Kufa Journal of Engineering
Vol. 16, No. 2, April 2025, P.P. 215 -233
Article history: Received 21 June 2024, last revised 2 October 2024, accepted 2 October 2024

TOWARD SALIENT KEY PHRASE FOR CANDIDATE TOPIC DETECTION

Yasser Saad[1,*] and Wafaa Al Hameed[2]

[1]Software Department, College of Information Technology, University of Babylon, Babylon, Iraq, Email: yasseraadj.sw@student.uobabylon.edu.iq.

[2] Software Department, College of Information Technology, University of Babylon, Babylon, Iraq, Email: it.wafaa.mohammed@uobabylon.edu.iq.

https://doi.org/10.30572/2018/KJE/160213

ABSTRACT

With exponential growth of digital information, the need for efficient methods for automatic keyphrase extraction has become increasingly important. Key phrase candidate topic detection (KPCTD) aims to automatically identify key phrases, i.e., phrases that capture the central meaning of a text document and associate them with their corresponding topics. We have developed an innovative method that combines statistical with contextual approaches (position and distance criteria in addition to semantic information). We present a comprehensive approach to text analysis; it enables the use of a harmonious mix of different features that allows for precise and effective extraction of relevant information. furthermore, for sifting the later extracted key phrases into condensed thematic (topic) key phrases written under (ABSTRACT) part, superiority of the various strategies is examined, such as approximate matching with key sentences at the beginning of the text, the identification of cluster foci, and the prioritization of frequent phrases. After extensive investigations on two datasets, semeval2017 and Inspec, the proposed PhraeRank approach outperforms the previous results. Quantitative metrics achieve a precision of 51.23% and a recall of 28.26% for top 5 keyphrases on the SemEval2017 dataset, and a precision of 47.89% and recall of 25.34% on the Inspec dataset. Additionally, value of a BLEU score is 0.62 on the SemEval 2017 dataset and 0.58 on the Inspec dataset. demonstrating significant improvement over existing methods. These results highlight the algorithm's ability to extract relevant information from text documents.

KEYWORDS

keyphrase extraction, NLP, information retrieval, topic keyphrase.

1. INTRODUCTION

keyphrase extraction is a crucial step in natural language processing (NLP) that enables the retrieval and summarization of information from large text datasets(Song, Feng and Jing, 2023). With the ever-growing wealth of digital content, more precise and effective approaches to keyword extraction are needed(Boudin, 2018). Conventional manual deletion methods are time-consuming and unproductive, especially considering the wealth of information available online(Sarwar, Noor and Miah, 2022). Therefore, the explore of keyphrase has become crucial for tasks such as sorting and organizing documents(Du et al., 2023). Several keyphrase extraction methods have been developed, including graph-based methods and unsupervised methods that use syntax. These strategies aim to recognise keyphrase and key phrases in the text to improve the search and distribution system. Typically, the filtering process involves segmenting the data into words and using methods such as N-Grammes (Chen et al., 2012). The target keyphrase are identified using Part-Of-Speech (POS)-based algorithms (Wu et al., 2005). Recent advances in the field of keyphrase extraction have led to the use of strong linguistic models such as BERT(Devika et al., 2021). KeyBERT, a technique that uses BERT embeddings, provides a simple method for extracting keyphrases and phrases from text(Song, Feng and Jing, 2022).

In addition, research is being conducted to refine classical algorithms such as TextRank (Mihalcea and Tarau, 2004)using deep learning techniques and semantic analysis offered by models such as BERT(Liu, Lin and Wang, 2021).These algorithms aim to improve the accuracy and relevance of keyphrase extraction, especially for scientific articles with important summaries (Papagiannopoulou and Tsoumakas, 2019). Linguistic features, context and domain-specific knowledge are all important factors that influence the performance of keyphrase extraction models, Research metrics and robust research prototype (Kong et al., 2023).

The proposed PhraseRank scoring is used to find salient key phrases by incorporating both statistical and contextual information. It operates by incorporating various metrics such as position, cosine similarity, and distance. The algorithm considers the position of keyphrases within the text (each keyphrase has a position weight related to its location). The distance metric measures how physically close the occurrences of the two keyphrases are within the text. emphasizing keyphrases that frequently appear near each other, It is consider all positions where each keyphrase appears and sums the inverse square of the differences between these positions. Cosine similarity evaluates the semantic similarity between keyphrases by comparing their vector representations, capturing how contextually similar the keyphrases are based on their

usage in the text. This ensures that keyphrases which are contextually similar are identified effectively. By integrating these metrics, PhraseRank constructs a graph where nodes represent keyphrases and edges signify their relationships, weighted by the combined measure of physical proximity and semantic similarity. This graph-based ranking algorithm, inspired by Google's PageRank, ranks the keyphrases to highlight the most important ones based on their contextual and positional relevance within the text.

Our study also focuses on extracting important key phrases to represent topics. To achieve the latter, we investigate methods such as approximate matching with key sentences at the beginning of the text, identifying central points in key phrase clusters Text document clustering (TDC) is an essential unsupervised learning technique in text mining, crucial for categorizing documents into meaningful groups based on content similarity (Abasi, Khader and Al-Betar, 2022), and prioritizing frequent keys. This method ensures precise and effective extraction of relevant information and identification of key themes and concepts from ABSTRACT part of the document.

2. RELATED WORKS

Patel and Cornelia Caragea (2017) introduce positionrank, an unsupervised method for keyphrase extraction from scholarly documents that incorporates information from all positions of a word's occurrences into a biased pagerank. The approach depends on where words are placed, which might not grasp how words or phrases relate in meaning . (Patel and Caragea, 2017).

Florian Boudin (2018) came up with a keyphrase extraction model that doesn't need supervision. It uses a graph with multiple parts to capture topic-related details in documents (Florescu and Caragea, 2017). His model captures the mutual reinforcement between keyphrase candidates and their associated topics, the ranking and selection of candidates by integrating a novel mechanism for adjusting edge weights based on keyphrase selection preferences.the complexity of constructing multipartite graphs can increase computational load. Secondly, the model less effective in handling texts with non-traditional or unclear topic interrelations(Boudin, 2018).

Krutarth Patel and Cornelia Caragea (2021) created kprank, a method that automatically finds important phrases in scientific papers without needing any guidance. This method uses a special type of graph and considers where words are placed in the text and what they mean in context. It uses a modified version of the pagerank algorithm that focuses on these aspects. Their method uses scibert, which is a tool that understands the meaning of words based on the context they

are used in, along with paying attention to where these words appear in the text.(Patel and Caragea, 2021).

Haoran Ding and Xiao Luo (2022) created agrank, a model that uses graphs and deep learning to find important phrases in text without needing labels. It combines techniques from deep learning and attention mechanisms from a pre-trained BERT model. This model creates a graph of potential keyphrases, enhanced with nodes that represent overall and specific contexts. Adding these context nodes means more adjustments are needed, which could lead to the model fitting too closely to the data and being easily affected by changes in the data.(Ding and Luo,2022).

Hung Du and Srikanth Thudumu et al. (2023) introduced ContextualRank, a method that doesn't need supervision to find important phrases in text. It works with layered diagrams that show meaning and considers both general ideas and the situation when deciding how important each phrase is. However, using a system called Hierarchical Topic Modeling (HTM) to create topics from these key phrases might not work as well because different key phrases chosen by people might actually be about the same topic.(Du et al., 2023).

Shengbin Liao and Zongkai and others (2023) have developed a new method called topiclprank, which enhances the existing topicrank method by combining information about the length and position of keyphrases. This new approach builds on non-supervised techniques by highlighting the significance of the structural characteristics of keyphrases and prompts a reevaluation of how keyphrases are valued in text analysis, moving away from just frequency-based methods. Although this method prioritizes structural aspects over frequency-based ones, it might not fully consider the semantic connections and relationships among keyphrases, which could result in less effective keyphrase extraction in some situations(Liao et al., 2023).

Alexander Tsvetkov and Alon Kipnis (2023) introduce entropyrank, an modern unsupervised approach for keyphrase extraction leveraging the ideas of facts principle. By exploiting a pre-trained language version, entropyrank evaluates phrases based on their conditional entropy, specializing in terms that decrease entropy in textual content compression contexts (Tsvetkov and Kipnis, 2023).

3. METHODOLOGY

Proposed method extracts keyphrases from textual content the use of advances in Natural Language Processing (NLP) and unsupervised mastering procedures. We comprise semantic context by way of the use of BERT embeddings, which give deeper contextual representations of terms. This permits for more correct identity of keyphrases We also introduce positional

weighting, giving higher importance to phrases in prominent positions within the text, reflecting their likely relevance. Additionally, improving graph construction by weighting edges based on the cosine similarity, position and distance of BERT embeddings, ensuring stronger connections between semantically similar phrases. These modifications result in enhanced performance by extracting contextually relevant and accurately positioned keyphrases. The technique comprises several phases, as shown below:

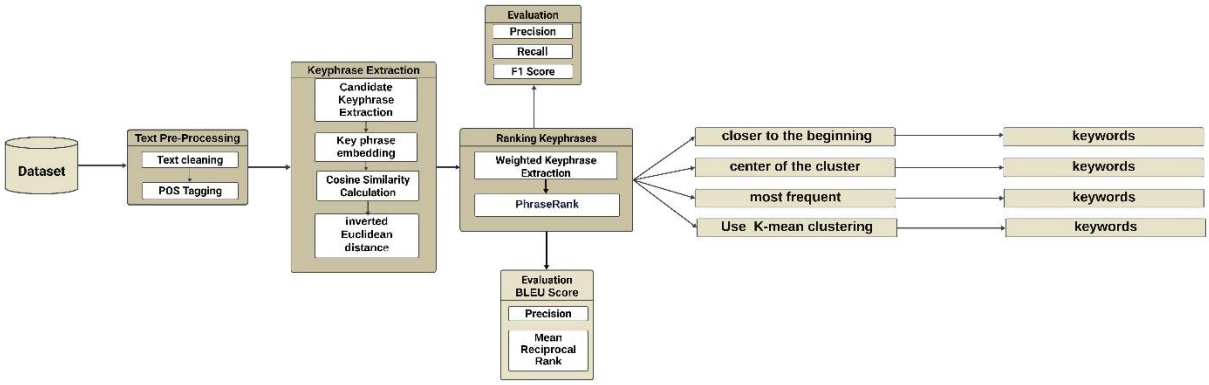

Fig 1: Candidate Topic Keyphrase Detection

3.1. Dataset

3.1.1. Inspec Dataset

Two datasets are used for developing and testing keyphrase extraction algorithms:

The Inspec dataset is commonly used in keyphrase extraction research. It comprises two main volumes: Volume 1: Text Files Containing abstracts **,**This volume contains text files where each file corresponds to a single document, which in this case, is an abstract of a journal article.Each text file includes a concise abstracts of the corresponding journal article. These abstracts are written in English and are generally concise, providing an overview of the main points of the article.There are a total of 2000 documents in the dataset, but a subset of 500 documents is often used for comparison with other models, as established by prior research. and Volume 2: Text Files Containing Main Phrases (Keyphrases) ,This volume contains text files where each file corresponds to the keyphrases associated with the summaries from the first volume. Each text file contains a list of keyphrases that have been identified as the most important phrases from the corresponding document abstracts. These keyphrases were manually extracted by experts, ensuring a high level of accuracy and relevance. The purpose of these keyphrases is to provide a benchmark against which the keyphrases extracted by automated methods can be evaluated.

3.1.2. SemEval 2017 Dataset

The SemEval 2017 dataset is another key dataset used for keyphrase extraction tasks, particularly in scientific domains. It also consists of two main volumes: Volume 1: Text Files

Containing Paragraphs. This volume consists of text files, with each file containing a paragraph extracted from a ScienceDirect journal article. The paragraphs come from articles in the fields of computer science, materials science, and physics.Each paragraph typically contains 176 tokens (words and punctuation), providing a short passage of text for analysis. The dataset includes 493 such paragraphs abstracts. 1.1.1.1 Volume 2: Text Files Containing Keyphrases This volume contains text files listing keyphrases extracted from each corresponding paragraph in the first volume. The keyphrases were annotated by students in the relevant fields (computer science, materials science, or physics) under the supervision of experts. Each paragraph contains, on average, 17 keyphrases, providing a rich set of phrases for evaluation purposes. The keyphrases serve as a ground truth for assessing the performance of automated keyphrase extraction systems.

3.2. Text Preprocessing

Text preprocessing is an essential step in preparing raw text data for natural language processing (NLP). It involves converting raw text into a clean and structured format, which facilitates more effective analysis and model performance. This process typically involves several key operations to improve the quality of the text data and make it suitable for further processing. It occurs in two stages as shown below.

a. Text cleaning: is required while preparing raw text for NLP, which comprises deleting unnecessary characters and formatting to optimize text analysis operations such as (“,’, , (,),{,}”…etc.) , The goal is to standardize the text and eliminate elements that might interfere with text analysis, thereby optimizing the text for further processing.

b. Word Tokenization and POS Tagging: The method of extracting vital phrases from textual content entails main steps: phrase tokenization and Part of Speech (POS) tags. Word tokenization separates sentences into person words, at the same time as POS tagging tags each word consistent with its syntactic characteristic, such as noun, verb, or verb. Tokenized Words: ["He", "loves", "programming"] and POS Tags: [Pronoun, Verb, Noun]. Phrase tokenization and POS tagging enhance textual content evaluation by using breaking down textual content into plausible gadgets and knowledge the grammatical roles of every word.

3.3. Keyphrase Extraction

Keyphrase extraction is a crucial project in herbal language processing (NLP) and facts retrieval, aiming to identify the maximum enormous and relevant terms inside a text. This technique entails several levels, beginning with candidate keyphrase extraction the use of a part of speech (POS) tags to identify potential keyphrases. These applicants are then embedded into

vectors the use of BERT, a deep learning model that captures contextual meanings. The semantic similarity among keyphrases is quantified the usage of cosine similarity, which aids in building a graph wherein nodes constitute keyphrases and edges represent their relationships. Finally, a graph-based totally ranking algorithm stimulated via Google's PageRank ranks the keyphrases, making sure that the most critical ones are highlighted based totally on their contextual and positional relevance inside the text.

3.3.1. Candidate Keyphrase Extraction

This stage uses part of speech (POS) tags to extract nouns and adjectives as possible keyphrases. Candidate keyphrases for the line "The quick brown fox jumps over the lazy dog" could include "quick brown fox" and "lazy dog," with a focus on nouns and adjectives. The prototype used to obtain a candidate key phrase:

$$(N)^* \mid (JJ)^* (N)^+$$

The prototype gives a framework for identifying key phrases in literature where ‘N’ and ‘JJ’ represent nouns and adjectives, respectively. (N)*: Matches any sequence of nouns, including no nouns at all, allowing for phrases like "dog" or "computer system".

(JJ)* (N)+: Matches sequences of zero or more adjectives followed by one or more nouns, useful for phrases such as "red", "old tree", or "small old house". (Du et al., 2023)

3.3.2. Key phrase embedding

BERT, a deep learning NLP model, encodes phrases as vectors including contextual meanings. Assume "bank" appears in various contexts ("river bank" vs. "bank account"), BERT's encoding will capture these distinctions via distinct vector representations(Florescu and Caragea, 2017).

3.3.3. Cosine Similarity

Cosine similarity is a widely used metric in text processing because it effectively measures the similarity between two text vectors by focusing on the cosine of the angle between them. This approach captures semantic similarity while being independent of the text length, making it ideal for comparing texts of varying sizes. Its efficiency and scalability further enhance its suitability for large datasets (Huang,2008). It is used to compare the semantic similarity of two keyphrases vectors representation using the following Eq.1. Fig.2 shown a Visualization of key phrase Embedding using BERT transformer.

$$\text{Cosine Similarity}\,(K1_{Bert}, K2_{Bert}) = \frac{k1_{bert} \cdot k2_{bert}}{\|k1_{bert}\| \|k2_{bert}\|} \qquad (1)$$

$K1_{bert} \cdot k2_{bert}$ are the vectors representing the two sets of embeddings or features. $K1_{bert} \cdot k2_{bert}$is the dot product of the vectors. This measures the extent to which the vectors align with each other.$\| k1_{bert} \| \| k2_{bert} \|$ are the magnitudes (norms) of the vectors.

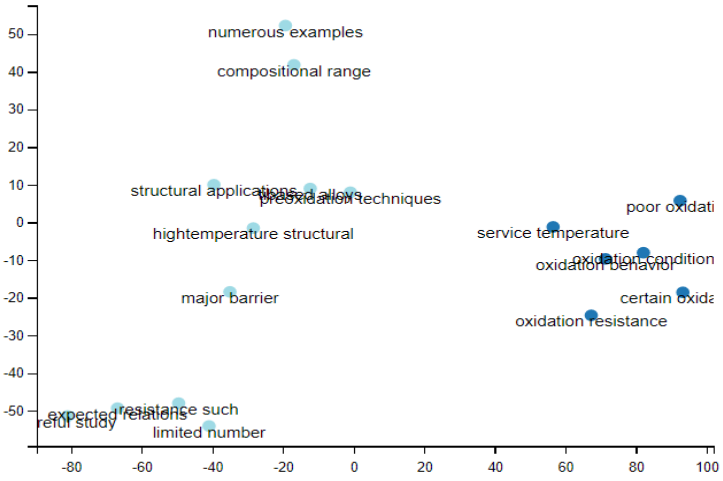

Fig 2. Visualization of key phrase Embedding using BERT transformer

3.3.4. Inverted Euclidean Distance

Inverted Euclidean distance is a measure used to transform the traditional Euclidean distance into a form that increases as the points are closer together. Then $Distance(c_k, c_l)$ represents a distance metric between the elements c_k *and* c_l benefits form positions feature value (Du et al., 2023)

$$\text{Dist}ance(c_k, c_l) = \sum_{p_x \in position(ck)} \sum_{p_y \in position(cl)} \frac{1}{|(px-py)^2|} \quad (2)$$

px: Represents a position in the text where the keyphrase ck appears.

py: Represents a position in the text where the keyphrase cl appears.

The positions px and py are essential for accurately determining the proximity of keyphrases, contributing to a more contextually relevant ranking of keyphrases in the text.

3.4. keyphrases Ranking

It is a critical step in natural language processing and information retrieval since it finds the most essential and relevant phrases in a text source. In this step conducted a weighted keyphrase extraction method that assigns values to words or phrases based on criteria such as frequency and relevance. The proposed computes the edge strength of any two keyphrase in the graph consider both statistical and contextual features.

3.4.1. Weighted Keyphrase Extraction

Weighted keyphrase extraction is a technique used to identify the most significant phrases in a document by considering both their importance and their relationships to each other. The edge weight between keyphrases can be calculated using a combination of distance and cosine similarity measures. The formula given for the edge weight is:

$$\text{EdgeWeight}(c_k, c_l) = \text{Distance}(c_k, c_l) \times \text{Cosine}(c_k, c_l) \quad (3)$$

By multiplying the distance metric with cosine similarity, the EdgeWeight provides a combined measure of physical proximity and semantic similarity. This ensures that keyphrases which are both close in the text and contextually similar are given higher importance, leading to a more accurate and contextually relevant ranking of keyphrases.In the context of similarity and ranking, Cosine Similarity is chosen for its ability to measure the orientation or direction of vectors, making it ideal for text and sparse data by focusing on relative similarity regardless of magnitude. Inverse Distance emphasizes proximity, giving higher similarity scores to closer points or values, which is useful for spatial and numerical contexts. Combining these metrics balances directionality and closeness, providing a comprehensive ranking based on both similarity and proximity. Fig 3. Demonstrate the weighed graph, where each node corresponds to the keyphrase connected with another keyhrase by weighted edges using Eq. 3.

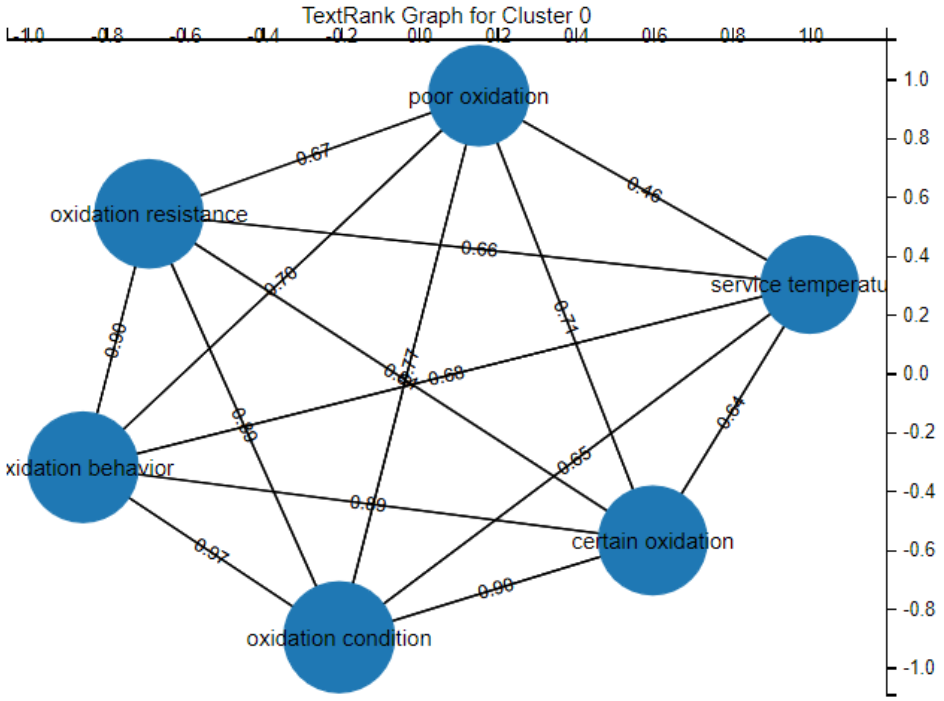

Fig 3. Weighted graph weight for candidate keyphrases for simple example

3.4.2. keyphrase ranking

A graph-based ranking algorithm inspired by Google's pagerank is used. This method visualizes text as a graph with nodes representing words or phrases and edges representing their relationships to determines node relevance by ranking keyphrases based on pagerank principle. The score equation used in (Du et al., 2023). was used in this research. Fig.4 display the most significant keyphrase result according to the scores.

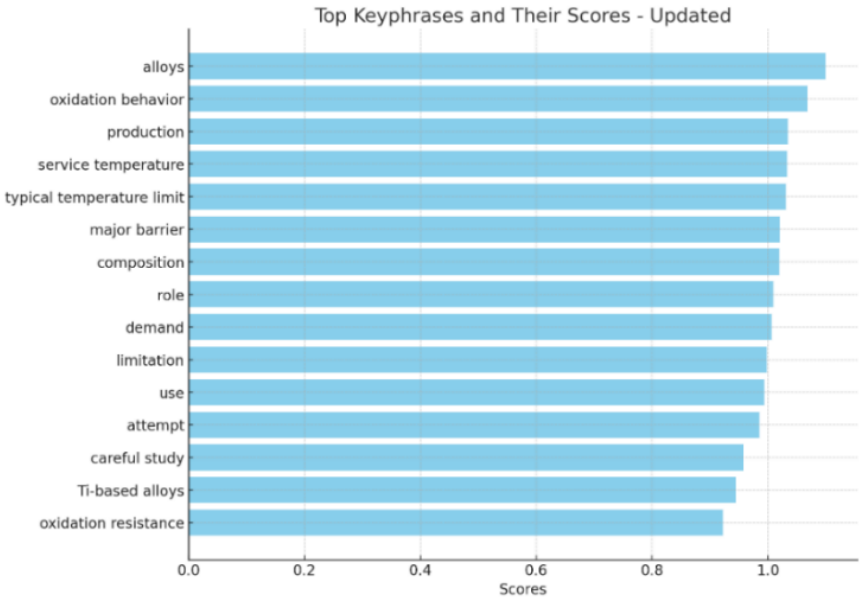

Fig 4 Top keyphrase and scores example first document in semevail2017

3.5. Candidate topic keyphrase detection

In a detailed experiment, work out important sentences that represent the essence of the research by proposing some strategies, each strategy representing a separate method that extracts candidate topic keyphrases separately from the other. These words are usually written underneath the summary. We propose the following strategies:

3.5.1. Relevant to the Key Sentence

In this strategy, we analyze the key sentences at the beginning of the text. It is assumed that important topics are introduced early in the document. First, the embedding vector of the key sentence is compared with a list of keyphrase that were previously extracted. Then the closest is marked as a topic keyphrase candidate as shown in the Fig.5 a Candidate topic Relevant to the Key Sentence. This approach helps quickly identify significant topics based on their early mention, potentially improving the relevance of extracted keyphrases.

3.5.2. Topic As Center keyphrase

In this strategy, we used the graph matrix of the previously extracted keyphrase to calculate the strength of the connections between them all. For each topic candidate, select the topic with the highest connection strength to other keyphrases by calculating the difference between the current expression and the other expressions and then adding the differences to obtain a score. Finally, sort the topic candidates according to their connection strength using the following equation (4). As shown in the Fig.6 Most close keyphrase from neighbors.

$$Candidate\ Topoc(\ k_i) = \text{MaxScore} \sum_{j=1}^{n} Sum\left|k_i - k_j\right| \qquad \ldots\ldots(4)$$

Where **Score(ki)** is the total score for the topic candidate ki, representing its connection strength based on its differences with other keyphrases. And,

$|k_i - k_j|$ is the absolute difference between the candidate keyphrase ki and another keyphrase kj. This measures how distinct ki is from kj. The proposed method based on Centrality Emphasis which identifies keyphrases based on their centrality within the network of keyphrases, ensuring that selected keyphrases are contextually relevant, central and provides a deeper understanding of keyphrase relationships, improving the identification of important topics.

Example input Text :-

Poor oxidation behavior is the major barrier to the increased use of Ti-based alloys in high-temperature structural applications. The demand to increase the service temperature of these alloys beyond 550°C (the typical temperature limit) requires careful study to understand the role that composition has on the oxidation behavior of Ti-based alloys [1–3]. The attempt to overcome this limitation in Ti-based alloys has led to the production of alloys with substantially improved oxidation resistance such as β-21S and also development of coatings and pre-oxidation techniques [1, 4–6]. While it is tempting to extrapolate the oxidation behavior (e.g. oxidation rate law, depth of oxygen ingress and scale thickness) observed for a limited number of compositions under a certain oxidation condition to a broader compositional range, there are numerous examples in the literature where deviations from the expected relations are observed [7,8].

- **candidate topic using Relevant To the Key Sentence**

Poor oxidation behavior , major barrier , use , Ti-based alloys , high-temperature structural applications,

Fig 5 Candidate topic Relevant to the Key Sentence

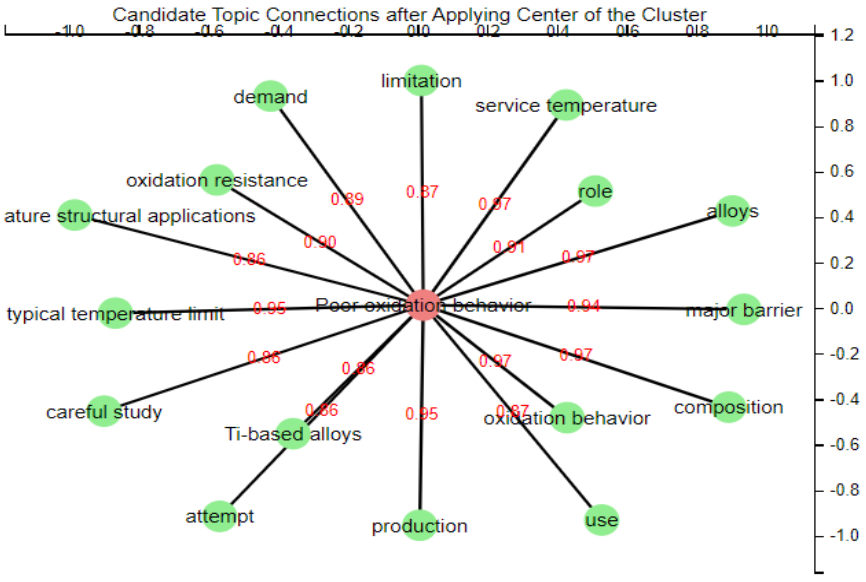

Fig 6. Most close keyphrase from neighbors

3.5.3. Most Frequent

In this strategy, we simply identify the most frequent keyphrase that we extracted from the document in the previous phase and assign them as topic candidates. A dictionary of topic candidates, paired with their frequency, sorted in descending order of frequency. As shown in the Fig.7. The proposed method based on prominent topics emphasis to highlights the most frequently mentioned keyphrases, often reflecting the core themes of the document.

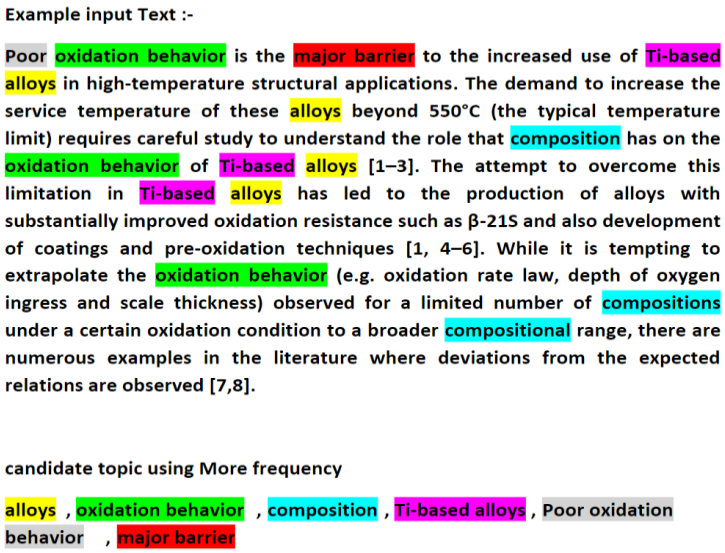

Fig 7. Most Frequent Candidate Topic

3.5.4. Topic keyphrase as a center using k-means

This strategy proposes to distribute the extracted key phrases across several clusters, considering the focus of each cluster as a candidate for a thematic key phrase. To achieve this, k-means technique was used due to its efficiency (Han, Kim and Choi, 2008). K-means clustering is aims to partition observations (keyphrases in this experiment) into k clusters, each observation belongs to the cluster with the closest mean value (cluster centers or cluster centroid), which serves as the prototype of the. The value of **k** determines the number of key phrases desired . The proposed method based on Cluster-Based Organization which groups keyphrases into clusters, with cluster centers representing central themes. This provides a structured approach to identifying thematic keyphrases. Fig. 8 display two clusters with centres.

3.6. Experiment and Analysis

The text test of this experiment was conducted with two data sets, SemEval 2017 (Augenstein et al., 2017). And Inspec (Hulth, 2003). Many comparisons were made with other benchmark models. The Inspec dataset consists of 2000 English abstracts of journal articles. 500 documents, as used by previous researchers, were used for an appropriate comparison of results. The Semeval2017 dataset contains 493 paragraphs from sciencedirect journal articles in the

fields of computer science, materials science, and physics. Each paragraph contains an average of 17 keyphrases and 176 tokens. The keyphrases were annotated by students of computer science, materials science, or physics under the supervision of keyphrase annotation experts as shown in Table 1.

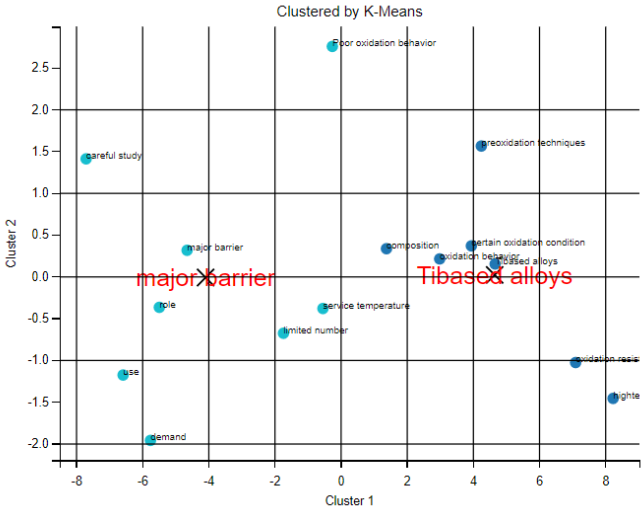

Fig8: Topic keyphrase as a center using K-means

Table 1: A Summary of Datasets

Dataset	Document Number	Average Sentence Number	Average Word Number
Inspec	500	6	134
SemEval2017	493	7	168

In this section, we observe the results of the proposed work according to the evaluation criteria for each of the cases that will be addressed. For both databases, each document has golden keyphrases. The precision, recall and F measure (Liu, Lin and Wang, 2021) precision is the proportion of correctly extracted keyphrases to the total number of extracted keyphrases. It measures the accuracy of the extraction process by indicating how many of the extracted phrases are actually relevant. Recall is the proportion of correctly extracted keyphrases to the total number of relevant keyphrases present in the original text. It measures the completeness of the extraction process by showing how many of the relevant phrases were successfully identified. The F-measure is the harmonic mean of precision and recall. It provides a single metric that balances the accuracy and completeness of the extracted keyphrases, offering a comprehensive assessment of the extraction process. were calculated to evaluate the effectiveness of this work according to the results extracted through experiments as follows:

$$P = \frac{TP}{TP + FP} \quad (5)$$

$$R = \frac{TP}{TP + FN} \quad (6)$$

$$F1 = 2 \times \frac{P \times R}{P + R} \quad (7)$$

Where TP and FP indicate the number of keyphrase that belong to and not belong to the extracted words, respectively, and FN and TN indicate the number of keyphrase that belong to and not belong to the words that have not been extracted, respectively and Table 2 show result precision, recall and F-measure for semeval2017 and inspect .

Table 2 result precision, recall and F-measure for semeval2017

Method	P@5	R@5	F1@5	P@10	R@10	F1@10	P@15	R@15	F1@15
PositionRank	39.86	21.04	28.12	45.64	23.43	32.87	47.92	24.15	33.32
MultipartiteRank	36.78	18.98	25.96	40.95	21.11	29.57	43.28	22.15	30.85
AGRank	49.06	26.89	34.59	56.54	30.41	40.70	58.76	32.23	41.15
PhraseRank	51.23	28.26	36.46	58.23	32.24	43.17	63.12	35.18	54.82

Table 3 result precision, recall and F-measure for inspect

Method	P@5	R@5	F1@5	P@10	R@10	F1@10	P@15	R@15	F1@15
PositionRank	33.28	13.39	18.23	36.78	19.26	26.30	41.61	20.89	30.55
MultipartiteRank	32.10	12.76	17.39	34.76	17.36	23.73	38.21	18.94	26.87
AGRank	44.53	17.76	24.13	48.86	24.58	33.46	50.98	27.51	37.21
PhraseRank	47.13	20.11	27.81	51.72	27.12	26.82	53.55	29.10	42.67

This study evaluates the performance of four keyword extraction methods—PositionRank, MultipartiteRank**,** AGRank**,** and PhraseRank—using precision, recall, and F1-measure metrics across two datasets: Semeval2017 and Inspect. Our results show that PhraseRank consistently achieves the highest values for precision, recall, and F1-measure at various cut-off points (P@5, P@10, P@15, and R@5, R@10, R@15) on both datasets, followed closely by AGRank. The superior performance of these methods can be attributed to their combination of statistical information with contextual approaches, setting them apart from previous studies that may have relied solely on statistical methods. The results underscore the robustness and effectiveness of these approaches, highlighting their utility in a wide range of academic and practical contexts, including text summarization, information retrieval, and automated indexing. This study also provides insights into the comparative advantages of different keyword extraction methods, suggesting that more sophisticated techniques can significantly enhance keyword accuracy and relevance. In addition, Fig. 9 shows a) an example of real input text from a summary and b) key phrases corresponding to the ground truth extracted from the document summary. This figure shows how the different methods compare to the original output, providing a clear visual representation of the effectiveness and accuracy of the keyword extraction methods discussed.

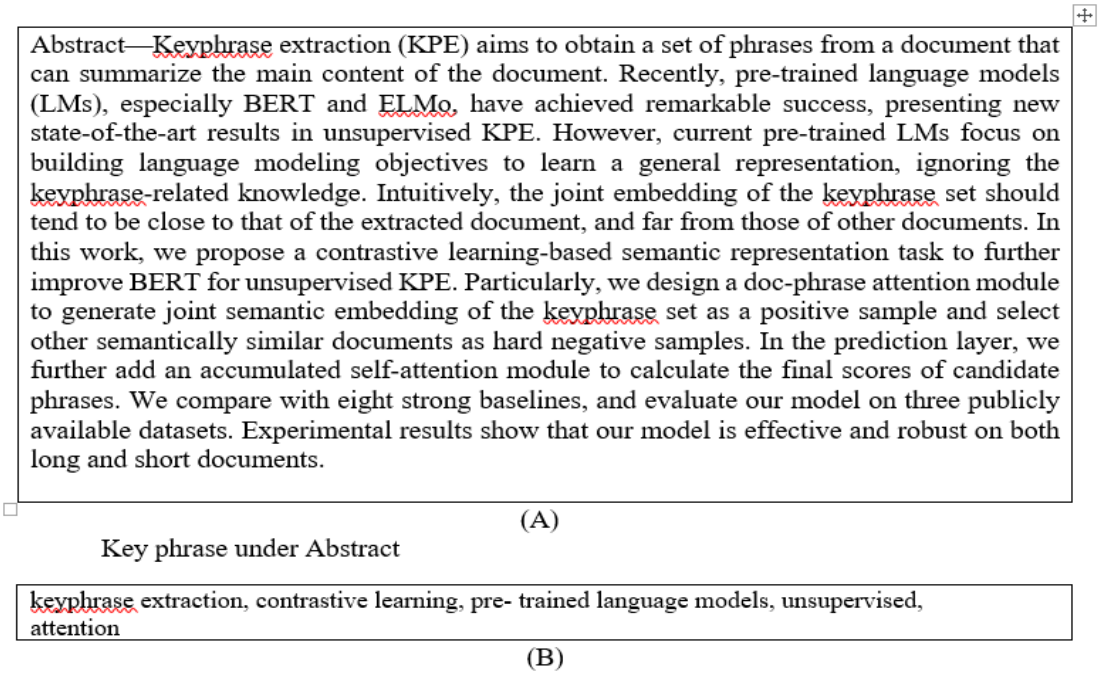

Fig9. A) Example real input text (abstract). B) Truth Key phrase under Abstract the document

The performance is evaluated using BLEU metrics (Papineni *et al.*, 2002). It is another evaluation measure and more effective in measuring the *n*-gram overlap than ROUGE. BLEU measures both the exact match and the approximate match between the extracted keyphrases and the human-annotated ones to evaluate the performance of proposed method. In machine translation, the modified *n*-gram precision between a *candidate* sentence and a *reference* sentence is estimated as in the equation (8):

$$\text{CountClip} = \min(\text{Count}(n-\text{gram}), \text{maxCount}_{r\in R}(n-\text{gram}' \in r)) \quad (8)$$

Count(n-gram): The count of a specific n-gram in the candidate keyphrase.

R: The set of reference keyphrases.

Count(n-gram' ∈ r): The count of the specific n-gram in the reference keyphrase r.

max_{r ∈ R} Count(n-gram' ∈ r): The maximum count of the specific n-gram found in any reference keyphrase.

The above function takes the minimum between the count of a specific n-gram in the candidate keyphrases and the maximum count of the same n-gram found in any reference keyphrase. It's used to prevent overcounting the same n-gram when it appears multiple times in the reference keyphrases. In the context of this evaluation, the modified *n*-gram precision was applied to measure the approximate match between a *candidate* keyphrase and a *reference* keyphrase illustrative in the equation (9).

$$\text{pn}= \frac{\sum c\in C\sum \text{n}-\text{gram}\in c\text{CountClip(n}-\text{gram)}}{\sum c\prime\in C\prime\sum \text{n}-\text{gram}\prime\in c\prime\text{Count(n}-\text{gram}\prime)} \quad (9)$$

C: The set of candidate keyphrases.

CountClip(n-gram): The clipped count of the specific n-gram in the candidate keyphrase.

C': The set of reference keyphrases.

Count(n-gram'): The count of the specific n-gram in the reference keyphrase c′.

Table 4 The performance of four approaches across two datasets using the average P@15

Approach	Semeval2017
Multipartite Graph	0.332
PositionRank	0.46
ContextualRank	0.533
Hierarchical topic modeling	0.328
PhraseRank	0.66

Table 5 The performance of four approaches across two datasets using the average MRR@15

Approach	inspect
Multipartite Graph	0.09
PositionRank	0.136
ContextualRank	0.136
Hierarchical topic modeling	0.1
PhraseRank	0.24

4. RESULT AND DISCUSSION

In this comprehensive analysis, we explore the performance metrics Precision (P), Recall (R),of various keyphrasextraction methods as outlined in Tables 2 and 3, which cover two distinct datasets: semeval2017 and Inspect. The methods under scrutiny include positionrank, multipartiterank ,sifrank,agrank and an enhanced variant known as "phraserank." These methods were evaluated at three different cutoff points: the top 5, 10, and 15 keywords, providing a thorough evaluation of their capability in keyword extraction.

The results from Table 4 for the semeval2017 dataset demonstrate that the improved version of textrank, referred to as "phraserank" significantly outperforms other methods with a BLEU score of 0.66, indicating a high degree of precision in matching the extracted keyphrase with the reference set. This superior performance suggests that "phraserank" likely incorporates advanced algorithmic enhancements that improve semantic understanding and relevance assessment. Other methods such as positionrank and contextualrank also show commendable performance with scores of 0.46 and 0.533, respectively, indicating their effectiveness in capturing contextually significant terms, albeit not as precisely as "phraserank." In contrast, Multipartite Graph and Hierarchical Topic Modeling, with scores of 0.332 and 0.328

respectively, exhibit moderate alignment with the reference keyphrase, highlighting potential areas for refinement in these approaches.

Further insights are provided by the analysis of Table 5, which reports the performance on the Inspect dataset using MRR. Here again, "phraserank" leads with an MRR score of 0.24, reflecting its capability not only to accurately identify relevant keyphrase but also to rank them higher than other methods. Positionrank and contextualrank, both scoring 0.136, show moderate effectiveness in ranking relevant keyphrase appropriately. Meanwhile, Hierarchical Topic Modeling and Multipartite Graph score 0.1 and 0.09, respectively, suggesting that while they can identify relevant terms, these terms often appear lower in the ranked list, thus indicating a delay in retrieving the most pertinent keyphrase.

5. CONCLUSION

In this study, an efficient method for keyphrase extraction with subsequent filtering was presented to obtain candidate keyphrase topics. The performance is evaluated using the Inspec and semeval2017 datasets. The proposed phraserank performs an efficient equation that utilizes both statistical and contextual information to compute the edge strength between any two keyphrases and create a graph, which is then used in the iterative graph-based ranking to identify salient keyphrases. PhraseRank's superior performance compared to other methods stems from its integration of multiple metrics position, cosine similarity, and distance which provide a comprehensive evaluation of keyphrases. By considering their significance, contextual similarity, and proximity within the text, PhraseRank constructs a graph where nodes represent keyphrases and edges signify their relationships, weighted by the combined measure of physical proximity and semantic similarity. This graph-based ranking algorithm, inspired by Google's PageRank, ranks the keyphrases to highlight the most important ones based on their contextual and positional relevance within the text. This balanced approach results in high precision, indicating accurate identification of relevant keyphrases, and high recall, ensuring a broad capture of pertinent keyphrases.

In another context, good candidates for thematic keywords expressing the basic concepts of the document were found through a series of appropriate strategies. The method proved to be clearly superior and showed the highest efficiency in precision, recall. Phraserank method not only achieved exceptional precision an indication of its ability to accurately identify relevant keyphrases but also showed considerable recall, suggesting that it can capture a comprehensive range of relevant keyphrases. Future work could focus on investigating whether "phraserank" works particularly well with various lengths of documents. In a complementary line of research,

we can also examine and evaluate the results of the strategies used to extract the topic (thematic) key phrases and determine whether they have achieved the desired goal.

6. REFERENCE

Abasi, A., Khader, A.T. and Al-Betar, M.A. (2022) 'AN IMPROVED MULTI-VERSE OPTIMIZER FOR TEXT DOCUMENTS CLUSTERING', Kufa Journal of Engineering, 13(2), pp. 28–42. Available at: https://doi.org/10.30572/2018/KJE/130203.

Augenstein, I. et al. (2017) 'SemEval 2017 Task 10: ScienceIE - Extracting Keyphrases and Relations from Scientific Publications'. Available at: http://arxiv.org/abs/1704.02853.

Boudin, F. (2018) 'Unsupervised Keyphrase Extraction with Multipartite Graphs'. Available at: http://arxiv.org/abs/1803.08721.

Chen, X. et al. (2012) CIKM'12 : the proceedings of the 21st ACM International Conference on Information and Knowledge Management : October 29 - November 2, 2012, Maui, Hawaii, USA.

Devika, R. et al. (2021) 'A Deep Learning Model Based on BERT and Sentence Transformer for Semantic Keyphrase Extraction on Big Social Data', IEEE Access, 9, pp. 165252–165261. Available at: https://doi.org/10.1109/ACCESS.2021.3133651.

Devlin, J. et al. (2018) BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Available at: https://github.com/tensorflow/tensor2tensor.

Ding, H. and Luo, X. (2022) AGRank: Augmented Graph-based Unsupervised Keyphrase Extraction. Long Papers. Available at: https://github.com/hd10-iupui/AGRank.

Du, H. et al. (2023) 'Contextual topic discovery using unsupervised keyphrase extraction and hierarchical semantic graph model', Journal of Big Data, 10(1). Available at: https://doi.org/10.1186/s40537-023-00833-1.

Florescu, C. and Caragea, C. (2017) 'PositionRank: An unsupervised approach to keyphrase extraction from scholarly documents', in ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers). Association for Computational Linguistics (ACL), pp. 1105–1115. Available at: https://doi.org/10.18653/v1/P17-1102.

Han, J., Kim, T. and Choi, J. (2008) 'Web Document Clustering by Using Automatic Keyphrase Extraction', in. Institute of Electrical and Electronics Engineers (IEEE), pp. 56–59. Available at: https://doi.org/10.1109/wi-iatw.2007.46.

Huang, A. (2018) Similarity Measures for Text Document Clustering.

Hulth, A. (2003) Improved Automatic Keyword Extraction Given More Linguistic Knowledge.

Kong, A. et al. (2023) PromptRank: Unsupervised Keyphrase Extraction Using Prompt. Long Papers.

Liao, S. et al. (2023) 'TopicLPRank: a keyphrase extraction method based on improved TopicRank', Journal of Supercomputing, 79(8), pp. 9073–9092. Available at: https://doi.org/10.1007/s11227-022-05022-0.

Liu, R., Lin, Z. and Wang, W. (2021) 'Addressing Extraction and Generation Separately: Keyphrase Prediction with Pre-Trained Language Models', IEEE/ACM Transactions on Audio Speech and Language Processing, 29, pp. 3180–3191. Available at: https://doi.org/10.1109/TASLP.2021.3120587.

Mihalcea, R. and Tarau, P. (2004) TextRank: Bringing Order into Texts.

Papagiannopoulou, E. and Tsoumakas, G. (2019) 'A Review of Keyphrase Extraction'. Available at: http://arxiv.org/abs/1905.05044.

Papineni, K. et al. (2002) BLEU: a Method for Automatic Evaluation of Machine Translation.

Patel, K. and Caragea, C. (2021) Exploiting Position and Contextual Word Embeddings for Keyphrase Extraction from Scientific Papers.

Sarwar, T. Bin, Noor, N.M. and Miah, M.S.U. (2022) 'Evaluating keyphrase extraction algorithms for finding similar news articles using lexical similarity calculation and semantic relatedness measurement by word embedding', PeerJ Computer Science, 8. Available at: https://doi.org/10.7717/peerj-cs.1024.

Song, M., Feng, Y. and Jing, L. (2022) Utilizing BERT Intermediate Layers for Unsupervised Keyphrase Extraction.

Song, M., Feng, Y. and Jing, L. (2023) A Survey on Recent Advances in Keyphrase Extraction from Pre-trained Language Models. Available at: https://huggingface.co/bert-base-cased.

Tsvetkov, A. and Kipnis, A. (2023) EntropyRank: Unsupervised Keyphrase Extraction via Side-Information Optimization for Language Model-based Text Compression.

Wu, Y.-F.B. et al. (2005) Domain-specific Keyphrase Extraction.