Utilizing Association Rules in Knowledge Graphs for Enhanced News Summarization

MVPT Lakshika[*], HA Caldera
University of Colombo School of Computing, Colombo, Sri Lanka.

***Abstract*—The rapid progress in web news articles has led to an abundance of text content, often than needed, and consequently, misleading readers. Recent Knowledge Graph (KG) based approaches have proven successful in abstract summary generation due to their ability to represent structured and interconnected knowledge with semantic context. The KG ranking algorithm responsible for selecting graph data for inclusion in the abstract still relies on traditional ranking algorithms which lack the consideration for semantic relationships between graph nodes, and are associated with high memory consumption, processing times, and increased complexity. Knowledge discovery plays a crucial role in improving the quality of summarization by uncovering hidden patterns and enhancing contextual understanding. Therefore, our study centers on introducing a novel KG ranking algorithm, aimed at a statistically significant enhancement in abstract generation by integrating knowledge discovery techniques. The suggested ranking algorithm considers the semantic and topological graph properties and interesting relationships, patterns, and features in text data using Association Rule Mining techniques to identify the most significant graph information for generating abstracts. The experiments conducted using the DUC-2002 dataset indicate that the suggested KG ranking algorithm is effective in producing detailed and accurate abstracts for a collection of web news articles.**

***Index Terms*—Knowledge graph, association rules, abstract generation, ranking algorithm**

I. Introduction

Online news portals are a comprehensive source of authenticated information that delivers real-time updates to the online population at their fingertips [1]. These freely available news platforms result in an excess of information and have created an extraordinary interest in news aggregation and browsing. Despite headlines focusing on key ideas, readers often face the challenge of reading lengthy articles in their quest for knowledge which is time-consuming. Specially this information overload misguides the readers when multiple news sources report the same incident with varying levels of detail [2]. The widely employed text summarization technique effectively addresses the challenges faced by news article readers, especially beneficial for those with limited time seeking a quick overview of a news story. The development of extractive [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14] and abstractive [12], [15], [16], [17], [18], [19], [20], [21], [22] Automatic Text Summarization (ATS) models has been a crucial point for researchers since the 1950s [3], [4], [5], [6]. These traditional ATS approaches have several weaknesses. Occasionally, the sentences may adhere to syntactic structure, but semantic meaningfulness is not guaranteed as adjacent sentences in the summary may not always correspond to the original text [3]. Further, the importance of rare words may not exclusively rely on frequency, a phrase associated with infrequent words might be decisive in summary generation [3]. However, the improvements with general sequential models [1]] overcome the above weaknesses, the abstracts generated using these models lack contextual information, hindering higher-level abstraction. Unfaithful content and near-extractive summaries due to the limitations of model structure and lack of semantic interpretation over the input are common pitfalls in sequential models [22]. Identifying entities across multiple sentences and articles further complicates sequential models' effectiveness [1],[14]. The abstractive summarization is more attentive due to the typically abstractive nature of human-generated summaries. Further classification of abstractive models is twofold: structure-based and semantic-based where semantic-based approaches provide more coherent, information-rich, and well-structured abstracts than structured-based approaches. Hence, semantic graph-based approaches have shown success [12], [15], [16], [17], [18], [19], [20], [21]. Current machine learning-based approaches have greatly improved the presentation accuracy, but they encounter challenges in handling contextual information and extensive datasets [12], [13]. These observations highlight the importance of developing precise, efficient, and well-structured abstractive text summarization approaches combining structured and semantic representation, particularly in the context of news summarization from a set of articles related to the same news incident.

A knowledge Graph (KG) is an efficient and powerful knowledge representation technique in the research community. Recent research efforts in knowledge representation using KGs incorporate background knowledge and domain specifics related to the source text, infer additional knowledge using rich and explicit semantics, and possess the logic of human reasoning which is more in line with human reading habits [2], [12], [14], [23], [24], [25]. Even though the currently popular knowledge graphs on the Web: Google Knowledge Graph, Freebase, YAGO2, NELL, and DBpedia contain millions and billions of entities and facts, most of them could not sustain the requirements in question answering, text summarization, recommendation systems, and other web search applications [14]. Also, such large-scale KGs are not scalable to com-

Correspondence: Thilini Lakshika (E-mail: tlv@ucsc.cmb.ac.lk)
Received: 16-06-2024 **Revised:** 12-08-2024 **Accepted:** 09-09-2024
Thilini Lakshika, Amitha Caldera are from University of Colombo School Of Computing (tlv@ucsc.cmb.ac.lk, hac@ucsc.cmb.ac.lk)
DOI: https://doi.org/10.4038/icter.v18i2.7297
The 2025 Special Issue contains the full papers of the abstracts published at the 24th ICTer International Conference.

pact and extend event-specific information present in news articles based on user queries [2]. Furthermore, mining and searching such large-scale KGs is challenging under real-world resource constraints such as memory and query response time [24]. Summarization using KGs employs traditional ranking algorithms based on Term Frequency-Inverse Document Frequency (TF-IDF) scores, heuristic rules, Depth First Search (DFS), Breadth First Search (BFS) algorithms, and PageRank algorithm to select the subset of features in the KG where top-k features are used to generate the summary [12], [14], [18], [19], [20], [21], [23]. However, these traditional ranking algorithms do not consider the semantic relationships between nodes, and they consume more memory and time, with high computational complexity. Due to these complications, ATS using KG-based approaches to support the decision-making process is largely unexplored. While many pieces of research focus on building general-purpose and domain-specific KGs [12], [25], [26], [27], [28], [29], [30] for efficient querying and storage, such KGs are neither user-driven nor flexible to changes in the data, both of which are important in the real world.

Knowledge discovery techniques including association rules play a crucial role in uncovering significant features, patterns, dependencies, and associations among terms in text documents. Hence, we focus on addressing the existing research gaps in generating highly abstractive textual summaries for web news articles using scalable KGs and association rules. Following the identified research gaps, our study centers around the main research question (RQ): How can the effectiveness of KG ranking algorithms, particularly in the context of news summarization, be enhanced to generate abstracts. We broke down the primary research question into the following sub-research questions.
RQ1: How can the performance of traditional ranking algorithms be improved by incorporating knowledge derived from association rules?
RQ2: How can the performance of traditional ranking algorithms be enhanced through the integration of syntactic structure within textual content?

This paper presents our research contributions through the introduction of a novel KG ranking algorithm that incorporates advance techniques including the knowledge derived from Association Rules Mining in data mining to identify the best subset of features in the KG for inclusion in the abstractive summary. Our study investigates whether the knowledge representation in a KG can be utilized as an efficient approach for generating highly abstractive ATS for news events in web news articles. The underlying process presented in this paper consists of three high-level steps: (1) construction of a KG to visualize the essence of the news event published in web news articles, (2) identification of the prominent elements in the generated KG using a KG ranking algorithm, and (3) finally, generation of an abstractive summary for the news event. The KG-based abstractive ATS approach presented in this paper would be helpful for web newspaper readers to promptly comprehend the dominant idea of a news event using a collection of news articles. Our approach sees the urgent need for a knowledge-based text summarization approach that can automatically extract, represent, and summarize the information presented in a collection of e-news articles.

To answer these RQs, the rest of the paper is organized as follows: We describe the related works in section II and a further discussion on the knowledge representation using KGs in section III. The discussion on Association Rule Mining using the FP-Growth algorithm is presented in section IV. The formulation of our KG-based abstract generation pipeline is elaborated in section V. The experiments and results are presented in section VI followed by a discussion in section VII. Finally, all the conclusions, limitations, and future directions are described in Section VIII.

II. Related work

A. Automatic Text Summarization (ATS) using graphs

The ATS approaches based on clusters, templates, ontologies, semantic graphs, machine learning, fuzzy logics, and neural networks have proven to be very useful over 50 years up to now [3], [4], [5], [6], [16], [18], [20]. Among these, semantic graph networks have been proven to be very successful over other methods [17], [18], [19], [20], [21], [31], [32]. Even though the currently popular machine learning approaches have greatly improved the presentation accuracy of text summarization compared to traditional methods [12], [14] they are poorly accommodated with the contextual information to acquire higher-level abstraction [12], [14] and also unable to meet the requirement of large data sets [14]. Even though the abstractive summary generation from sequence-to-sequence models has been studied broadly, those summaries normally suffer from bogus contents [22].

Graphical knowledge representation using graphs has contributed much to developing knowledge-based applications. The Knowledge Base (KB) is the firstly used structure with knowledge-based systems for reasoning and problem solving. Later, frame-based language, rule-based, and hybrid representations were used for knowledge representation [33]. Following the first release of WordNet in 1995, several other open datasets with general ontological knowledge that are openly published and maintained by communities or research institutions were published including DBpedia, YAGO, Freebase, NELL, and Wikidata. For the time being, many general KGs, commercial datasets, and domain-specific KBs have been released to facilitate the research community.

The recent research efforts in text summarization using KG-based approaches [2], [12], [14], [23], [24], [25], [26] have been proven to be very successful over other methods since KGs can be incorporated with the background knowledge and domain specifics related to the source text. Furthermore, KG can possess the logic of human reasoning and the capability of inferring additional knowledge using rich and explicit semantics that align with human reading habits. Even though the currently popular KGs on the web including Google Knowledge Graph, Freebase, YAGO2, NELL, and DBpedia contain millions of entity instances and facts, they cannot endure the requirements in web news summarization as they are not supportable to compact or extend with event-specific information in news articles [14].

B. Knowledge Representation Using Knowledge Graph

A KG is viewed as a graphical representation of structured human knowledge [26] which requires a precise understanding of text content in a document and accurate modeling of cross-paper relationships. The main elements in a KG which are used to represent the logical structure of knowledge expressed in natural language, are nodes and edges. A novel semantic graph approach in [34] represents document nouns as nodes and captures semantic relationships between them as edges. Furthermore, the Predicate-argument structures (PASs), extracted through semantic role labeling (SRL) also used to form nodes, while semantic similarity weights from PAS relationships serve as edge weights [17]. KGs facilitate the integration of more context including heterogeneous information, intelligence, rich ontologies and semantics for knowledge acquisition and representation, and multi-lingual knowledge. Recent advancements in KGs have focused on statistical relational learning [35] and several emerging topics such as Knowledge Graph refinement [2], [12], [22], [25], [27], [28], [30], [36], [37], [38], knowledge reasoning [39], Knowledge Representation Learning (KRL) [2], [25], [35] and Knowledge Graph Embedding (KGE) [40], [41], [42] all of which have proven extremely useful for a wide range of knowledge-aware applications and graph analysis tasks.

News documents are rich in technical terms, abbreviations, and more importantly, domain-specific entities and relations which play a vital role in conveying crucial information. The state-of-the-art summarization approaches treat all text units equally, which inevitably ignore the salient information of some less frequent technical terms and abbreviations. Additionally, the relationships among topic-related documents, such as sequential, parallel, complementary, and contradictory connections are particularly important in multi-document summarization [41]. Document content modeling, cross-paper relationship identification, and precise modeling of cross-paper relationships are the main issues in multi-document summarization [41]. Multi-Document Scientific Summarization (MDSS) [41] framework addresses the above-mentioned issues by leveraging salient text units, entities, and relations using clusters of topic-relevant scientific papers. The Relationaware-related work Generator (RRG) [39], is a sequence-to-sequence model designed for identifying cross-paper relationships, yet it exhibits limitations in effectively identifying cross-paper relationships.

Although recent research efforts address the effective storage of knowledge in KGs and querying, existing methods lack user-driven flexibility and struggle to accommodate continuous data updates [24], [27], [43], [42]. Typically, large-scale KGs consist of millions of entities and facts describing these entities. Hence, exploring techniques for entity-based text summarization is another challenging necessity. The novel approach; FACeted Entity Summarization (FACES) [13], [14] summarizes a single entity by considering the popularity, uniqueness, and diversity of the facts selected for the summary while the RElatedness-based Multi-Entity Summarization (REMES) approach [13], [14] generates summaries at multiple entity levels.

C. Summary generation using Knowledge Graphs

The major challenge in automatically generating summaries from the knowledge representation in a KG of domineering features for summary generation. The selection of a subset of features (nodes and edges) from the KG can be introduced as a ranking problem where the selected top-k features are used for the summary generation [12], [14], [18], [19], [20], [21], [23]. Initial graph-based approaches feat relationships and associations between sentences in text documents using content similarity measures, Term Frequency - Inverse Document Frequency (TF-IDF) measures, sets of manually generated heuristic rules [20], and human-labeled training sets that don't consider semantic relationships between sentences in the original text. Google's PageRank and Hyperlink-Induced Topic Search (HITS) algorithms are two frequently used effective graph-based ranking algorithms in Web-link analysis and social networks [17]. The PageRank algorithm performs poorly with news article summarization since the PageRank values favor older web pages over current events as they have more associations with other web pages. Distance-based scoring functions which measure the plausibility of facts in the graph network and semantic similarity-based scoring functions which measure the plausibility of facts by semantic matching [26] are the two typical types of scoring functions. More frequently used degree centrality-based ranking algorithms rank the graph nodes based on the number of connections (edges) coming towards the node and nodes with more connections are considered more important. Similarly, the ranking algorithms that consider the betweenness centrality measure how often a particular node appears on the shortest paths between other nodes. These pure ranking algorithms are less effective in news article summarization because ranking alone cannot determine what kinds of features are selected for the summary [14].

The recent developments in graph-based summarization approaches [34] use many properties of the text content to construct the graph and their graph ranking approach gives more priority to the semantic importance of each word in the document. Further, they [34] consider the closeness centrality and eccentricity of a graph node to enhance the keyword extraction criterion. Numerous graph-based ranking algorithms have addressed abstractive summarization for individual documents, with limited attention to multi-document summarization [17]. Modified Weighted Graph-based Ranking Algorithm (MWGRA) [17] is an example of multi-document summarization that incorporates edge weights in graph vertices for the ranking process. This algorithm uses Jiang's measure which depends on WordNet to calculate the semantic similarity scores for each pair of PASs in the semantic graph. However, the semantic relations in WordNet don't capture the semantic relationships associated with proper nouns, and temporal and location arguments [17]. The personalized KGs [27] group nodes and edges into "super nodes" and "super edges," which yields promising results in adaptive summarization. The RDF-based Query-oriented summarization approach presented by [36] identifies the nodes with shared ingoing and outgoing edges as equivalence classes and groups those as supernodes. Even though this approach lowers overall storage costs, the modeling errors are high. The FACES approach [14] in single-entity summarization considers the popularity, uniqueness, and diversity of facts to identify imperative facts. This approach combines semantic expansion with hierarchical incremental clustering to conceptually group facts, employing Information Retrieval (IR) techniques to rank facts and finally select those with the highest ranking for the summarization. This approach performs beyond the syntactic similarity measures and especially it adds diversity to entity-based summarization making them more inclusive. The REMES approach [14] in multi-entity summarization concurrently processes facts belonging to the given multiple entities and combines graph-based relatedness, semantic expansion, and combinatorial optimization techniques to generate summaries. The ASGARD framework [22] is trained to identify essential content for abstract generation through the alignment of graphs with human-written summaries and further utilizes the document and graph encoders to generate abstracts. The MDSS [41] model follows a two-stage decoding approach to make better use of the information included in the KG for the summary generation. The PAS-based semantic graph approach [17] uses language generation models to generate summary sentences from the top-ranked PASs.

Inspired by recent progress in automatic summary evaluation, most of the KG-based abstractive summarization approaches [12], [17], [19], [22], [31], [32], [41], [44] apply automatic evaluations based on ROUGE metrics for the performance evaluations of the model generated summary. In addition to automatic evaluations, several approaches such as the ASGARD framework [22], Multi-Source Transformer summarization [37], MDSS [41], and RRG [39] have also been evaluated through human assessments. Generating abstractive summaries from information-sparse Wikipedia articles is quite challenging since Wikipedia text does not have predefined summary sentences or highlighted sentences [37]. However, the abstractive summarization approach presented in [37] uses the Wiki-Sum dataset, a dataset derived from English Wikipedia pages that is suitable for multi-document abstractive summarization for evaluation purposes. Many of the summarization approaches [12], [22], [37] conduct their model evaluations and performance comparisons on the CNN/DailyMail dataset and NYT (New York Times) Corpus, which is a popular, standard, and easily accessible text summarization dataset. The improved semantic graph approach for MDS [31], Genetic Semantic Graph Approach [17], and COMPENDIUM text summarization [19] assess its model using DUC 20022, which is a benchmark dataset that includes text documents with both human-made extractive and abstractive summaries.

D. Association Rule Mining for Textual Feature Extraction

Over the past decade, there has been significant progress in Data Mining techniques for analyzing data from diverse perspectives to unveil interesting relationships and knowledge. Among them, Association Rule Mining (ARM) using text documents is a more frequent and domineering research approach for finding out the most significant patterns and features in the text documents while lessening the time for reading all the documents [45], [46], [47], [48], [49], [50], [51], [52], [53], [54], [55]. Applications of the ARM technique span across many domains as per the demand and nature of the problem. Many such applications focus on refining and developing highly efficient novel algorithms in mining association rules from text documents. Apriori and FP-Growth are frequently used ARM algorithms while literature documents the application of Eclat, Tertius Association Rules, CLOSET+, CLOSET, and FP-CLOSE for mining sequential and emerging patterns from textual documents.

The application of ARM to the newspaper domain is very useful since it helps readers easily discover important information without having to read the entire article. The study in [51] performs the Knowledge Discovery from web articles using the GARW (Generating Association Rule using Weighting Scheme) algorithm which works similarly to Apriori but is found to be better than the Apriori algorithm. This study uses a set of web news articles related to diseases to mine association rules. Another experimental study [45] conducted using web news articles discusses the performance of two most important algorithms, Apriori and FP Growth. A recent study on MDS using Fuzzy Logic and Association Rule Mining [56] is very related to our study and it has used association rules extracted from the Apriori algorithm along with linguistic and statistical features of the sentences for the summarization. Another summarization approach [55] which summarizes transaction datasets of network traffic into a compact version uses the frequent itemsets generated from the apriori algorithm.

Owing to the shortcomings in applying traditional Apriori algorithms to bulk data, recently, many researchers have focused on refinements to improve the effectiveness of ARM. The novel algorithm, FEM (FP-growth & Eclat Mining) [45] which utilizes both FP-tree (Frequent-Pattern tree) and TID-list (Transaction ID list) data structures have proven their efficiency in mining short and long patterns from both sparse and dense datasets. Another study presented in [48] deviated from traditional ARM algorithms and presented a fuzzy extended Boolean framework for identifying association rules in a text document and used the generated rules for query refinement in Information Retrieval. The DCI CLOSED [50], which is a novel and scalable algorithm based on the divide-and-conquer approach has been used to mine closed frequent itemsets using the bitwise vertical representation of the database. Apriori MSG-P [53] is another modified version of the traditional Apriori algorithm for discovering hidden information in ARM which has proven its efficiency using an operational database at a hospital. The novel concept of rough association rules [54] which consists of a set of terms and a frequency distribution of terms is capable of exploring more specific information than normal association rules.

III. KNOWLEDGE REPRESENTATION USING A KNOWLEDGE GRAPH

A KG uses nodes and edges for the structured representation of facts, entities, relationships, and semantic descriptions [2], [14], [23], [24], [26], [40]. There are no proper definitions for a KG. Several studies have given definitions for KG by relating critical characteristics of semantic representations in text. We outline a Knowledge Graph, KG = E, R, F where E, R, and F are sets of entities, relations, and facts, respectively. A fact, $f \in F$ is denoted as a triple or triad (h, r, t) where h, r, and t are head, relation, and tail, respectively. The nodes and edges are used to represent the triples in the form of subject–predicate–object (SPO), where the subject and object are entities, while the predicate is the relationship amongst those entities. The smallest KG which has only two nodes and one edge is interpreted as a triple. Different representations and modeling instruments including entities, classes, relationship types, categories, ontologies, and free text descriptions are used in KGs [2], [12], [14], [23], [24], [25]. The real-world objects and abstract concepts are mapped into these modeling instruments which are represented using nodes. The relationships between the above-mentioned, semantic descriptions, and properties are represented using edges with a well-defined meaning [26].

Based on the Resource Description Framework (RDF), the knowledge represented in a KG can be expressed with factual triads in the form of Head-Relation-Tail, Entity-Relationship-Entity or Subject-Predicate-Object (SPO). A KG has two identical layers where the data layer represents domain specifics using entities and their descriptions while the schema or ontology layer on top of the data layer defines relationships of the entity descriptions and their classes. Furthermore, the schema or ontology layer represents the hierarchy of the relationships and classes, relationship types, categories, free text descriptions, rich and explicit semantics to infer additional knowledge, facts (features) about the concepts, controlled vocabularies, taxonomies, schemas, ontologies and it can be linked to the web. The techniques used in KG construction can be classified into five categories [39]: knowledge extraction, knowledge representation learning, knowledge mining, knowledge fusion, and knowledge reasoning. Knowledge Graph Completion (KGC), triple classification, entity recognition, relation extraction, relation classification, and open knowledge enrichment are several knowledge acquisition tasks in KG construction [2], [26].

IV. ASSOCIATION RULE MINING USING THE FP-GROWTH ALGORITHM

A. FP-Growth algorithm for text mining

ARM using text documents has become an interesting approach to finding out the most important information in the text documents including text patterns and features [45], [47], [51], [57]. Recent studies use numerous data mining techniques including classification, clustering, regression analysis, and ARM to discover knowledge from bulk data using novel approaches. Many of such studies [49], [50], [53], [54], [56], [45], [58] represent refinements to the existing popular ARM algorithms including Apriori and FP-Growth. The FP-Growth algorithm is traditionally used for mining frequent itemsets in transactional databases, but recently it has been successfully applied to text data mining as well. Compared to the Apriori algorithm, the FP Growth algorithm constantly executes better. The experimental study conducted using web news articles [45] has proven that the FP-Growth algorithm performs better than the Apriori algorithm by overcoming the major weaknesses in the traditional Apriori algorithm such as several transactional database scans, higher execution time, and large memory consumption. More importantly, the FP Growth algorithm performs well disregarding the number of textual contents included in web news articles [45]. Since the FP-Growth algorithm has been proposed as an alternative to the Apriori algorithm, its efficiency in mining frequent item sets is also high [58], [59], [60], [61].

The generation of conditional pattern base, generation of conditional FP-Tree, and identification of frequent itemsets are the major steps in the FP-Growth algorithm. Following the required text pre-processing techniques, the algorithm mines the Frequent Itemsets in text documents. The FP-Growth algorithm requires several parameters where users have the flexibility to set the parameter values based on their requirements. The minimum support (min sup) determines the minimum frequency threshold an item must meet to be considered frequent, whole minimum confidence (min con) sets the minimum confidence threshold for an association rule to be considered interesting, other key parameters include the minimum length (min len) and maximum length (max len) of a frequent itemset. Users may also specify the number of top frequent itemsets they need to return from the algorithm to return. Since the user-defined values for the above parameters have a significant impact on the final results of the ARM process, it is recommended to experiment with different parameter combinations to identify the optimal value set.

B. Association Rule Generation using Frequent Itemsets

Association rules represent ”one implies the other” or ”occur together” among keywords within an indexed document. These rules provide valuable insights into the relationships between words in the text documents using the occurrence of one set of items associated with the occurrence of another set of items. The Association rule generation process requires frequent itemsets consisting of high-frequency keywords in the indexed documents. Given a collection of keywords A = w1, w2,..., wn and a collection of indexed documents D = d1,d2,..., dm, where each document di is a collection of keywords such that $d_i \subseteq A$. Let Wi be a set of keywords. A document di is said to contain Wi if and only if $W_i \subseteq d_i$. The typical form of an association rule is presented as $W_i \Rightarrow W_j$where Wi and Wj are sets of keywords in a collection of indexed documents such that $W_i \subseteq A$, $W_j \subseteq A$, and $W_i \cap W_j = \emptyset$. Wi is also known as the antecedent whereas Wj is the consequent. The rule can be explained as ”if Wi occurs, then Wj is likely to occur.” The importance of an association rule is measured by three key measures: support(s), confidence(c) and lift (l) [45], [47], [62]. If the support value is high, then the rule applies to a significant portion of the dataset. If the confidence value is high, then the reliability of the rule is high. There is a higher likelihood of Wj occurring when Wi is present. The process involving the ARM consists of two steps. The first step is to find frequent itemsets where the support value of all the itemsets is above the minimum support value. The second step is to generate the association rules using the output of step 1 while discarding the rules below the minimum confidence value. Here, the two threshold values, minimum support, and minimum confidence are the constraint values defined by the user. The first step in ARM requires more time and effort, but the second step is more forthright.

V. RESEARCH METHODOLOGY

Our research approach to abstract generation using web news documents illustrated in Fig. 1 consists of five phases. The source texts relevant to KG construction were taken from the Document Understanding Conference (DUC) – 2002 dataset for multi-document summarization.

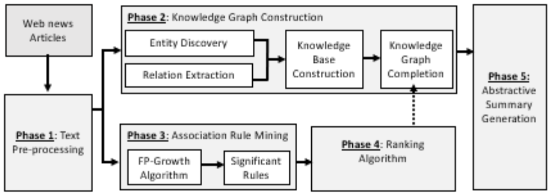

Web news Articles
Phase 1: Text Pre-processing
Phase 2: Knowledge Graph Construction
Entity Discovery
Relation Extraction
Knowledge Base Construction
Knowledge Graph Completion
Phase 3: Association Rule Mining
FP-Growth Algorithm
Significant Rules
Phase 4: Ranking Algorithm
Phase 5: Abstractive Summary Generation

Fig. 1: The overall construction process of KG-based abstractive summary generation

Phase 1: Text pre-processing
We applied major text pre-processing tasks including lowercasing, tokenization, contraction expandition, punctuation removal, stop-word removal, special characters removal, and lemmatization for the identification of linguistic elements and patterns within the text data.

Phase 2: Knowledge acquisition for KG Construction
Building the KG involves a series of sequential operations arranged in a pipeline, and we adopted the methodology outlined in [2]. The specific knowledge acquisition tasks used in this approach are divided into three categories as below.

- Entity discovery - Identification of entities from the source document.
- Relation extraction - Discover the relational facts in the source document.
- KG Completion – Construction of the KG by merging triples and relation path reasoning.

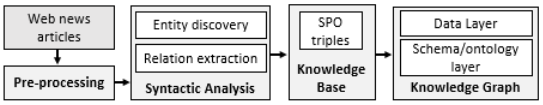


Fig. 2: The overall process of knowledge acquisition for KG generation

The news article contents underwent additional processing using co-reference resolution and pronominal reference resolution [18], [20] with spaCy libraries. Next, the deep syntactic analysis steps illustrated in Fig. 2 are applied to extract essential information, such as entities, relationships between entities, and attributes. Attributes are properties associated with entities and relations. Based on the output, we identify the logical meaning of document sentences and extract logical form triads. The deep syntactic analysis process uses several NLP techniques such as Named Entity Recognition (NER), Part of Speech (POS) tagging, and entity disambiguation. A triple is a combination of subject, predicate, and object (SPO) in a sentence where the subject and object are considered as the entities that are involved in a relationship defined by the predicate. The spaCy libraries were employed to transform every sentence in the source document, encompassing subject, object, and predicate into a triple and subsequently stored within the Knowledge Base (KB) [2]. The SPO triples are used to construct the data layer in the KG and this process involves two tree data structures: a constituency parse tree, which breaks down a sentence into sub-phrases, and a dependency parse tree, which analyzes the grammatical construction of a sentence. The schema or ontology layer in the KG is constructed with the automatic hypernym detection using the Hearst-pattern-based methods [2], [61] which is one of the most influential approaches for identifying hierarchical level relationships in text contents. Once the KB is populated with the triples which represent the two main layers in the KG, the construction of the KG is done by merging the triples, predicting links and relationships. Modeling of the cross-paper relationships in multiple documents related to the same news event happened through entity interactions and information aggregation. This process focus on inferencing novel relation paths.

Phase 3: Association Rule Mining

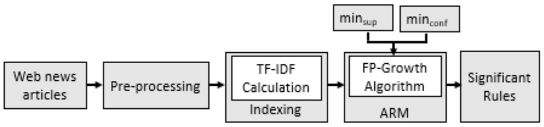


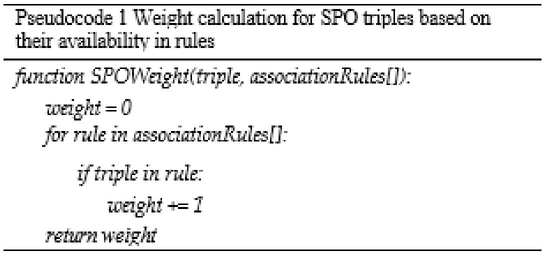
Fig. 3: The overall process in association rule generation

Referring to the depiction in Fig. 3, the collection of e-news articles was prepared for the ARM process following the basic text pre-processing steps described in Phase 1. Our approach uses the TF-IDF numerical statistic to generate index terms from the pre-processed web news articles and the FP Growth algorithm for mining association rules since it is faster than the traditional Apriori algorithm [58], [59], [60], [61]. The transaction generation process treats each document as a transaction, where the terms or words are the items in the transaction. The algorithm first converts the text documents into a highly condensed conditional FP-Tree and later extracts the frequent itemsets. Secondly, it uses the identified frequent keyword sets in step 1 to generate the association rules by discarding those rules below minimum confidence.

Phase 4: KG ranking algorithm This phase describes our novel solution for the RQs on enriching the accuracy of the KG ranking algorithm for generating a better abstractive summary. The selection of a subset of domineering features from the KG for inclusion in summary generation is the major challenge in KG-based text summarization. Since there are several KG ranking algorithms, the decision to select the most suitable ranking algorithm depends on the application domain and specific characteristics of the knowledge graph. We proposed six different KG ranking algorithms based on combinations of three diverse components, degree centrality, SPO availability in association rules, and association rule-based statistics to measure the plausibility of factual triples in the KG. The degree of centrality used in traditional KG ranking algorithms provides an insight into the prominence of nodes in a KG based on the number of relationships they have. We consider this measure which is calculated using Equation 1 as our first component since it highlights the entities that play key roles.

$$\text{Degree Centrality} = \frac{\text{total number of nodes}}{\text{number of edges connected to node}} \tag{1}$$

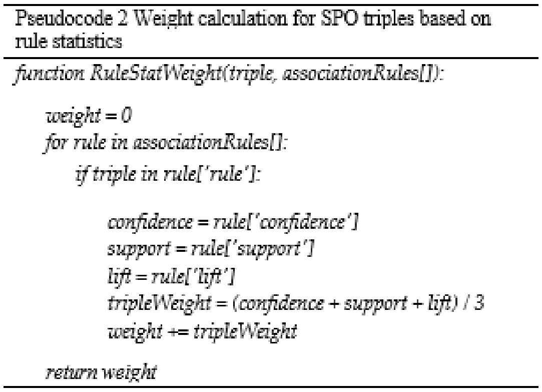
Since the subject, predicate, and verb are essential components of a sentence that play distinct roles in conveying meaning and structuring language, next we consider the availability of SPOs in the filtered association rules. All the filtered association rules and each triple in the KB are considered for the weight calculation illustrated in Pseudocode 1.



Pseudocode 1 Weight calculation for SPO triples based on their availability in rules

```
function SPOWeight(triple, associationRules[]):
    weight = 0
    for rule in associationRules[]:
        if triple in rule:
            weight += 1
    return weight
```

Further, we consider additional factors, such as support, confidence, and lift level of the association rules to calculate a more sophisticated weight. The third component calculates a weight for each triple using pseudocode 2 to statistically analyze and understand the patterns and relationships in filtered association rules.



Pseudocode 2 Weight calculation for SPO triples based on rule statistics

```
function RuleStatWeight(triple, associationRules[]):
    weight = 0
    for rule in associationRules[]:
        if triple in rule['rule']:
            confidence = rule['confidence']
            support = rule['support']
            lift = rule['lift']
            tripleWeight = (confidence + support + lift) / 3
            weight += tripleWeight
    return weight
```

The composition of the above components is explained in Table 1. Algo 0 considers only the degree of centrality which is the traditional graph ranking approach. Except for Algo 1 and Algo 3, each algorithm listed in Table I provides an average weight value for every triple in the KB. The normalized weights are utilized to facilitate a more effective comparison across all algorithms.

TABLE I: The composition of components in the designed KG ranking algorithms

	Proposed KG Ranking Algorithms						
Component	Algo 0	Algo 1	Algo 2	Algo 3	Algo 4	Algo 5	Algo 6
Degree centrality	x	-	x	-	-	x	x
SPO availability in rules	-	x	x	-	x	-	x
Association rule-based statistics	-	-	-	x	x	x	x

Phase 5: Abstractive Summary Generation The final phase in our approach is to generate abstractive summaries using the salient triples identified through the KG ranking algorithm. We used the language model, NLG-Py (Natural Language Generation in Python) for natural language abstract generation. It provides a set of modules for generating sentences from structured data, making it suitable for converting triples into abstractive sentences. The selection of triples from the KG for abstract generation is determined by the compression rate. Given that the DUC-2002 dataset comprises human-generated abstracts with word limits of 50, 100, and 200 words, we consider these specific word limits as the compression rates.

VI. Experimental Results

A. Environment and Dataset

The minimum requirement for the implementation of KG, ARM using FP Growth algorithm, and natural language generation is Python version - 3.7.0 using a workstation with Intel(R) Core (TM) i3-8130U CPU @ 2.20 GHz and 4 GB main memory. All the experiments described in this paper were done using a benchmark dataset that includes a collection of e-news documents in the English language released by the DUC – 2002. This dataset specifically relates to the task of extractive and abstractive summarization and it includes text documents with both human-made extractive and abstractive summaries. The dataset contains 128 event-related e-news article sets where each set consists of an average of 10 articles. Each e-news article contains a minimum of 10 sentences and no specific maximum length.

B. KG construction using e-news articles

The major steps in the KG constructions are described using the news articles included in the Hurricane [d061jb] news document collection taken from the DUC – 2002 dataset. We apply the specific knowledge acquisition tasks to discover entities and extract relationships in the e-news articles following the text pre-processing steps described under Phase 1. An example sentence included in the Hurricane [d061jb] news document collection, DUC – 2002 dataset is presented below.

> “Prime Minister Edward Seaga of Jamaica alerted all government agencies, saying Sunday night: Hurricane Gilbert appears to be a real threat and everyone should follow the instructions and hurricane precautions issued by the Office of Disaster Preparedness in order to minimize the danger”.

The NLP techniques such as NER and POS tagging are applied to each sentence in the news article collection to extract the combination of subject, object, and predicate (SPO) in the form of triples and store those within the Knowledge Base (KB). Furthermore, the sub-phrases in each sentence were analyzed using the constituency parse tree. Grammatical structure of a sentence was analyzed using the dependency parse tree. Later, the SPO triples stored in the KB are used to construct the data layer in the KG. The hierarchical level relationships in the e-news articles are extracted to build up the schema layer in the KG using the automatic hypernym detection using the Hearst-pattern-based methods [2], [61]. Once the SPO triples which represent the two main layers in the KG stored in the KB, the construction of the KG (Fig. 4) is done by merging the triples in the KB, Meanwhile, this process predicts novel links between entities, identifying novel relationships and model the inter-document relationships related to the collection of e-news articles.

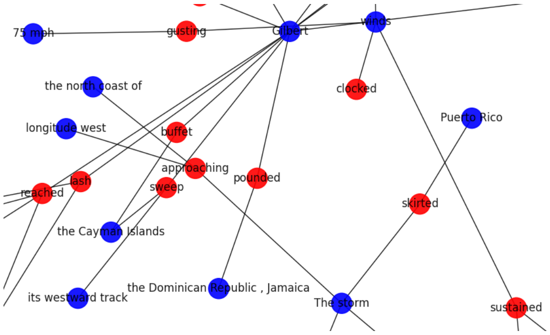
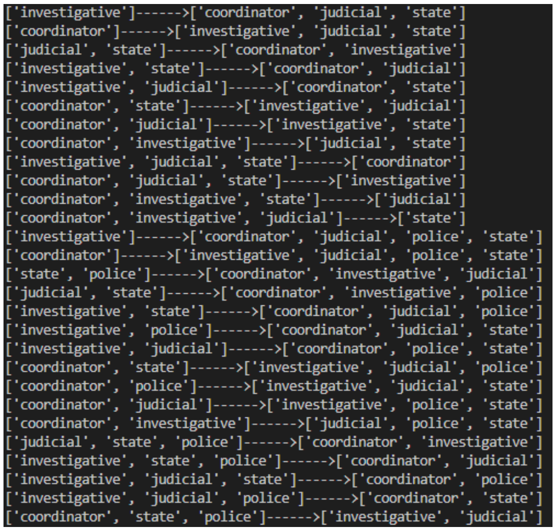


Fig. 4: An extract from the constructed KG for Hurricane news document collection – DUC 2002

C. ARM using the FP-Growth algorithm

The experimental design for ARM described in this study was carried out using a list of three topics related to web news article sets named Hurricane [d061jb], Bomb Blast [d063j], and McDonald [d064jb]. Following the required text pre-processing steps, the index terms that are above the threshold value of 0.6 for the TF-IDF score were extracted to represent each e-news article. We conducted the experiments on rule generation by manipulating the number of top N terms with the highest TF-IDF values which occupied the final list of keywords used in the ARM. Based on the experimentation results in [45] for selecting the most suitable algorithm for ARM using e-news articles, we applied the FP-Growth algorithm for ARM. Our approach considers each sentence in the web news article collection as a database transaction. Depending on the average number of words in a sentence, we manipulated the number of keywords used to represent a sentence for generating association rules. An extract from the list of association rules generated for the document collection [d061jb], containing 17 keywords is displayed in Fig. 5..



```
['investigative']------>['coordinator', 'judicial', 'state']
['coordinator']------>['investigative', 'judicial', 'state']
['judicial', 'state']------>['coordinator', 'investigative']
['investigative', 'state']------>['coordinator', 'judicial']
['investigative', 'judicial']------>['coordinator', 'state']
['coordinator', 'state']------>['investigative', 'judicial']
['coordinator', 'judicial']------>['investigative', 'state']
['coordinator', 'investigative']------>['judicial', 'state']
['investigative', 'judicial', 'state']------>['coordinator']
['coordinator', 'judicial', 'state']------>['investigative']
['coordinator', 'investigative', 'state']------>['judicial']
['coordinator', 'investigative', 'judicial']------>['state']
['investigative']------>['coordinator', 'judicial', 'police', 'state']
['coordinator']------>['investigative', 'judicial', 'police', 'state']
['state', 'police']------>['coordinator', 'investigative', 'judicial']
['judicial', 'state']------>['coordinator', 'investigative', 'police']
['investigative', 'state']------>['coordinator', 'judicial', 'police']
['investigative', 'police']------>['coordinator', 'judicial', 'state']
['investigative', 'judicial']------>['coordinator', 'police', 'state']
['coordinator', 'state']------>['investigative', 'judicial', 'police']
['coordinator', 'police']------>['investigative', 'judicial', 'state']
['coordinator', 'judicial']------>['investigative', 'police', 'state']
['coordinator', 'investigative']------>['judicial', 'police', 'state']
['judicial', 'state', 'police']------>['coordinator', 'investigative']
['investigative', 'state', 'police']------>['coordinator', 'judicial']
['investigative', 'judicial', 'state']------>['coordinator', 'police']
['investigative', 'judicial', 'police']------>['coordinator', 'state']
['coordinator', 'state', 'police']------>['investigative', 'judicial']
```

Fig. 5: An extract from the list of association rules generated for the document collection [d061jb] – DUC 2002

D. KG ranking algorithms

We designed six different KG ranking algorithms combining three diverse components, degree centrality, SPO availability in association rules, and association rule-based statistics. These algorithms were used to measure the plausibility of factual triples in the KG and to generate abstractive summaries using the highest-ranked triples. The composition of the components in each algorithm is explained in Table 1. Algorithm 1 calculates a score for each triple in the KG considering only the availability of each SPO triple in the generated rules. Algorithm 2 is an enhanced version of algorithm 1 which considers the graph property, and node degree along with SPO availability in association rules. Algorithm 3 calculates a score for each triple in the KG based on three factors: the number of rules a triple is visible, the number of rules where the SPO in triples appears on the RHS side of the ruleand is preceded by words on the LHS and the confidence value of such rules. The combination of the two components, SPO availability in rules and association rule-based statistics are considered in Algorithm 4, and the combination of degree centrality and association rule-based statistics is considered in Algorithm 5. Algorithm 6 is the combination of all three components.

E. Abstractive summary generation

The generic human-written abstractive summaries in the DUC-2002 dataset provide four types of summaries based on the number of words present in the summary. We have considered the abstracts with 50,100 and 200 words for each document and we omitted the 10-word abstract since it took the form of a headline. Our model generates three abstracts by compressing the information in the original web news articles into 50, 100, and 200-word abstracts.

TABLE II: The experimental setup for selected news topics in the DUC 2002 dataset

Collection ID	# documents	# database transactions	# average words in a sentence	# keywords	Min. Sup.	Min. Con.	# total Rules
[d061jb]	4	170	20	17	50	80	1542
				20	50	80	1563
				23	50	80	1598
[d063j]	7	195	22	19	50	80	2224
				22	50	80	2251
				25	50	80	2278
[d064jb]	5	95	31	28	50	80	1475
				31	50	80	1626
				34	50	80	1870

We have conducted the experiments manipulating the variables; the number of keywords for frequent itemset generation, minimum support, minimum confidence for rule generation, and the compression rate (number of words present in the model-generated abstracts). Every experiment was repeated with each KG ranking algorithm presented in Table I. The outcomes obtained from several experiments conducted with the manipulated variables, as depicted in Table II, are presented in this paper.

TABLE III: Variable values in the test case 1

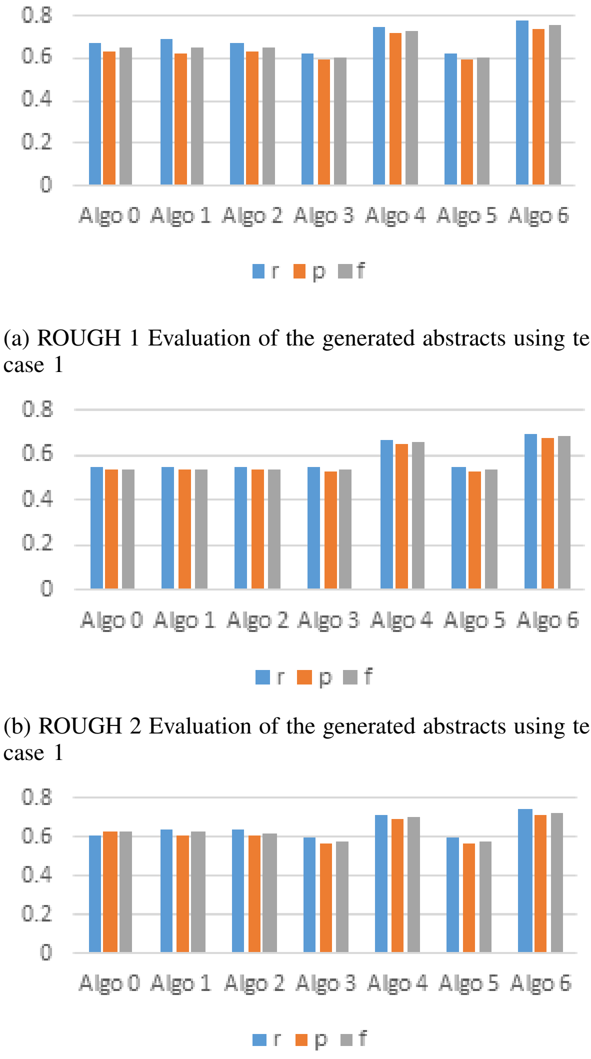
Collection ID	# keywords	Min. Sup.	Min. Con.	# words in model-generated abstracts	Ranking Algorithm
[d061jb]	17	50	80	50	Algo 6

The 50-word model-generated abstract derived from the KG for test case 1 in Table III is presented below.

> Hurricane Gilbert, classified as a category 5 storm, resulted in death, extensive flooding, and significant damage as it moved through the Caribbean Islands and moved toward the Yucatan Peninsula. After skirting various island nations, it has caused substantial loss of life and damage in Jamaica. Subsequently, it beat the Yucatan Peninsula before scattering.

The automatic evaluation of the model-generated abstract was conducted using the ROUGH 1, ROUGH 2, and ROUGH L measures [12], [17], [19], [22], [31], [32], [41], [44]. Recall (r), Precision (p), and F1-measure (f) values were computed for each ROUGH measure, by comparing model-generated abstracts to a 50-word human-generated abstract from DUC-2002 which was used as the reference summary for evaluation.

Our model generated a 50-word abstract for test case 1 using six different algorithms. Subsequently, we calculated ROUGH measures for each summary produced by the model, assessing the quality and effectiveness of the generated summaries. Similarly, we generated a 50-word abstract using Algo 0 to represent the pure ranking algorithms described in the literature. The received results for test case 1 are presented in Table IV. The detailed comparison of the ROUGH 1, ROUGH 2, and ROUGH L measures in Table IV are illustrated in Fig. 6.



(a) ROUGH 1 Evaluation of the generated abstracts using test case 1

(b) ROUGH 2 Evaluation of the generated abstracts using test case 1

(c) ROUGH L Evaluation of the generated abstracts using test case 1

Fig. 6: ROUGH evaluation of the generated abstracts using test case 1

Since algorithm 6 shows a significant improvement in the ROUGH measures compared to the other algorithms, we further compared the ROUGH measures of model-generated abstracts with 50, 100, and 200 words (Table V).

The detailed comparisons of the results received for test case 2 are illustrated in Fig. 7.

VII. Discussion

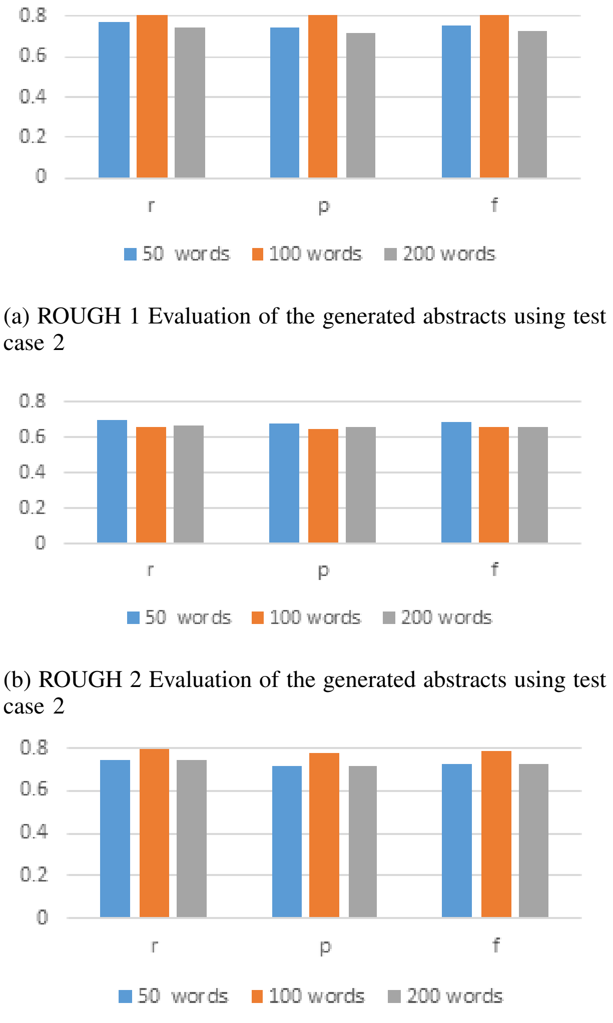
We assess the effectiveness of our proposed model on a set of selected news document collections in the DUC-2002 dataset for abstract generation. The analysis done based on the experimental results received for test case 1 shows that the

Measure	Algo 0			Algo 1			Algo 2			Algo 3			Algo 4			Algo 5			Algo 6		
	r	p	f	r	p	f	r	p	f	r	p	f	r	p	f	r	p	f	r	p	f
ROUGH 1	0.6707	0.6346	0.6505	0.6951	0.6270	0.6538	0.6707	0.6346	0.6505	0.6220	0.5926	0.6053	0.7450	0.7156	0.7283	0.6220	0.5926	0.6053	0.7730	0.7436	0.7563
ROUGH 2	0.5426	0.5323	0.5367	0.5426	0.5323	0.5367	0.5426	0.5323	0.5367	0.5426	0.5294	0.5348	0.6656	0.6524	0.6578	0.5426	0.5294	0.5348	0.6936	0.6804	0.6858
ROUGH L	0.6109	0.6237	0.6311	0.6431	0.6048	0.6321	0.6419	0.6058	0.6217	0.5932	0.5638	0.5765	0.7162	0.6868	0.6995	0.5932	0.5638	0.5765	0.7442	0.7148	0.7275

TABLE IV: Comparison of various algorithms based on precision (p), recall (r), and F-measure (f).

TABLE V: Variable values in the test case 2

Collection ID	# keywords	Min. Sup.	Min. Con.	# words in model-generated abstracts	Ranking Algorithm
[d061jb]	17	50	80	50,100,200	Algo 6



(a) ROUGH 1 Evaluation of the generated abstracts using test case 2

(b) ROUGH 2 Evaluation of the generated abstracts using test case 2

(c) ROUGH L Evaluation of the generated abstracts using test case 2

Fig. 7: ROUGH evaluation of the generated abstracts using test case 2

KG ranking algorithm 6 has the best performance over other algorithms. Also, algorithm 4 outperforms other algorithms securing the next best ranking. Based on the complete analysis done with all the test cases in Table II shows that the KG ranking algorithm 6 has the best performance and algorithm 4 has the next best ranking. This makes the broad conclusion that Algorithm 6 consistently excels in abstract generation using the DUC-2002 dataset and it is important to consider specific variations in performance across different datasets.

After establishing Algorithm 6 as the top performer, our next motive was to examine how the number of words appearing in the model-generated summary impacts its effectiveness. Both ROUGH 1 and ROUGH L measures show strong performance with the 100-word abstract while demonstrating less effectiveness in the case of 50 and 200-word abstracts. The ROUGH 2 measures show strong performance with the 50-word abstract while demonstrating equal effectiveness in the case of 100 and 200-word abstracts. On average, it can be concluded that Algorithm 6 shows strong performance in generating 100-word abstracts.

Finally, we explored the influence of the number of keywords on frequent itemset generation when abstracts are generated with Algorithm 6. Both 100 and 200-word abstracts generated with 20 keywords on frequent itemsets perform well over 50-word abstracts. Abstracts generated with 17 keywords on frequent itemsets had a good performance in 50-word abstracts while abstracts generated with 20 keywords on frequent itemsets had a poor performance. The average number of words in a sentence in the test case 1 equals 20. Similarly model generated abstracts using 20 keywords on frequent itemsets perform well over other keyword combinations. So, it concludes that abstract generation performs well when the number of keywords on frequent itemsets equals the average number of words in a sentence.

The experiment results show that the novel KG ranking algorithm with all three components produces significantly higher ROUGE scores than the remaining algorithms we experienced. We also proved that the compression rate of 100 words in the final abstract has a significant involvement in obtaining a better performance in abstract generation. Furthermore, abstract generation performs well when the number of keywords on frequent itemsets equals the average number of words in a sentence in the news article collection. The illustration of the experimental results shows that our proposed

KG-based abstract generation model achieves considerable improvement compared with the baselines. All these conclusions are specific to the DUC-2002 dataset and variations in performance may occur when applying the same methods to different datasets.

VIII. CONCLUSIONS AND FUTURE WORKS

While answering the main RQ, we implemented an accurate KG ranking algorithm for generating better abstracts in the context of news summarization. Our experiments were focused on six different KG ranking algorithms to bridge the knowledge gap between NLP and Data Mining fields. In addressing RQ1, we employed association rule-based statistics, an effective and influential factor in extracting insights from association rules to enhance the efficacy of traditional KG ranking algorithms. Next, we considered the SPO availability in the association rules, which is an essential component of a sentence that plays distinct roles in conveying meaning and structuring language to answer RQ2. Algorithm 4, a combination of rule statistics and SPO availability, outperforms algorithms 1 and 3, each of which considers only individual components. Algorithm 6, which integrates the knowledge derived from association rules along with degree centrality and SPO availability in association rules is capable of taking out the best subset of triples from the KG for abstract generation.

The main contribution of this study is the novel KG ranking algorithm which is based on knowledge derived from association rules that play an important guiding role in selecting the best subset of triples for final abstractive summary generation. Our experimental results have proven that a KG ranking algorithm that considers different textual features performs better than the existing pure ranking algorithms. The automatic evaluation results on the DUC-2002 dataset show the superiority of our approach in generating abstracts for e-news articles. However, variations in model performance may occur when applying the same methods to different datasets on news article summarization.

In the future, we would like to expand our experiments on model evaluations and performance comparisons using the CNN/DailyMail dataset and NYT Corpus, which is a popular, standard, and easily accessible text summarization dataset [12], [22], [37]. In this study, we mainly focused on general-purpose summary generation for the entire population of e-news readers. Moreover, the integration of personalization to enhance the summarization approach by adjusting the ranking algorithms is another future direction. Finally, we are planning to expand our model evaluations by manipulating the parameters required for association rule generation and the human evaluations for the model-generated abstracts with the support of domain experts.

REFERENCES

[1] (2023) Digital 2023: Sri lanka. [Online]. [Online]. Available: https://datareportal.com/reports/digital-2023-sri-lanka

[2] M. V. P. T. Lakshika and H. A. Caldera, "Knowledge graphs representation for event-related e-news articles," *Machine Learning and Knowledge Extraction*, vol. 3, no. 4, pp. 802–818, Sep. 2021.

[3] G. Sharma and D. Sharma, "Automatic text summarization methods: A comprehensive review," *SN Computer Science*, vol. 4, no. 1, p. 33, Oct. 2022.

[4] R. Chettri, "Automatic text summarization," *International Journal of Computer Applications*, vol. 161, no. 1, pp. 5–7, Mar. 2017.

[5] N. Bhatia and A. Jaiswal, "Trends in extractive and abstractive techniques in text summarization," *International Journal of Computer Applications*, vol. 117, no. 6, pp. 21–24, May 2015.

[6] D. K. Gaikwad and C. N. Mahender, "A review paper on text summarization," *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, vol. 5, no. 3, pp. 154–160, Mar. 2016.

[7] D. Das and A. F. T. Martins, "A survey on automatic text summarization," p. 32, Nov. 2017, literature Survey for the Language and Statistics II course at CMU 4.

[8] A. R. Pal, P. K. Maiti, and D. Saha, "An approach to automatic text summarization using simplified lesk algorithm and wordnet," *International Journal of Control Theory and Computer Modelling*, vol. 3, no. 5, pp. 15–23, Sep. 2013.

[9] K. Sarkar, "Automatic single document text summarization using key concepts in documents," *Journal of Information Processing Systems*, vol. 9, no. 4, pp. 602–620, Dec. 2013.

[10] A. Patil, "Automatic text summarization," *International Journal of Computer Applications*, vol. 109, no. 17, pp. 18–19, Jan. 2015.

[11] M. S. A. Babar and M. Tech-CSE, "Text summarization: An overview," p. 6, 2013.

[12] P. Wu, Q. Zhou, Z. Lei, W. Qiu, and X. Li, "Template oriented text summarization via knowledge graph," in *International Conference on Audio, Language and Image Processing (ICALIP)*. IEEE, Jul. 2018, pp. 79–83.

[13] K. Gunaratna, "Semantics-based entity summarization," p. 5.

[14] K. Gunaratne, "Semantics-based summarization of entities in knowledge graphs," Ph.D. dissertation, Wright State University, Dayton, OH, USA, 2017, ph.D. dissertation.

[15] P. Kouris, G. Alexandridis, and A. Stafylopatis, "Abstractive text summarization based on deep learning and semantic content generalization," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2019, pp. 5082–5092.

[16] M. Subramaniam and V. Dalal, "Test model for rich semantic graph representation for hindi text using abstractive method," *International Research Journal of Engineering and Technology (IRJET)*, vol. 02, no. 02, pp. 113–116, May 2015.

[17] A. Khan, N. Salim, and Y. J. Kumar, "Genetic semantic graph approach for multi-document abstractive summarization," in *Fifth International Conference on Digital Information Processing and Communications (ICDIPC)*. Sierre, Switzerland: IEEE, Oct 2015, pp. 173–181.

[18] D. Rusu, M. Grobelnik, and D. Mladenić, "Semantic graphs derived from triplets with application in document summarization," *Informatica*, vol. 33, pp. 357–362, Nov 2008.

[19] E. Lloret and M. Palomar, "Analyzing the use of word graphs for abstractive text summarization," in *IMMM 2011 : The First International Conference on Advances in Information Mining and Management*, 2011, pp. 61–66.

[20] J. Leskovec, M. Grobelnik, and N. Milic-Frayling, "Learning substructures of document semantic graphs for document summarization," in *LinkKDD 2004*, Aug 2004.

[21] F. Liu, J. Flanigan, S. Thomson, N. Sadeh, and N. Smith, "Toward abstractive summarization using semantic representations," *arXiv preprint arXiv:1805.10399*, May 2018, accessed: Feb. 08, 2020. [Online]. Available: http://arxiv.org/abs/1805.10399

[22] L. Huang, L. Wu, and L. Wang, "Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2020, pp. 5094–5107.

[23] K. Hulliyah and H. Kusuma, "Application of knowledge graph for making text summarization (analizing a text of educational issues)," in *Proceeding of the 3rd International Conference on Information and Communication Technology for the Moslem World (ICT4M)*. Jakarta, Indonesia: IEEE, Dec 2010, pp. E79–E83.

[24] Q. Song, Y. Wu, and X. Dong, "Mining summaries for knowledge graph search," in *2016 IEEE 16th International Conference on Data Mining (ICDM)*, Barcelona, Spain, Dec 2016, pp. 1215–1220.

[25] S. Elhammadi and et al., "A high precision pipeline for financial knowledge graph construction," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain: International Committee on Computational Linguistics, Dec 2020, pp. 967–977.

[26] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition and applications," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2021.

[27] L. Faber, T. Safavi, D. Mottin, E. Müller, and D. Koutra, "Adaptive personalized knowledge graph summarization," in *Proceedings of the 14th International KDD Workshop on Mining and Learning with Graphs (MLG)*, London, UK, Aug 2018.

[28] D. Nicholson and C. Greene, "Constructing knowledge graphs and their biomedical applications," *Computational and Structural Biotechnology Journal*, vol. 18, pp. 1414–1428, 2020.

[29] T. Safavi, C. Belth, L. Faber, D. Mottin, E. Muller, and D. Koutra, "Personalized knowledge graph summarization: From the cloud to your pocket," in *2019 IEEE International Conference on Data Mining (ICDM)*. Beijing, China: IEEE, Nov 2019, pp. 528–537.

[30] P. Wang, H. Jiang, J. Xu, and Q. Zhang, "Knowledge graph construction and applications for web search and beyond," *Data Intelligence*, vol. 1, no. 4, pp. 333–349, Nov 2019.

[31] A. Khan, N. Salim, and Y. Kumar, "Document abstractive summarization based on semantic role labelling," *Applied Soft Computing*, vol. 30, pp. 737–747, May 2015.

[32] W. Van Melle, "Mycin: a knowledge-based consultation program for infectious disease diagnosis," *International Journal of Man-Machine Studies*, vol. 10, no. 3, pp. 313–322, May 1978.

[33] M. Nickel, K. Murphy, V. Tresp, and E. Gabrilovich, "A review of relational machine learning for knowledge graphs," *Proceedings of the IEEE*, vol. 104, no. 1, pp. 11–33, Jan 2016.

[34] C. Yadav, A. Sharan, and M. Joshi, "Semantic graph based approach for text mining," in *2014 International Conference on Issues and Challenges in Intelligent Computing Techniques (ICICT)*. IEEE, Feb 2014, pp. 596–601.

[35] H. Paulheim, "Knowledge graph refinement: A survey of approaches and evaluation methods," *Semantic Web*, vol. 8, no. 3, pp. 489–508, Dec 2016.

[36] T. Chen, X. Wang, T. Yue, X. Bai, C. X. Le, and W. Wang, "Enhancing abstractive summarization with extracted knowledge graphs and multi-source transformers," *Applied Sciences*, vol. 13, no. 13, p. 7753, Jun 2023.

[37] S. Malviya and U. S. Tiwary, "Knowledge based summarization and document generation using bayesian network," *Procedia Computer Science*, vol. 89, pp. 333–340, 2016.

[38] R. Liu, R. Fu, K. Xu, X. Shi, and X. Ren, "A review of knowledge graph-based reasoning technology in the operation of power systems," *Applied Sciences*, vol. 13, no. 7, p. 4357, Mar 2023.

[39] Q. Wang, Z. Mao, B. Wang, and L. Guo, "Knowledge graph embedding: A survey of approaches and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 12, pp. 2724–2743, Dec 2017.

[40] P. Wang *et al.*, "Multi-document scientific summarization from a knowledge graph-centric view," Sep 2022, accessed: Sep. 06, 2023. [Online]. Available: http://arxiv.org/abs/2209.04319

[41] Université Paris-Sud and Université Rennes, "Query-oriented summarization of rdf graphs," *Proceedings of the VLDB Endowment*, vol. 8, no. 12, pp. 2012–2015, Sep 2015.

[42] X. Chen *et al.*, "Capturing relations between scientific papers: An abstractive model for related work section generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, 2021, pp. 6068–6077, online.

[43] A. Mohamed and S. Rajasekaran, "Improving query-based summarization using document graphs," in *2006 IEEE International Symposium on Signal Processing and Information Technology*. IEEE, 2006, pp. 408–410.

[44] T. Lakshika and A. Caldera, "Association rules for knowledge discovery from e-news articles: A review of apriori and fp-growth algorithms," *Advances in Science, Technology and Engineering Systems Journal*, vol. 7, no. 5, pp. 178–192, 2022.

[45] J. Manimaran and T. Velmurugan, "A survey of association rule mining in text applications," in *IEEE International Conference on Computational Intelligence and Computing Research*. Enathi, Tamilnadu, India: IEEE, 2013, pp. 1–5.

[46] H. Mahgoub, D. Rösner, N. Ismail, and F. Turkey, "A text mining technique using association rules extraction," *International Journal of Computer, Electrical, Automation, Control, and Information Engineering*, vol. 2, no. 6, pp. 2044–2051, 2008.

[47] M. Delgado, M. J. Martín-Bautista, D. Sánchez, J. M. Serrano, and M. Á. Vila, "Association rule extraction for text mining," in *Flexible Query Answering Systems*, J. G. Carbonell, J. Siekmann, T. Andreasen, H. Christiansen, A. Motro, and H. L. Larsen, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, vol. 2522, pp. 154–162.

[48] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference of Very Large Data Bases, VLDB*, Santiago, Chile, 1994, pp. 487–499.

[49] C. Lucchese, S. Orlando, and R. Perego, "Fast and memory efficient mining of frequent closed itemsets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 1, pp. 21–36, Jan 2006.

[50] M. Kulkarni and S. Kulkarni, "Knowledge discovery in text mining using association rule extraction," *International Journal of Computer Applications*, vol. 143, no. 12, pp. 30–35, Jun 2016.

[51] H. Mahgoub, "Mining association rules from unstructured documents," *International Journal of Applied Mathematics and Computer Sciences*, vol. 1, no. 4, pp. 201–206, 2008.

[52] T. Ouypornkochagorn, "Mining rare association rules on banpheo hospital (public organization) via apriori msg-p algorithm," *ECTI Transactions on Computer and Information Technology*, vol. 6, no. 2, pp. 156–165, Jan 1970.

[53] Y. Li and N. Zhong, "Rough association rule mining in text documents for acquiring web user information needs," in *2006 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06)*. Hong Kong, China: IEEE, Dec 2006, pp. 226–232.

[54] V. Chandola and V. Kumar, "Summarization - compressing data into an informative representation," in *5th IEEE International Conference on Data Mining (ICDM'05)*, vol. 1, no. 8, 2005.

[55] C. M. Rahman, F. A. Sohel, P. Naushad, and S. M. Kamruzzaman, "Text classification using the concept of association rule of data mining," in *Proceedings of the International Conference on Information Technology*, Kathmandu, Nepal, May 2010, pp. 234–241.

[56] G. Alaghband and L. Vu, "A fast algorithm combining fp-tree and tid-list for frequent pattern mining," in *Proceedings of Information and Knowledge Engineering*, 2011, pp. 472–477.

[57] J. Arora, "An efficient arm technique for information retrieval in data mining," *International Journal of Engineering Research*, vol. 2, no. 10, 2013.

[58] A. Kaur and G. Jagdev, "Analyzing working of fp-growth algorithm for frequent pattern mining," *International Journal of Research Studies in Computer Science and Engineering*, vol. 4, no. 4, pp. 22–30, 2017.

[59] M. Hahsler and R. Karpienko, "Visualizing association rules in hierarchical groups," *Journal of Business Economics*, vol. 87, no. 3, pp. 317–335, Apr 2017.

[60] O. Netzer, R. Feldman, J. Goldenberg, and M. Fresko, "Mine your own business: Market-structure surveillance through text mining," *Marketing Science*, vol. 31, no. 3, pp. 521–543, May 2012.

[61] M. A. Hearst, "Automatic acquisition of hyponyms from large text corpora," in *Proceedings of the 14th conference on Computational linguistics*. Nantes, France: Association for Computational Linguistics, 1992, p. 539.

[62] S. Malallah and Z. Ali, "Multi-document text summarization using fuzzy logic and association rule mining," *Journal of Al-Rafidain University College for Sciences*, no. 3, pp. 241–258, Oct 2021.