International Journal of Information Retrieval Research
Volume 10 • Issue 1 • January-March 2020

Introducing Word's Importance Level-Based Text Summarization Using Tree Structure

Nitesh Kumar Jha, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India
https://orcid.org/0000-0002-5209-1683

Arnab Mitra, Siksha 'O' Anusandhan (Deemed to be University), Bhubaneswar, India
https://orcid.org/0000-0002-7382-089X

ABSTRACT

Text-summarization plays a significant role towards quick knowledge acquisition from any text-based knowledge resource. To enhance the text-summarization process, a new approach towards automatic text-summarization is presented in this article that facilitates level (word importance factor)-based automated text-summarization. An equivalent tree is produced from the directed-graph during the input text processing with WordNet. Detailed investigations further ensure that the execution time for proposed automatic text-summarization, is strictly following a linear relationship with reference to the varying volume of inputs. Further investigation towards the performance of proposed automatic text-summarization approach ensures its superiority over several other existing text-summarization approaches.

KEYWORDS

Automatic Text Summarization, Directed Graph, Extractive Summary, Importance Level of Keywords, Tree

1. INTRODUCTION

Our present modern lives are blessed with the Internet. Among several others, most importantly, Internet helps us to acquire / enhance our knowledge. Due to ever increasing volume of text documents, it is very hard to read every single line of every single document. For this reason, text summarization plays a crucial role towards knowledge acquisition from available text documents (Pokojski et al., 2018).

Text summarization incorporates the uses of keywords. Keywords provide a compact representation about contents of a document. Keyword extraction is considered as primary task towards the automatic summarization of documents. Several text mining applications e.g., 'just-in-time (JIT)' based information retrieval, automatic classification, summarization, and filtering etc. were presented which uses keywords (Zhang, 2008; and Reddivari et al., 2018). Manual keyword extraction from any text document is time consuming, costly and tedious task. Furthermore, the ever-increasing number of the online documents makes the situation more critical towards manual

DOI: 10.4018/IJIRR.2020010102

processing. For this reason, automated text summarization and keyword extraction have attracted the attention of investigators over the past years (Beliga et al., 2015).

Automatic text summarization is a text mining mission that facilitates quick grasp of the overall perception for a text document (Thakkar et al., 2010; and Bharti et al., 2017). Text summarization may be achieved in the form of an abstractive summary or, as an extractive summary. Abstractive Summaries are often achieved after learning the internal representation of the article and the quality of summary is similar to the quality as produced by human being (https://rare-technologies.com/text-summarization-in-python-extractive-vs-abstractive-techniques-revisited/). Google's "Textsum" (https://rare-technologies.com/text-summarization-in-python-extractive-vs-abstractive-techniques-revisited/) is a state of art open-source abstractive summarization available at present days. It is capable for the creation of headlines from news-based articles using neural network (https://rare-technologies.com/text-summarization-in-python-extractive-vs-abstractive-techniques-revisited/). On the other hand, extractive summary extracts detail from the input article and presents the result to the user (Bharti et al., 2017). In our studies, we found that extractive summarization (based on keywords extraction) is mostly popular. For this reason, in our present research, we have focused towards the keywords-based extractive summarization. In our study, we find that graph-based approach is popular towards text summarization (Thakkar et al., 2010). Hence, a brief description on graph as presented in (Ruohonen, 2013), is presented next.

Graph G is a pair, containing set of vertices $V\left(G\right)$, set of edges $E\left(G\right)$, and a relation associated with each edge. Mathematically graph (*G*) as found in (Ruohonen, 2013) is presented once again in following Equation 1 (Ruohonen, 2013):

$$G = \left(V, E\right) \tag{1}$$

where, G denotes graph, V denotes set of vertices and E denotes edges formed by pair of vertices i.e, an arc or, edge between vertex u and vertex v is described as $\left(u, v\right)$.

Digraph (a directed graph) is designed with vertices (nodes) those are linked using directed edges (arrows). In digraph, elements of edges $\left(E\right)$ are strictly ordered pairs i.e, the arc from vertex u to vertex v is presented as $\left(u, v\right)$ (Ruohonen, 2013). On the other hand, undirected graphs do not have any directed edges (Ruohonen, 2013).

We found, past researches on text-summarization were based on undirected graph (Thakkar et al., 2010; and Bharti et al., 2017). In our study, we realized that level-based keyword's importance factor might be helpful towards automatic text-summarization. We found that the consideration of level-based keyword's importance factor was not possible in undirected graphs. A directed graph-based text summarization approach might be advantageous towards the automated text-summarization, as it facilitates the leveling of keywords' importance factor. Hence, we have presented a directed graph-based text-summarization by constructing an equivalent tree from the directed graph.

Major contributions as presented in this article are as followed:

1. A directed graph-based text summarization is presented;
2. Construction of an equivalent tree from the directed graph is shown;
3. Keyword's importance factor is introduced in equivalent tree generation process;
4. A new algorithm along with set of conditions are presented towards equivalent tree generation from the directed graph;
5. Breadth first search (BFS) based tree traversal is presented targeting efficient text-summarization.