC 2002 is a standard corpus used in text summarization research, which contains documents along with their human model summaries (both extractive and abstractive summaries). The data set chosen for our work refers to task2 (multi-document extractive summarization) and task3 (multi-document abstractive summarization) in DUC 2002. There are also other editions of DUC data sets, however they lack human produced abstracts..

This study employs two standard evaluation metrics: ROUGE [39] and Pyramid [40], for the evaluation of our proposed approach. The Pyramid metric measures the quality of system generated summary by comparing it with human model summaries. Pyramid score (Mean Coverage Score or Recall) [40] for peer summary or candidate summary is computed as follows:

$$Mean\ Coverge\ Score = \frac{Total\ Peer\ SCUs\ Weight}{Average\ SCU\ in\ the\ Model\ Summary} \quad (11)$$

Where SCUs refers to the summary content units and their weights correspond to number of model (human) summaries they appeared in. The precision for peer summary [40] or candidate summary is computed as follows.

$$Precision = \frac{Number\ of\ Model\ SCUs\ expressed\ in\ Peer\ Summary}{Average\ SCU\ in\ the\ Peer\ Summary} \quad (12)$$

The F-measure for peer summary can be computed from (11) and (12) as follows:

$$F - Measure = \frac{2 \times Mean\ Coverage\ Score \times Precision}{Mean\ Coverage\ Score + Precision} \quad (13)$$

There are many variants of ROUGE evaluation measures and we found based on literature that ROUGE-1, ROUGE-2 are effectively employed for multi-document extractive summarization [39]. ROUGE − N can be defined [39] as an n-gram recall between a system summary and a set of reference summaries. The results of optimal feature weighting obtained using genetic algorithm is depicted in Fig. 3.

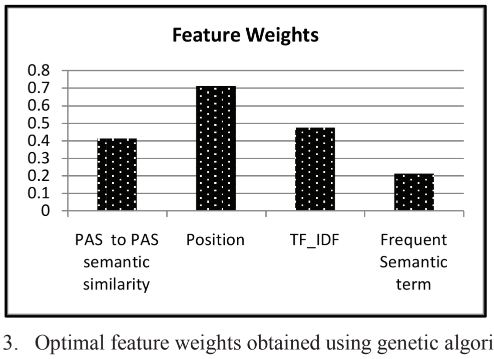

Fig. 3. Optimal feature weights obtained using genetic algorithm

The optimal feature weights obtained are 0.41313752, 0.7118985, 0.475472, 0.21308642 correspond to the weights for PAS to PAS semantic similarity, position, TF_IDF and frequent semantic term respectively.

The proposed approach is evaluated in the context of multi-document abstractive and extractive summarization shared tasks, using 59 news articles/data sets provided by the Document Understanding Evaluations 2002 [41]. For each data set, our proposed approach generates a 100 words summary, the tasks undertaken by other systems participating in multi-document abstractive and extractive summarization tasks. To compare the performance of our proposed approach (Sem-Graph) in the context of DUC 2002 multi-document abstractive summarization shared task, we setup three comparison models (Best, Avg, AS) besides the average of human model summaries (Humans). For comparative evaluation, Table I shows comparison of abstractive summarization results for different systems over the mean coverage score (average recall), average precision and average F-measure obtained on DUC 2002 dataset.

TABLE I. Comparison of multi-document abstractive summarization results in DUC 2002 based on mean coverage score, average precision and average F-Measure

System	Mean Coverage Score	AVG-Precision	AVG-F-Measure
DUC-2002 Humans	0.6910	0.8528	0.7634
Sem-Graph	**0.5247**	**0.7267**	**0.6094**
AS [14]	0.4378	0.643	0.5209
DUC 2002 Best (System 19)	0.2783	0.7452	0.4053
DUC 2002 Avg	0.1775	0.6700	0.2806

On other hand, Table II shows comparative evaluations of the proposed approach with other extractive summarization models based on recall obtained with ROUGE-1 and ROUGE-2 measures, achieved on DUC 2002 data set.

TABLE II. Comparison of the proposed approach (Sem-Graph) with multi-document extractive summarization systems based on recall obtained with ROUGE-1 and ROUGE-2.

System	ROUGE-1	ROUGE-2
DUC-2002 Best (System 21)	0.395	0.103
DUC-2002 Humans	0.418	0.102
Event graph [24]	0.415	**0.116**
Sem-Graph	**0.417**	0.101

V. Discussion

This section discusses the results presented in previous section. First we discuss the results given in Table I, we can observe that on mean converge score, average precision and average F-measure, the proposed approach (Sem-Graph) outperforms the best system and the recent multi-document abstractive summarization approach (AS) [14], and came second to the average of human model summaries (Humans). In order to validate the results of the proposed approach and other comparison models, we also carried out a statistical significance test (Paired-Samples T-test), and achieved low significance value of $p < 0.05$. These results suggest that the summary produced by our approach (Sem-Graph) is more closer to the way humans produce summary as compared to other comparison models. Moreover, refer to the results given in Table II, the proposed summarization approach (Sem-Graph) performs better than the best system and recent graph based multi-document extractive summarization approach (Event graph) based on ROUGE-1 measure. However, based on ROUGE-2 measure, the performance of the proposed approach slightly degrades as compared to the recent graph based extractive summarization approach (Event graph). This might be due to the fact that our proposed abstractive summarization approach generates summary that contains compressed version of original source sentences; while on other hand, extractive summarization systems generate summary that contains original source sentences. ROUGE-1 and ROUGE-2 measures look for exact matches of text snippets while comparing system summary against human produced summary(extracts). Thus, the abstractive summary produced by our approach will contain less matching text snippets with the human produced summary as compared to event graph based extractive summarization system. Paired-Samples T-test is also carried out to validate the results of the proposed approach and other extractive summarization models and obtained a significance value of $p < 0.05$. These results confirm that PAS-to-PAS semantic similarity and PAS-to-document set relationship employed in the graph based ranking algorithm helps to improve summarization results.

VI. Conclusion

Although fully abstractive summarization is a big challenge, our proposed semantic graph based approach shows the feasibility of this new direction for summarization research. Existing graph based approaches treat sentence as bag of words and cannot capture redundant sentences that are semantically equivalent as they mostly rely on content similarity measure. The proposed approach assumes semantic structure of sentence - predicate argument structure as graph node, and establish semantic relationships between PASs using Jiang semantic similarity measure. The semantic similarity measures assists in detecting redundancy by capturing semantically equivalent predicate argument structures. The proposed graph based approach incorporates PAS-to-PAS semantic similarity and PAS-to-document set relationship into the graph-based ranking algorithm, and experimental results demonstrate that modified ranking algorithm improves summarization results. The approach is promising enough to be applicable to any domain and does not require any intervention of human experts. In future, we will explore Cross-Document Structural Theory (CST) relations for multi-document abstractive summarization and examine their impact on summarization.

Acknowledgment

This work is supported by the Higher Education Commission (HEC), Islamia College Peshawar (Chartered University), Pakistan, Soft Computing Research Group (SCRG) of Universiti Teknologi Malaysia (UTM) and Ministry of Higher Education (MOHE). This work is also supported in part by grant from Fundamental Research Grant Scheme (Vot R.J130000.7828.4F373).

References

[1] A. Khan, N. Salim, and Y. J. Kumar, "A framework for multi-document abstractive summarization based on semantic role Labelling," *Applied Soft Computing,* 2015.

[2] Y. J. Kumar, N. Salim, A. Abuobieda, and A. Tawfik, "Multi document summarization based on cross-document relation using voting technique," in *Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference on*, 2013, pp. 609-614.

[3] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, 1995, pp. 68-73.

[4] K. Knight and D. Marcu, "Statistics-based summarization-step one: Sentence compression," in *Proceedings of the National Conference on Artificial Intelligence*, 2000, pp. 703-710.

[5] B. Larsen, "A trainable summarizer with knowledge acquired from robust NLP techniques," *Advances in Automatic Text Summarization,* p. 71, 1999.

[6] M. A. Fattah, "A hybrid machine learning model for multi-document summarization," *Applied Intelligence,* vol. 40, pp. 592-600, 2014.

[7] G. Erkan and D. R. Radev, "LexPageRank: Prestige in Multi-Document Text Summarization," in *EMNLP*, 2004, pp. 365-371.

[8] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Intell. Res.(JAIR),* vol. 22, pp. 457-479, 2004.

[9] R. Mihalcea and P. Tarau, "A language independent algorithm for single and multiple document summarization," 2005.

[10] X. Wan and J. Yang, "Improved affinity graph based multi-document summarization," in *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume: Short Papers*, 2006, pp. 181-184.

[11] R. Barzilay and K. R. McKeown, "Sentence fusion for multidocument news summarization," *Computational Linguistics,* vol. 31, pp. 297-328, 2005.

[12] R. Barzilay, K. R. McKeown, and M. Elhadad, "Information fusion in the context of multi-document summarization," in *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, 1999, pp. 550-557.

[13] H. Tanaka, A. Kinoshita, T. Kobayakawa, T. Kumano, and N. Kato, "Syntax-driven sentence revision for broadcast news summarization," in *Proceedings of the 2009 Workshop on Language Generation and Summarisation*, 2009, pp. 39-47.

[14] P.-E. Genest and G. Lapalme, "Framework for abstractive summarization using text-to-text generation," in *Proceedings of the Workshop on Monolingual Text-To-Text Generation*, 2011, pp. 64-73.

[15] S. M. Harabagiu and F. Lacatusu, "Generating single and multi-document summaries with gistexter," in *Document Understanding Conferences*, 2002.

[16] P.-E. Genest and G. Lapalme, "Fully abstractive approach to guided summarization," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Short Papers-Volume 2*, 2012, pp. 354-358.

[17] C.-S. Lee, Z.-W. Jian, and L.-K. Huang, "A fuzzy ontology and its application to news summarization," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 35, pp. 859-880, 2005.

[18] C. F. Greenbacker, "Towards a framework for abstractive summarization of multimodal documents," *ACL HLT 2011,* p. 75, 2011.

[19] I. F. Moawad and M. Aref, "Semantic graph reduction approach for abstractive Text Summarization," in *Computer Engineering & Systems (ICCES), 2012 Seventh International Conference on*, 2012, pp. 132-138.

[20] J. C. K. Cheung and G. Penn, "Towards Robust Abstractive Multi-Document Summarization: A Caseframe Analysis of Centrality and Domain," in *ACL (1)*, 2013, pp. 1233-1242.

[21] F. Wei, W. Li, Q. Lu, and Y. He, "A document-sensitive graph model for multi-document summarization," *Knowledge and information systems,* vol. 22, pp. 245-259, 2010.

[22] S. S. Ge, Z. Zhang, and H. He, "Weighted graph model based sentence clustering and ranking for document summarization," in *Interaction Sciences (ICIS), 2011 4th International Conference on*, 2011, pp. 90-95.

[23] T.-A. Nguyen-Hoang, K. Nguyen, and Q.-V. Tran, "TSGVi: a graph-based summarization system for Vietnamese documents," *Journal of Ambient Intelligence and Humanized Computing,* vol. 3, pp. 305-313, 2012.

[24] G. Glavaš and J. Šnajder, "Event graphs for information retrieval and multi-document

summarization," *Expert Systems with Applications,* vol. 41, pp. 6904-6916, 2014.

[25] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web," 1999.

[26] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research,* vol. 12, pp. 2493-2537, 2011.

[27] M. F. Porter, "Snowball: A language for stemming algorithms," ed, 2001.

[28] M. Yavaş and G. Yücel, "Impact of homophily on diffusion dynamics over social networks," *Social Science Computer Review,* p. 0894439313512464, 2014.

[29] S. Yi-Lun, "Multi-type directed scale-free percolation," *Communications in Theoretical Physics,* vol. 57, p. 701, 2012.

[30] Y. Li, Z. A. Bandar, and D. McLean, "An approach for measuring semantic similarity between words using multiple information sources," *Knowledge and Data Engineering, IEEE Transactions on,* vol. 15, pp. 871-882, 2003.

[31] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," *arXiv preprint cmp-lg/9709008,* 1997.

[32] G. A. Miller, "WordNet: a lexical database for English," *Communications of the ACM,* vol. 38, pp. 39-41, 1995.

[33] M. Srinivas and L. M. Patnaik, "Genetic algorithms: A survey," *Computer,* vol. 27, pp. 17-26, 1994.

[34] S. Brin and L. Page, "The anatomy of a large-scale hypertextual Web search engine," *Computer networks and ISDN systems,* vol. 30, pp. 107-117, 1998.

[35] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," *Journal of the ACM (JACM),* vol. 46, pp. 604-632, 1999.

[36] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents and producing summaries," in *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 1998, pp. 335-336.

[37] J. J. Jiang and D. W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," *Proc. of the Int'l. Conf. on Research in Computational Linguistics* pp. 19-33, 1997.

[38] A. Gatt and E. Reiter, "SimpleNLG: A realisation engine for practical applications," in *Proceedings of the 12th European Workshop on Natural Language Generation*, 2009, pp. 90-93.

[39] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 2004, pp. 74-81.

[40] A. Nenkova and R. Passonneau, "Evaluating content selection in summarization: The pyramid method," 2004.

[41] DUC, "Document understanding conference 2002," 2002.