

# Pazesh: A Graph-Based Approach to Increase Readability of Automatic Text Summaries

Nasrin Mostafazadeh<sup>1</sup>, Seyed Abolghassem Mirroshandel<sup>1</sup>,  
Gholamreza Ghassem-Sani<sup>1</sup>, and Omid Bakhshandeh Babarsad<sup>2</sup>

<sup>1</sup> Computer Engineering Department,

Sharif University Of Technology, Tehran, Iran

<sup>2</sup> Mechatronics Research Laboratory, Computer Engineering Department,

Qazvin Azad University, Qazvin, Iran

{mostafazadeh,mirroshandel}@ce.sharif.edu

sani@sharif.edu, o.bakhshandeh@qiau.ac.ir

**Abstract.** Today, research on automatic text summarization challenges on readability factor as one of the most important aspects of summarizers' performance. In this paper, we present Pazesh: a language-independent graph-based approach for increasing the readability of summaries while preserving the most important content. Pazesh accomplishes this task by constructing a special path of salient sentences which passes through topic centroid sentences. The results show that Pazesh compares approvingly with previously published results on benchmark datasets.

## 1 Introduction

Research in automatic text summarization (ATS) area dates back to late 60's [1], though solving the problem in a substantial manner seems to yet require a long trail to work on. Among a variety of different approaches to address this problem, **Graph-Based methods** have noticeably attracted attention. For the first time in ATS history, Salton [2] proposes a graph as a model for input text. Recent graph-based approaches compute sentence importance based on the eigenvector centrality concept and applying ranking algorithms (e.g., TextRank [3], LexRank [4]). Today, most summarization researches are focused on *extractive* genre which tends to select a number of sentences out of the initial text. Normally there are many topic shifts in a text and highly scored sentences can come from diverse important topics which require careful output sentences selection. Some methods have already been devised to optimize the search problem of finding the best scoring summary [5,6] and ordering text entities based on chronological order of events [7]. Such methods might construct a sentence-to-sentence coherent body of information but they *neglect preserving the most important content*.

In this paper, we introduce Pazesh: a new extractive, graph-based, and language-independent approach to address both readability and informativeness criteria of single-document summaries. At first, Pazesh segments the text

in order to find topic centroid sentences. Then it ranks text entities by its specially constructed graphs and at last it finds the most precious path passing through centroid sentences. The obtained evaluation results show that our algorithm performs well on both readability and informativeness aspects. The rest of this paper is organized as follows: Section 2 introduces centroid finding and graph construction phases. In Section 3, the main phase of algorithm, “addressing Readability”, is revealed. Section 4 evaluates Pazesh, and finally Section 5 concludes the paper’s overall idea and focus.

## 2 Finding Centroid Sentences

Pazesh follows three steps for finding topic centroid sentences: segmenting the text into coherent partitions, scoring sentences of each segment and scoring segments individually. A *topic* is what a discourse segment or a sentence is about. A text can be segmented by its different topics denoted by sentences. In Pazesh, we utilize a segmentation algorithm to find the topic sentences to be used as landmarks of the final coherent path. Here we have used a simple partitioning approach: *TextTiling* [8] algorithm which is a method for partitioning a text into a number of coherent multi-paragraph partitions representing the text’s subtopics. This algorithm doesn’t rely on semantic relationships between concepts for discovering subtopic structures so that it is **language-independent** [8] and is highly suitable for Pazesh.

After segmentation phase, we construct a *weighed undirected graph* for each segment. The sentences within a segment would be the graph’s nodes and similarity between sentences would be graph’s weighed edges. We weight word  $w$  of segment  $S$  named  $W(w,S)$  by  $freq(w,S)*IDF(w)$  where  $freq(w,S)$  is the number of sentences in segment  $S$  containing  $w$ , and  $IDF$  as the inverse document frequency of word  $w$  equals  $log(\frac{\# \text{ all sentences}}{\# \text{ sentences containing } w})$ . Then the similarity between sentences of each segment can be measured using cosine similarity formula as follows:

$$edge(S_1, S_2) = \frac{\sum_{w:w \in S_1 \text{ and } S_2} W(w, S)^2}{\sqrt{\sum_{w:w \in S_1} W(w, S)^2} * \sqrt{\sum_{w:w \in S_2} W(w, S)^2}} \quad (1)$$

where  $S$  is the segment containing  $S_1$  and  $S_2$ . Now that we have the segments’ graph, we use PageRank[9] scoring function for scoring nodes of these graphs. The use of ranking algorithms in ATS has been first introduced in TextRank. PageRank does not require a deep linguistic knowledge and is highly suitable for Pazesh. Suggested  $PR(V)$  score of vertex  $V$  is computed as follows:

$$PR(V_i) = (0.25) + 0.25 * \sum_{V_j \in Ln(V_i)} \frac{w_{ij}}{\sum_{V_k \in Ln(V_j)} w_{jk}} * PR(V_j) \quad (2)$$

Where  $Ln(V)$  is a link of the node. We call above score the **segment-global score** of each sentence. We define *centroid sentence* of each segment as its representative and the most salient entity having highest segment-global score.

In Pazesh we achieve both non-redundancy and noise-removing by constructing and ranking another graph: ‘Segment graph’ which has segments as its nodes and segment similarities as the edges.

The final graph of the whole text would be a ‘directed weighed’ graph with each sentence as its nodes and sentence similarities as its edges. This graph is constructed to compute the document global importance of each sentence. The same cosine similarity function here is calculated for all sentences of the text. We apply the PageRank algorithm to this final graph and generate each sentence’s **document-global score**. For tuning this final graph for Path-Finding phase we make it a *directed acyclic graph* (DAG). The direction of each edge would be from a predecessor sentence in initial text to any successor sentence. Also, we tag all edges weighed below a certain threshold  $\gamma$ , shallow edges and the rest edges are tagged deep edges. The final graph looks like Fig. 1.

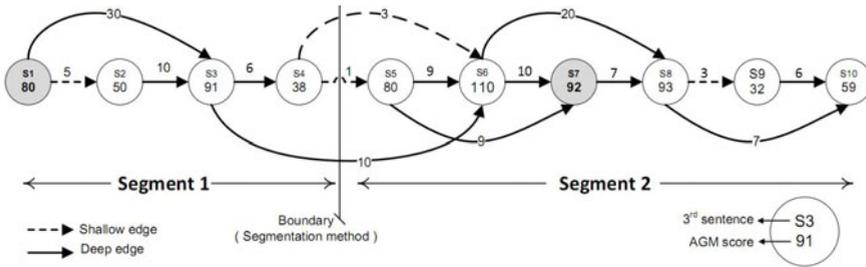
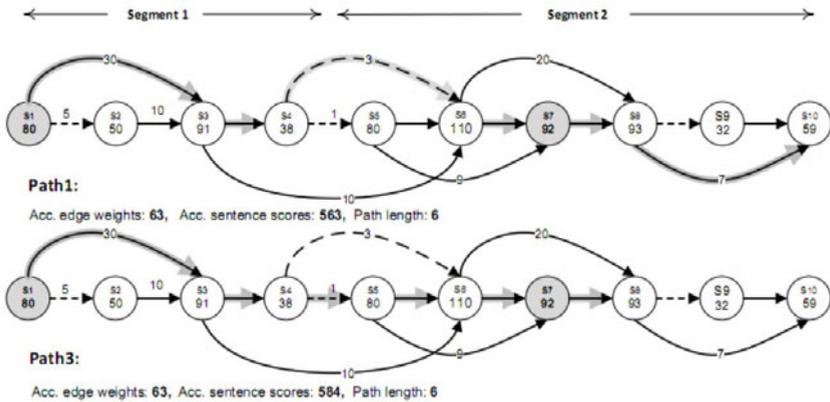


Fig. 1. A sample final graph. Nodes in gray are centroids of the corresponding segment.

### 3 Addressing Readability

Text readability is a measure of how well and easily a text conveys its intended meaning. In Document Understanding Conference (DUC) the linguistic quality markers defined to evaluate the readability aspects of summaries are: **Grammaticality**, **Non-Redundancy**, **Referential Clarity**, **Focus**, and **Structure and Coherence**. In Pazesh, we address these readability criteria as follows: By being committed to chronological order of the initial document and its structure –as a grammatically accepted text–, we can avoid fundamental grammatical errors. Though being extractive undesirably lead the output to yet have some errors. We address non-redundancy and focus by “filtering centroid sentences” and also “retaining the informativeness of the output summary”. Though in Pazesh no alternations are made to input text’s sentences, so repeated use of nouns or noun-phrases are probable. Also, Implementing Pazesh’s idea for addressing referential clarity has been left as future work. Finally, **Cohesion** can be defined as the “links” that hold sentences of a text together and give the whole text a clear meaning. Here, we use this term on behalf of *lexical relationship* of sentences.

In the case of our final graph in order to have a cohesive output, each sentence should be followed by another sentence already being linked to that. Therefore, the final cohesive text can be assumed as a **path**. Pazesh’s path is built using deep edges (introduced earlier) to guarantee the readability of the path. Since topic shifts from one centroid to another is sensible, we accept shallow edges only as connectors between different segments. The path obtained from the final graph should “pass all centroid sentences” – **landmarks** – and have the “highest accumulative sentence score”. Passing through centroids has two outcomes: Firstly, centroids as the most salient sentences are guaranteed to be included in the summary. Secondly, the remainder sentences of the summary are connecting centroids; thus they come from prominent sub-topic segments and are important themselves. In Fig. 2 the path-finding method is applied to the graph in Fig. 1 and the output paths are depicted in gray.



**Fig. 2.** Path-Finding phase of Pazesh: Compression ratio= 0.4. Connecting two centroids is possible through 3 different Paths. Each path has a different accumulative sentences score/edge weight. Note that the path can go beyond the last centroid in order to meet the summary length constraint. In scenario Paz1, Path1 would be the output. In scenario Paz2, path1 and path3 have identical scores and one of them would be selected.

## 4 Evaluation

To evaluate the system, we used two distinct evaluation methods: an automatic evaluation by ROUGE toolkit<sup>1</sup> and a manual readability evaluation based on the DUC readability assessment guidelines. The informativeness of Pazesh was evaluated in the context of an extractive single-document summarization task, using 567 news articles of DUC2002. For each article, the evaluation task is to generate a 100-word summary. For evaluation, we used recall score of the three different

<sup>1</sup> ROUGE is available at <http://berouge.com/>

metrics of ROUGE: ROUGE-1 (unigram-based), ROUGE-2 (bigram-based), and ROUGE-W. Two of Pazesh’s evaluation scenarios are: Constructing final path based on **Paz1** (accumulative sentences score) vs. **Paz2**: (edge weights). These settings of Pazesh are compared with 4 baselines in 2 categories: 1) Outperforming informative systems (not being readable) including SentenceRank and another single-document extractive summarizer called method1 [10]. 2) A readable system, A\* searching [5]. Due to lack of a universal assessment methodology and a few systems using identical measurement, comparing readability factor of our system with previous works is not feasible in high quality and quantity. Table 1 depicts mentioned ROUGE scores.

**Table 1.** ROUGE scores of different systems. ‘-’: not reported.

System	ROUGE-1	ROUGE-2	ROUGE-W
Paz1	<b>0.38</b>	<b>0.24</b>	0.08
Paz2	0.23	0.19	0.05
SentenceRank	0.45	0.19	0.15
Method1	0.47	0.2	0.16
A*	0.37	0.08	-

Comparing Pazesh1 with Pazesh2 reveals that taking into account the sentences scores result in undisputedly better informativeness than considering edge weights. The ROUGE-2 score is competitive with informative systems since n-gram length greater than 1 in ROUGE estimates the fluency of summaries. However, the overall results are not outperforming systems focusing on only informativeness factor but are competitive with them to some degree. This is due to the fact that preserving both informativeness and readability -specially for the cases having summary length constraint-is a trade-off.

Since 2006, a separate set of quality questions have been introduced by DUC to evaluate readability aspects of summaries. However, there are still no automatic evaluations to assess such aspects. For manual evaluation, we used 20 documents of the same dataset and scenarios used in automatic evaluation. Evaluation was accomplished by ten human judges who had read the DUC guidelines for readability assessment earlier. For each scenario, three different variations of summary length constraint were applied. Table 2 show the results of our manual evaluation on five-point scale. The results show that Pazesh performs very well on criteria it intended to address: Coherency and Focus. The overall results on ratio 0.6 outperform ratio 0.3 and no-ratio (minimum possible length) setting. This was expected since the readability aspect of a text inherently can be meaningful in a long text rather than a short text. Comparing the results reveals that Paz2 outperforms Paz1 on coherency aspect which could be estimated. Though Paz1 performs stronger than Paz2 on focus aspect. Putting all together, obtained results show that Pazesh has accomplished its intended mission: meeting both readability and informativeness.

**Table 2.** Readability Assessment. Left to right: ratio=0.6, ratio=0.3, ratio=not given.

Criterion	Paz1	Paz2	Paz1	Paz2	Paz1	Paz2
Grammaticality	5	5	5	5	5	5
Non-redundancy	4	5	5	5	5	5
Referential clarity	2	3	2	2	1	2
Focus	5	4	3	3	4	3
Struct. & Coherence	5	5	3	4	3	4

## 5 Conclusion

In this paper, we introduced Pazesh: a graph-based approach to address both readability and informativeness of automatic text summaries. This is accomplished by constructing a path of highly ranked sentences, as a readable sequence, passing through centroid sentences. As it is shown in experimental results, Pazesh is a powerful and also simple summarizer system.

**Acknowledgments.** Special thanks go to Dr. Yahya Tabesh, Ali Moini for igniting the initial motivation on this research. This research is honoured to be supported by a grant from Iran’s National Foundation of Elites.

## References

1. Luhn, H.P.: The automatic creation of literature abstracts. *IBM Journal of Research Development* 2(2), 159–165 (1958)
2. Salton, G., Singhal, A., Mitra, M., Buckley, C.: Automatic Text Structuring and Summarization. *Information Processing and Management Journal* 33(2), 193–207 (1997)
3. Mihalcea, R.: Graph-based ranking algorithms for sentence extraction, applied to text summarization. In: *Proceedings of ACL, Spain* (2004)
4. Erkan, G., Radev, D.R.: LexRank: Graph-based Lexical Centrality as Saliency in Text Summarization. *JAIR* 22, 457–479 (2004)
5. Aker, A., Cohn, T., GaizauskasMulti, R.: Document summarization using A\* search and discriminative training. In: *Proceedings of EMNLP, USA*, pp. 482–491 (2010)
6. Riedhammer, K., Gillick, D., Favre, B., Hakkani-T, D.: Packing the meeting summarization knapsack. In: *Proceedings of the Interspeech Conference, Australia*, pp. 2434–2437 (2008)
7. Barzilay, R., Elhadad, N., McKeown, K.R.: Inferring strategies for sentence ordering in multidocument news summarization. *JAIR* 17, 35–55 (2002)
8. Hearst, M.A.: Texttiling: segmenting text into multi-paragraph subtopic passages. *CL* 23(1), 33–64 (1997)
9. Page, L., Brin, S., Motwani, R., Winograd, T.: The pagerank citation ranking: Bringing order to the web. Technical report, Stanford University, USA (1998)
10. Xiaojun, W., Jianwu, Y., Jianguo, X.: Towards a reinforcement approach for simultaneous document summarization and keyword extraction. In: *Proceedings of the 45th ACL, Czech Republic*, pp. 552–559 (2007)