Bulletin of Electrical Engineering and Informatics
Vol. 14, No. 2, April 2025, pp. 1518~1525
ISSN: 2302-9285, DOI: 10.11591/eei.v14i2.8293

Fuzzy logic: a novel approach to compound noun extraction

Latifa Rassam, Ahmed Zellou
Software Project Management Research Team, Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Mohammed V University in Rabat, Rabat, Morocco

Article Info

Article history:

Received Mar 12, 2024
Revised Nov 9, 2024
Accepted Nov 19, 2024

Keywords:

Compound noun extraction
Fuzzy indexing
Fuzzy logic
Natural language processing
N-gram graph

ABSTRACT

Compound noun extraction from textual documents presents a unique challenge due to the inherent complexity and variability in linguistic structures. Traditional approaches often struggle to accurately capture the nuanced semantics of compound nouns, primarily due to their rigid reliance on exact matches. In response, this research underscores the pivotal role of fuzzy logic in addressing the challenges associated with ambiguity and imprecision within compound noun extraction. Leveraging the inherent flexibility of fuzzy logic, we propose a novel approach that surpasses the limitations of traditional methods. Our method embraces the adaptability of fuzzy logic, providing a powerful and context-aware solution for compound noun extraction. Empirical evaluation demonstrates superior performance, with a macro precision of 0.572, recall of 0.607, and F-measure of 0.589, compared to traditional approaches. By incorporating fuzzy logic, our approach excels in handling variations and uncertainties present in natural language, ultimately offering a more accurate and nuanced representation of compound nouns within textual documents. This research not only advances the field of compound noun extraction but also underscores the efficacy of fuzzy logic in overcoming challenges associated with linguistic intricacies in information extraction tasks.

Corresponding Author:

Latifa Rassam
Software Project Management Research Team
Ecole Nationale Supérieure d'Informatique et d'Analyse des Systèmes, Mohammed V University in Rabat
Avenue Mohammed Ben Abdellah Regragui, Madinat Al Irfane, BP 713, Agdal Rabat, Morocco
Email: rassamlatifa@gmail.com

1. INTRODUCTION

In textual document indexing, the complexities arising from compound nouns pose significant challenges to traditional approaches. Compound nouns, composed of two or more individual words, often carry nuanced meanings that can be difficult to capture accurately using strict Boolean logic [1]. The incorporation of fuzzy logic, coupled with the innovative method of compound noun indexing and n-gram graph analysis, offers a promising solution to enhance the precision and flexibility of document retrieval systems [2].

Compound nouns, formed by the combination of individual terms, frequently exhibit multifaceted meanings. The inherent ambiguity in these expressions necessitates a more nuanced approach to indexing, one that can dynamically adapt to the diverse semantic interpretations [3]. Fuzzy logic, renowned for its ability to model uncertainty and imprecision, becomes a natural candidate for addressing the challenges associated with the representation and retrieval of compound nouns.

Fuzzy logic operates on the principle of gradual truth, allowing terms to belong to sets with varying degrees of membership. This adaptive precision aligns seamlessly with the flexible nature of compound

nouns, enabling a more accurate representation of their semantic richness [4]. By introducing fuzzy membership functions to individual terms within compound nouns, the indexing process becomes inherently more elastic and responsive to the inherent variability in language.

Incorporating the n-gram graph method adds a layer of sophistication to the indexing process [5]. N-grams, representing contiguous sequences of n terms, provide a contextual understanding of word combinations within documents. Constructing a graph based on these n-grams allows for the exploration of semantic relationships between compound nouns, contributing to a more comprehensive and context-aware document representation [6]. Hence, this work primarily encompasses four key contributions:

- introducing a novel optimal N-gram graph-based method for extracting compound words. Seeing that the incorporation of n-gram graphs enables the indexing system to dynamically adjust to the context in which compound nouns are used, capturing subtle shifts in meaning and context.
- presenting an enhanced algorithm for textual indexing.
- developing innovative fuzzy logic-based methods to assess the relevance degree of generated keywords in a corpus of textual documents.
- implementing and assessing the proposed technique using real-world domain datasets.

Our research journal starts with the related work section, providing context by reviewing existing methods and their shortcomings. The method section follows, where we introduce the foundational indexing technique and elaborate on the novel application of fuzzy logic to address linguistic variability. The evaluation section then examines the approach's performance and discusses the proposed method's advantages. Finally, the conclusion and future work section reflects on the research contributions and proposes avenues for future exploration.

2. RELATED WORK

2.1. Unsupervised approaches

Earlier research [7]-[9] has introduced creative unsupervised techniques for keyword extraction by integrating n-grams with both word and document embeddings [10]. These methods generate document vectors by combining word embeddings with their inverse document frequency (idf) values [11], while also using higher-order word n-grams to enrich unigram embeddings, thereby producing more robust document embeddings [12]. Experiments conducted on various datasets have shown that merging higher-order word n-grams with enhanced Glove embeddings [10] and document embeddings can effectively improve keyword extraction. Particularly, the use of bi-gram enhanced embeddings [13] has shown considerable advancements compared to traditional methods.

2.2. Semi-supervised approaches

Ye and Wang [14] proposed a probabilistic model specifically designed for semi-supervised learning contexts. This model integrates graph-based document data within a Bayesian framework, utilizing an informed prior to boost the model's ability for formal statistical reasoning and better capture the nuances in documents. Their approach applies a fuzzy inference system to assess sentence scores and employs bidirectional gated recurrent units to remove redundant or similar sentences. Moreover, their technique generates abstractive summaries based on the selected sentences [15].

2.3. Supervised approaches

In their research, Terrada *et al.* [16] present a post-processing strategy aimed at enhancing automatic keyword extraction (AKE) methods by integrating semantic understanding through part-of-speech (PoS) tagging. Terrada *et al.* [17] assess their supervised approach by combining PoS tagging word types, domain-specific terms from specialized thesauri, and named entities [16]-[18] as semantic resources. Their findings show that incorporating these semantic elements significantly improves AKE performance. Meanwhile, Xia *et al.* [19] and Ye *et al.* [20] concentrate on refining the management of extensive multi-document datasets.

These indexing techniques face several key challenges. First, these methods often generate a vast array of potential indexes, which include both accurate and inaccurate entries. This variability requires manual review to ensure the correctness of the indexes produced. Second, many of these techniques are optimized for structured data and do not fully leverage the advantages of indexing large volumes of unstructured text. Key organizational elements within documents, such as subtitles, titles, keywords, and chapter headings, which are rich with relevant information, are frequently overlooked by these methods. Finally, current indexing techniques and performance metrics often fall short in precisely aligning generated indexes with the annotated keywords in textual documents, leading to discrepancies in their accuracy. In the following section, we will present our fuzzy logic-based method, which aims to improve index generation performance. This approach accommodates the uncertainty and imprecision inherent in linguistic expressions, allowing for a more flexible and detailed handling of document content.

3. METHOD

3.1. The fuzzy indexing n-gram graph process

The value of a document is closely linked to the meaning and significance of each sentence and its associated terms, which together contribute to the overall knowledge presented in the document [21]. Therefore, assessing the document's value involves evaluating the importance of each term it contains. The schema shown in Figure 1 effectively illustrates the proposed indexing approach, covering the entire process from the initial segmentation phase to the subsequent phase of matching accuracy.

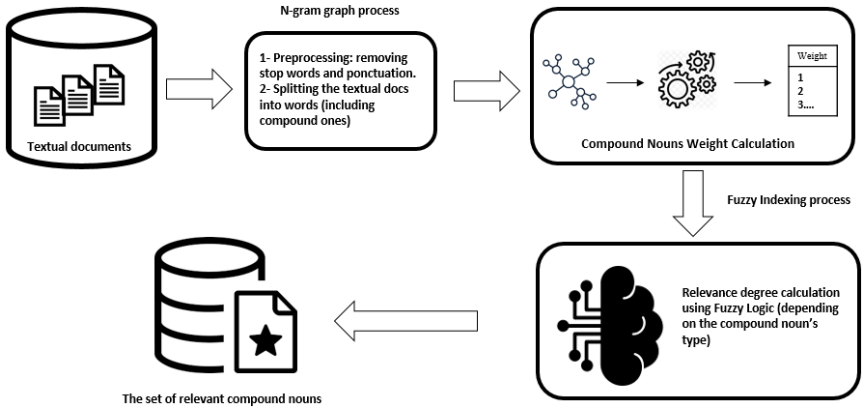

Figure 1. Workflow of the proposed fuzzy indexing method

An n-gram graph is a straightforward graph that replaces a segmented textual document $\Omega=\{V_\Omega, E_\Omega,$ L, W$\}$, where: V_Ω is the vertex's set; E_Ω is the edges set; L is the function that assigns labels to the set of vertex and edges; and W is a function assigning a weight to every edge. In the graph, the vertices (denoted as v) are part of the set V_Ω and represent the n-grams present within the document. The edges (denoted as e) are part of the set E_Ω and indicate the proximity between the n-gram vertices. These edges illustrate the connections or closeness between the n-grams in the graph. Figure 2 demonstrates the initial phase of the indexing process, which transitions from a structured textual document to an n-gram graph. This phase begins with the transformation step, specifically tokenization using the Levenshtein Algorithm 1.

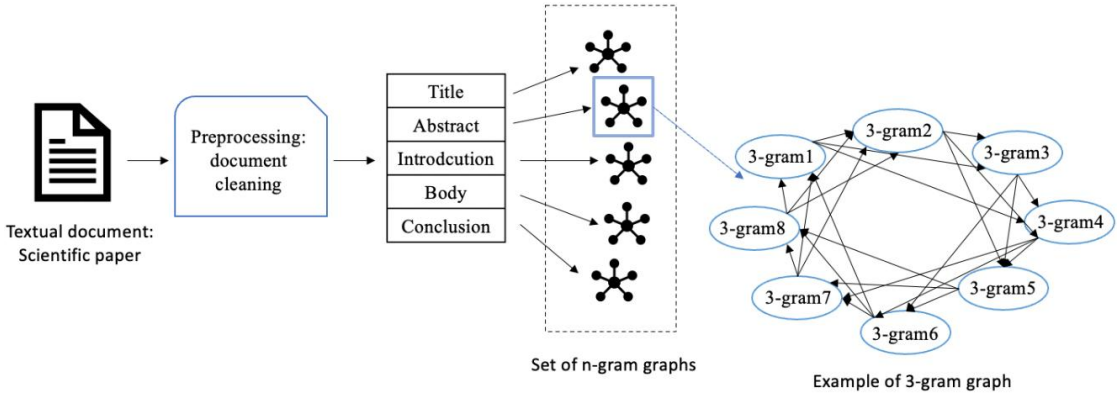

Figure 2. The comprehensive procedure for word extraction using the n-gram graph method

We have enhanced the efficiency of the proposed algorithms to address limitations commonly encountered in other algorithms, such as the Jaccard similarity algorithm [22] and cosine similarity algorithm [22], by implementing the following optimizations:

- Utilizing Porter stemming function [23]: this function reduces words to their root forms, aiding in consolidating similar words and diminishing vocabulary size.
- Incorporating stop words from the NLTK library [24]: a predefined set of stop words is employed to filter out n-grams containing commonly occurring and less informative words.

– Leveraging NLTK library's n-grams function: this function efficiently generates n-grams from the document, streamlining the process.

Last but not least, using defaultdict class from the collection module: this class is utilized to store the n-gram graph, offering a convenient method to manage occurrence counts. The complexity of the proposed algorithms is directly linked to the size of the textual document |td|, scaling linearly [25].

Algorithm 1. Tokenization of textual document into n-gram graph

```
Start
  Inputs:
       ptd: A preprocessed textual document
       n: Number of n-grams
  Outputs:
       SΩ_m : Set of n-gram words
       // Initialize SΩ_m as an empty dictionary with integer default values
       SΩ_m← defaultdict(int)
       // Initialize stemmer using PorterStemmerc
       stemmer ← PorterStemmer()
       // Define stop_words as the set of English stopwords
       stop_words ← set(stopwords.words('English))
       // For each ngram in the set of n-grams extracted from ptd with size n
       For ngram in ngrams(ptd, n):
       // Apply stemming to each token in the n-gram, resulting in stemmed_ngram
          stemmed_ngram ←  tuple(stemmer.stem(token))
       // Check if any token in stemmed_ngram is a stop word
          If any(token in stop_words for token in stemmed_ngram):
       // If true, convert stemmed_ngram to a string ngram_str
            ngram_str ←  join((stemmed_ngram), ' ')
       // Increment the count of ngram_str in SΩ_m
            SΩ_m[ngram_str] + 1
          End if
       End for
       Return SΩ_m
  End
```

3.2. Measuring n-gram word frequency with our fuzzy logic method

In fuzzy logic, a document converted into an n-gram graph is represented by a set of fuzzy terms, denoted as x. The relationship between the document and the expression can be defined as (1):

$$\Upsilon = TD \times \text{Ṭ} \rightarrow [0,1]\,,\ \mathcal{B}(td_x) = \Upsilon(td,t) \quad (1)$$

where Ṭ is the set of all terms, and TD is the set of documents within the corpus.

In the context of a large collection of textual documents (td_i), let's examine a specific document, td1, represented using an n-gram graph-based approach. Through this n-gram graph indexing method, we have identified three keywords x_i, x_j, x_k. To measure the membership degree of a particular expression x_i, we use the core mathematical function $Calculatoccurence$(t,td). This function counts how many times the expression x appears within the document td. By applying this function, we can assess the relevance and presence of the expression td_1 [26]:

$$Degree(x_i, td) = \frac{\sum_{i=1}^{n} Calculatoccurence(x_i, td)}{m} \quad (2)$$

The term 'm' represents the frequency of the most repeated term within the textual document td while 'n' denotes the total number of grams used in the document's indexing process. 'n' acts as a normalization factor to assess the relative importance of other terms compared to the most frequently occurring term [27]. To determine the relevance degree of a term within a document, we use (1) and (2), which measure the relationship of an expression within the specific document. We identify two primary cases:

The occurrence degree $\varphi x_i(td_i)$ is computed using the function outlined in (2). The proposed fuzzy functions can be applied to various terms across different documents. Evaluations performed on dataset extracts revealed a significant improvement in efficiency, underscoring the effectiveness of these fuzzy functions. This implementation not only illustrates the practicality of the approach but also highlights its potential to enhance overall performance in managing textual data.

Therefore, the general representation of the proposed fuzzy functions is as (3) and (4):

– For different terms in the same textual document:

$$\varphi x_{i,j}(td_1) = \begin{Bmatrix} 0 \\ \frac{Maximum\left(\left(\varphi x_i(td_1)\right),\left(\varphi x_j(td_1)\right)\right)\times n}{n-1} \\ 1 \end{Bmatrix} \quad (3)$$

– For multiple different terms in many distinct textual documents:

$$\psi x_{i,j}(td_1, td_2) = \begin{Bmatrix} 0 \\ \frac{Minimum\left(\left(\varphi x_i(td_1)\right),\ldots,\left(\varphi x_w(td_w)\right)\right)\times n}{n-1} \\ 1 \end{Bmatrix} \quad (4)$$

A fuzzy value of 1 means that the keywords appear in every document within the corpus, showing that they are universally present throughout the entire collection. Conversely, a fuzzy value between 0 and 1 suggests that the keyword or index is found multiple times across different documents but is not present in every document, indicating its frequent occurrence. A fuzzy value of 0, on the other hand, indicates that the keyword does not appear in any of the documents within the corpus, signifying its total absence from the collection.

4. EVALUATION

4.1. Datasets

This section provides an overview of the datasets employed to evaluate the proposed fuzzy logic-based method. Various datasets were used for training and testing AKE approaches, with a summary provided in Table 1. It's worth noting that deep learning (DL) techniques often require extensive datasets for effective training. Prior to 2017, the largest available dataset contained just 2,304 scientific articles [28], which was insufficient for training recurrent neural networks (RNNs). Furthermore, most methods under evaluation rely on five distinct datasets for performance assessment, as outlined in Table 1. Interestingly, the recently introduced KPTimes dataset [29], which contains a large volume of training documents, was not included in the AKE evaluation.

Table 1. The datasets used for comparing the various aaproaches

Dataset	Documents	Documents type	Language	Annotation	Test documents	Usage rate (%)
NUS	211	Full paper	English	Reader	211	50
Krapivin	2,304	Full paper	English	Author	404	42
Semeval	244	Full paper	English	Both	-	58
Inspec	300	Full paper	English	Reader	230	49
Thesis100	420	Full paper	English	Both	310	53

4.2. Evaluation metrics

Alongside the fuzzy logic-based degree of relevance calculation method, we employ the following metrics [30] to assess the accuracy of the indexes produced by the new approach:

$$Precision = \frac{Accurate\ index}{Accurate\ index + Inaccurate\ index} \quad (5)$$

$$Recall = \frac{Accurate\ index}{Missed\ index + Accurate\ index} \quad (6)$$

$$Overall = Recall \times \left(2 - \frac{1}{Macro\ Precision}\right) \quad (7)$$

$$F - Measure = 2 \times \left(\frac{Macro\ Precision \times Recall}{Macro\ Precision + Recall}\right) \quad (8)$$

The metric (5) is used to measure the accuracy of the generated indexes, while the metric (6) gauges their effectiveness by calculating the proportion of relevant terms accurately identified and included. The metric (7) confirms both the accuracy and effectiveness of the new indexing method, and the metric (8) is designed to evaluate and validate the performance of the generated indexes.

These measurement outcomes are relative and can vary based on factors such as the number of extracted words and the document's content and length. Additionally, methods that do not include phrases

outside of the document's content may show fewer results when evaluated with these metrics. Therefore, it is essential to explore alternative performance evaluation techniques to address these constraints.

4.3. Results and discussions

Table 2 highlights that the membership degree indicator rises with word graph-based indexing at n=2, while it declines at n=4 and stabilizes around n=3. This suggests that, within this framework, word graph-based indexing with n=2 yields the most efficient indexing of textual documents. The data in the first paragraph will be visualized with two graphs, each containing 14 vertices corresponding to n values of one, two, and three, as outlined in Tables 2 and 3. To thoroughly evaluate the quality of the indexing process and the indexes generated, it is essential to consider more than just the count of n-grams, relevance, and membership degree. Additional metrics, including macro precision, recall, and overall effectiveness, are also crucial and are influenced significantly by the choice of n. Tables 2 and 3 provides further insights into how these elements interact.

Table 2. Precision metrics for various techniques and datasets

Methods	Dataset				
	NUS	Krapivin	Semeval	Inspec	Thesis100
The proposed approach	**0.415**	**0.536**	**0.572**	**0.294**	0.24
Enhanced words	0.316	0.488	0.475	0.251	0.28
PoS-tagging	0.347	0.398	0.349	0.222	**0.291**
Fuzzy Bi-GRU	0.237	0.436	0.465	0.129	0.168

Table 3. Recall metrics for different techniques and datasets

Methods	Dataset				
	NUS	Krapivin	Semeval	Inspec	Thesis100
The proposed approach	**0.343**	**0.544**	**0.607**	**0.287**	0.316
Enhanced words	0.285	0.5	0.45	0.226	0.347
PoS-tagging	0.313	0.405	0.359	0.2	**0.354**
Fuzzy Bi-GRU	0.311	0.342	0.345	0.1	0.079

Analyzing these values uncovers the complex relationships between various factors, enabling an assessment of the overall quality of the indexing process and the indexes generated. The tables presented provide the foundation for the graph-based illustrations. The graphs in Figure 3 display the variation in F-measure values throughout the analysis based on the N value, compared to metrics from three extraction systems discussed in the related work section. For these three alternative approaches, we applied our developed fuzzy function to calculate metrics based on their reported empirical results.

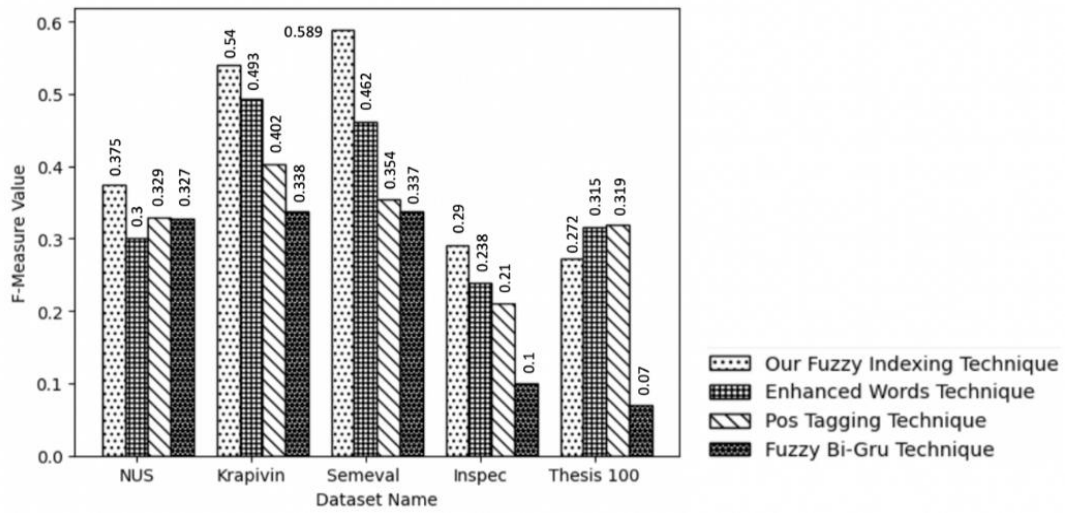

Figure 3. Comparison of the F-measure scores of the proposed fuzzy indexing technique with other indexing methods across different datasets

Ideally, all metrics would reach a maximum value of 1, signifying perfect performance. This would mean that macro precision, recall, and F-measure each equal 1, reflecting optimal alignment across all metrics. To validate our proposed fuzzy logic-based method, we tested it on a real-world domain dataset and compared the generated indexes with those produced by three existing extraction systems: enhanced word and document embedding [11], PoS-Tagging with advanced semantic understanding [17], and the fuzzy Bi-GRU hybrid model for extractive and abstractive summarization of extensive multi-documents [15]. Our evaluation prioritized three key metrics macro precision, recall, and F-measure to assess the consistency of classifications and properties. Our approach showed high matching accuracy with the real-world dataset, as evidenced by average metric values (for N=2) of macro precision=0.572, recall=0.607, and F-measure=0.589. In contrast, established indexing systems demonstrated lower accuracy, with average metric ranges of 0.129 to 0.488 for macro precision, 0.079 to 0.405 for recall, and 0.07 to 0.493 for F-measure.

5. CONCLUSION

This research aimed to identify the most efficient approach for indexing textual documents and extracting relevant compound terms, alongside establishing a strong evaluation framework for corpus indexing through fuzzy logic. We highlighted the importance of fuzzy logic in indexing and identifying text data, along with employing a graph-based n-gram word approach and a novel indexing algorithm. Through practical trials and real-world case studies, we analyzed how different n values impact the selection of indexes within the graph-based n-gram model. The intricacies of this indexing method, which hinge on the n value, were fully implemented and detailed. Additionally, an in-depth assessment was performed to pinpoint the most effective technique by examining factors like precision, relevance, membership degree, and the quantity of N-grams produced. This study introduced a fuzzy logic-based evaluation function that integrates multiple factors such as n and degree of relationship to assess compound terms and address issues and constraints in conventional methods. It also lays the groundwork for extending these techniques to multilingual corpora, providing a promising avenue for future research.

ACKNOWLEDGMENTS

The authors did not receive any specific support or funding for this research. The authors would like to thank the reviewers and editorial team for their valuable comments and insights during the review process.

REFERENCES

[1] S. R. El-Beltagy and A. Rafea, "KP-Miner: A keyphrase extraction system for English and Arabic documents," *Information Systems*, vol. 34, no. 1, pp. 132–144, 2009, doi: 10.1016/j.is.2008.05.002.

[2] H. Li, J. Zhu, J. Zhang, C. Zong, and X. He, "Keywords-guided abstractive sentence summarization," *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 5, pp. 8196–8203, Apr. 2020, doi: 10.1609/aaai.v34i05.6333.

[3] D. Buscaldi, G. Felhi, D. Ghoul, J. Le Roux, G. Lejeune, and X. Zhang, "Calcul de similarité entre phrases : quelles mesures et quels descripteurs ?," in *Actes de la 6e conférence conjointe Journées d'Études sur la Parole (JEP, 33e édition), Traitement Automatique des Langues Naturelles (TALN, 27e édition), Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCIT)*, 2020, pp. 14–25.

[4] R. K. Mishra, G. Y. S. Reddy, and H. Pathak, "The Understanding of Deep Learning: A Comprehensive Review," *Mathematical Problems in Engineering*, pp. 1–5, 2021, doi: 10.1155/2021/5548884.

[5] P. Yang, Y. Ge, Y. Yao, and Y. Yang, "GCN-based document representation for keyphrase generation enhanced by maximizing mutual information," *Knowledge-Based Systems*, vol. 243, p. 108488, 2022, doi: 10.1016/j.knosys.2022.108488.

[6] N. Nikzad-Khasmakhi *et al.*, "Phraseformer: Multimodal Key-phrase Extraction using Transformer and Graph Embedding," *arXiv*, pp. 1-5, 2021, doi: 10.48550/arXiv.2106.04939.

[7] O. Bouidghaghen, M. Boughanem, H. Prade, and I. Mallak, "A fuzzy logic approach to topic extraction in texts," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 17, no. supp01, p. 81-112, 2009, doi: 10.1142/S0218488509006042.

[8] E. Papagiannopoulou and G. Tsoumakas, "A review of keyphrase extraction," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, no. 2, 2020, doi: 10.1002/widm.1339.

[9] X. Li and M. Daoutis, "Unsupervised key-phrase extraction and clustering for classification scheme in scientific publications," *arXiv*, vol. 2831, pp. 1–8, 2021, doi: 10.48550/arXiv.2101.09990.

[10] S. Siddiqi and A. Sharan, "Keyword and Keyphrase Extraction Techniques: A Literature Review," *International Journal of Computer Applications*, vol. 109, no. 2, pp. 18–23, 2015, doi: 10.5120/19161-0607.

[11] L. Chi and L. Hu, "ISKE: An unsupervised automatic keyphrase extraction approach using the iterated sentences based on graph method," *Knowledge-Based Systems*, vol. 223, pp. 1–12, Jul. 2021, doi: 10.1016/j.knosys.2021.107014.

[12] F. Boudin, "Unsupervised keyphrase extraction with multipartite graphs," in *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 2018, pp. 667–672, doi: 10.18653/v1/n18-2105.

[13] Y. Sun, H. Qiu, Y. Zheng, Z. Wang, and C. Zhang, "SIFRank: A New Baseline for Unsupervised Keyphrase Extraction Based on Pre-Trained Language Model," *IEEE Access*, vol. 8, pp. 10896–10906, 2020, doi: 10.1109/ACCESS.2020.2965087.

[14] H. Ye and L. Wang, "Semi-supervised learning for neural keyphrase generation," in *Proceedings of the 2018 Conference on*

Empirical Methods in Natural Language Processing, EMNLP, 2018, pp. 4142–4153, doi: 10.18653/v1/d18-1447.
[15] X. Li, P. Lu, L. Hu, X. G. Wang, and L. Lu, "A novel self-learning semi-supervised deep learning network to detect fake news on social media," *Multimedia Tools and Applications*, vol. 81, no. 14, pp. 19341–19349, 2022, doi: 10.1007/s11042-021-11065-x.
[16] O. Terrada, S. Hamida, B. Cherradi, A. Raihani, and O. Bouattane, "Supervised machine learning based medical diagnosis support system for prediction of patients with heart disease," *Advances in Science, Technology and Engineering Systems*, vol. 5, no. 5, pp. 269–277, 2020, doi: 10.25046/AJ050533.
[17] O. Terrada, B. Cherradi, A. Raihani, and O. Bouattane, "Atherosclerosis disease prediction using Supervised Machine Learning Techniques," in *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology, IRASET 2020*, IEEE, Apr. 2020, pp. 1–5, doi: 10.1109/IRASET48871.2020.9092082.
[18] A. Conneau, D. Kiela, H. Schwenk, L. Barrault, and A. Bordes, "Supervised learning of universal sentence representations from natural language inference data," in *EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings*, 2017, pp. 670–680, doi: 10.18653/v1/d17-1070.
[19] M. Xia, A. Anastasopoulos, R. Xu, Y. Yang, and G. Neubig, "Predicting performance for natural language processing tasks," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 8625–8646, doi: 10.18653/v1/2020.acl-main.764.
[20] J. Ye, T. Gui, Y. Luo, Y. Xu, and Q. Zhang, "ONE2SET: Generating diverse keyphrases as a set," 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)*, 2021, pp. 4598–4608, doi: 10.18653/v1/2021.acl-long.354.
[21] L. Rassam, A. Zellou, and T. Rachad, "Empirical Study: What is the Best N-Gram Graphical Indexing Technique," in *International Conference on Big Data and Internet of Things (BDIoT)*, 2021, pp. 387–398, doi: 10.1007/978-3-031-07969-6_29.
[22] S. Al-Hagree, M. Al-Sanabani, M. Hadwan, and M. A. Al-Hagery, "An Improved N-gram Distance for Names Matching," in *2019 1st International Conference of Intelligent Computing and Engineering: Toward Intelligent Solutions for Developing and Empowering our Societies, ICOICE*, 2019, pp. 1–7, doi: 10.1109/ICOICE48418.2019.9035154.
[23] S. Al-Hagree and G. Al-Gaphari, "Arabic Sentiment Analysis on Mobile Applications Using Levenshtein Distance Algorithm and Naive Bayes," in *2022 2nd International Conference on Emerging Smart Technologies and Applications (eSmarTA), Ibb, Yemen, 2022*, IEEE, 2022, pp. 1–6, doi: 10.1109/eSmarTA56775.2022.9935492.
[24] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv*, Sep. 2016, doi: 10.48550/arXiv.1609.04747.
[25] L. Rassam, C. Aldiebesghanem, A. Zellou, and E. Ben Lahmar, "Fuzzy Logic-based N-gram Graph Technique for Evaluating Textual Documents Indexes," in *2022 4th International Conference on Computer Communication and the Internet (ICCCI)*, Chiba, Japan, IEEE, Jul. 2022, pp. 78–82, doi: 10.1109/ICCCI55554.2022.9850268.
[26] L. Ajallouda, K. Najmani, A. Zellou, and E. H. Benlahmar, "Doc2Vec, SBERT, InferSent, and USE Which embedding technique for noun phrases?," in *2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*, IEEE, Mar. 2022, pp. 1–5, doi: 10.1109/IRASET52964.2022.9738300.
[27] L. Rassam, M. Raoui, A. Zellou, and M. H. El Yazidi, "Analyzing Textual Documents Indexes by Applying Key-Phrases Extraction in Fuzzy Logic Domain Based on A Graphical Indexing Methodology," in *2022 International Conference on Computational Modelling, Simulation and Optimization (ICCMSO), Pathum Thani, Thailand*, IEEE, Dec. 2022, pp. 122–126, doi: 10.1109/ICCMSO58359.2022.00035.
[28] M. Krapivin, A. Autaeu, and M. Marchese, "Large Dataset for Keyphrases Extraction," *Technical Report DISI-09-055*, no. DISI-09-055, pp. 1-6, 2009.
[29] S. N. Kim, O. Medelyan, M. Y. Kan, and T. Baldwin, "SemEval-2010 Task 5: Automatic keyphrase extraction from scientific articles," in *Proceedings of the 5th International Workshop on Semantic Evaluation*, 2010, pp. 21–26.
[30] M. Sokolova and G. Lapalme, "A systematic analysis of performance measures for classification tasks," *Information Processing and Management*, vol. 45, no. 4, pp. 427–437, 2009, doi: 10.1016/j.ipm.2009.03.002.

BIOGRAPHIES OF AUTHORS

Latifa Rassam received her Eng. degree in Web Engineering and Mobile Computing from the High National School of Computer Science and System Analysis (ENSIAS) at Mohammed V University in Rabat, Morocco in 2018. Currently, she is a Ph.D. student in the Software Project Management (SPM) Research Team at ENSIAS, and also at Mohammed V University in Rabat. Her research interests include primarily the natural language processing, data indexing and the internet of things domains where she is the author/coauthor of over 7 research publications. She can be contacted at email: rassamlatifa@gmail.com and latifa_rassam@um5.ac.ma.

Ahmed Zellou received his Ph.D. in Applied Sciences at the Mohammedia School of Engineers, Mohammed V University, Rabat, Morocco 2008, his habilitation to supervise research work in 2014. He becomes full professor in 2020. His research interests include interoperability, mediation systems, distributed computing, data, indexing, recommender systems, data quality, and semantic web where he is the author/coauthor of over 100 research publications. He can be contacted at email: ahmed.zellou@um5.ac.ma.