IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 22 January 2024, accepted 13 February 2024, date of publication 20 February 2024, date of current version 26 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3368003

RESEARCH ARTICLE

Expert Profile Identification From Community Detection on Author-Publication-Keyword Graph With Keyword Extraction

WILLIAM FU AND SAIFUL AKBAR, (Member, IEEE)
School of Electrical Engineering and Informatics, Bandung Institute of Technology, Bandung 40132, Indonesia
Corresponding author: William Fu (williamfu@office.itb.ac.id)

This work was supported by the School of Electrical Engineering and Informatics, Institut Teknologi Bandung.

ABSTRACT Expert profiling aims to discover the expertise of an author. This task is useful for identifying the research groups existing within an organization as well as measuring the similarities between authors' expertise. Thus, identifying areas of expertise becomes a critical part of this task, especially in cases where the publications are unannotated. Commonly used topic modeling methods such as Latent Dirichlet Allocation still fall short in determining the number of topics automatically and discovering the hierarchical relationships between topics. To solve these issues, we adopted a graph-based approach which constructs a graph from publication features such as authors and keywords (Silva et al., 2018). We applied the Louvain algorithm repeatedly to discover the topics with hierarchical order automatically. We utilize keyword extraction methods to generate keywords for each respective publication to handle the missing values. We perform experiments to determine the optimum HPMI value. Results showed that graphs constructed from default and SIFRank keywords with transformation weights of $\alpha = 0.5$ and $\beta = 1.0$ produce topics with the best HPMI score. We evaluate the profiles from this method (CDT) with ATM as the baseline. It is shown that CDT produces better MAP, MRR, and nDCG scores than ATM. The work in this manuscript shows how community detection and keyword extraction could be utilized in expert profiling tasks. Our observation shows that the Louvain algorithm used only cluster publications into one topic, and thus still has limitations in classifying multidisciplinary publications. Further development could be done to handle such publications and increase the quality of keywords.

INDEX TERMS Expert profiling, keyword extraction, community detection.

I. INTRODUCTION

Scientific works are being published at an increasing rate [2] along with the growth of knowledge and technology improvements. To better understand the myriad collection of publications, Academic Social Networks (ASN) are used to represent the entities involved within a publication like authors, keywords, venues, etc. This representation is useful for scholarly mining tasks such as research interest discovery, expert recommender systems, and community detection [3].

Expert profiling is a part of an expert retrieval task that associates individual authors with their relevant domain topics [4]. Within an organization's scope, this task helps future collaborators in finding a candidate author's research topics. Moreover, expert profiling helps identify areas of expertise available within an organization. These areas of expertise can be predetermined by previous expert annotators or automatically discovered through the topic identification process. The latter approach is commonly adapted if the areas of expertise have not previously been known from the collection of publications. LDA-based topic modeling approaches [5], [6] are commonly used to identify topic groups in an unsupervised way.

LDA is a generative model for discovering topics in a text corpus. This model represents the topic as a distribution of words. A text document in LDA is considered as a mixture of

The associate editor coordinating the review of this manuscript and approving it for publication was Fu Lee Wang.

different topics; thus, the words determine the proportion of topics in that document [5]. LDA uses an iterative process to learn the distribution of topics and the words in those topics. LDA is frequently used to discover general insights from a large collection of texts [7], [8]. However, topic words generated with LDA are still considered ambiguous and too common [1]. Moreover, the number of topics needs to be determined beforehand as a parameter of the model.

As an alternative to these issues, a graph-based approach is introduced that discovers topic groups from a heterogeneous graph. The constructed graph represents the entities related to publications [1], such as authors and keywords. Louvain algorithm is then applied in this approach to discover publication groups. This community detection algorithm automatically determines the number of communities i.e., topics [1], [9]. The detection process is done repeatedly for each group to identify hierarchical relations within each topic [10]. This approach offers different granularities of topics to describe an author's expertise.

Constructing a heterogeneous graph requires features such as authors and keywords from a publication [1], [10] as shown in Figure 1. The constructed graph consists of three kinds of vertices: authors, publications, and keywords. This heterogeneous graph follows a star-schema structure, with the publication node acting as the star node, and the author and keyword nodes acting as attribute nodes [1]. In this case study, we found that a significant portion of publications in our dataset have missing keywords, even though this feature is integral in describing each topic group and measuring the similarity between publications. Thus, handling these missing values is a vital part of completing this pipeline. To complete this feature, we put the title and abstracts into use, since both features combined sufficiently represent the overall content of a publication [11]. We apply keyword extraction methods to discover keywords from the title and abstract of a publication.

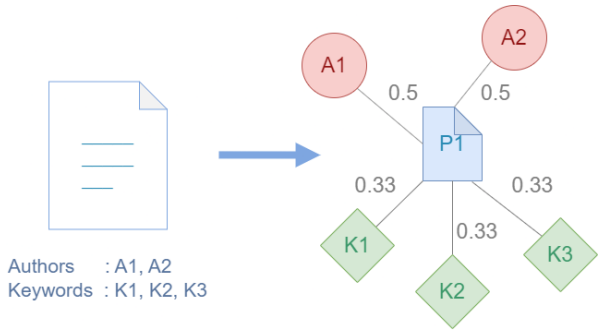

FIGURE 1. Heterogeneous graph construction process.

Keyword extraction methods have also been performed on another expert profiling approach [9], albeit to construct a homogeneous word co-occurrence graph. However, the keyword extraction methods used in this approach still lack document context understanding [12]. More recent keyword extraction methods utilize language models as external knowledge in providing the text's semantic context [12], [13].

This paper introduces a pipeline for discovering individual expertise profiles from an author's scientific publications. The method presented in this paper utilizes a heterogeneous graph [1] along with keyword extraction methods to generate relevant keywords from the title and abstract of a publication [9]. Keyword disambiguation is also performed to reduce any ambiguous keywords present in the constructed graph.

The contributions of this paper are as follows. First, we employed an unsupervised keyword detection algorithm in the graph-based expert profiling model. Second, we utilized *string similarity* measures using the Longest Common Subsequence (LCS) based method to disambiguate similar keywords. Lastly, we measure the quality of profiles from this method with retrieval metrics [14]. These evaluation metrics have not been used for profiles generated from the graph-based approach.

The remaining section of this manuscript consists of sections as follows. In section II, we briefly discussed the studies performed to gain a better understanding of the task at hand. Afterward, we introduce our pipeline and experiment design for research interest discovery in section III. Section IV shows the experiment results along with the analysis. Lastly, we present our conclusion in section V.

II. RELATED WORKS

Expert profiling is useful to automatically describe the expertise of an individual, based on information such as their works and previous collaboration. This makes it practical for future collaborators to search for people whose expertise matches their needs. This task complements the expert search task whose aim is to search a list of people whose expertise matches an expertise query [4]. The expert profiling task provides the details of a person that is discovered from an expert search system.

In this section, we briefly present the methods used for automated expert profile discovery from scientific publication dataset. These include concise explanations of graph-based approaches for expert profiling and the Louvain method for community detection.

A. GRAPH-BASED EXPERT PROFILING

Graph structures intuitively represent the relationship of entities related to scientific publications such as authors, keywords, or publication venues. These entities could be represented as vertices to a homogeneous graph, such as a co-authorship or word co-occurrence network, or a heterogeneous network known as an academic social network (ASN) [3].

In expert profiling tasks, both homogeneous and heterogeneous graph structures could be used to represent a relationship between documents with their authors and keywords [1], [9], [10]. These approaches utilize a collection of document keywords to illustrate the contents of a publication. The resulting groups were then used as topic descriptions

according to their relevance. In this paper, two approaches are going to be discussed, both of which use community detection methods to discover topic groups towards different graph representations.

1) KEYWORD CO-OCCURRENCE NETWORK

Co-occurrence networks are commonly used in natural language processing tasks to represent co-occurrence relationships between terms in a document. In this approach, automated keyword extraction (AKE) methods are employed to generate vertices to a co-occurrence graph. As the frequently used keyword extraction methods, graph-based methods such as TextRank [15], RAKE [16], and MultipartiteRank [17] are adapted to extract in the experiments.

The use of co-occurrence relationships still has not taken the semantics of the word into account. This additional information would potentially increase the quality of topics generated. Thus, a semantic similarity network is included to consider the semantics of keywords in the topic discovery process [9]. To achieve this, the first article on Wikipedia search results for each keyword is used to determine the semantic similarity. The semantic similarity between two words is defined as the number of overlapping keywords between the two corresponding articles. A hybrid network is then constructed by combining the co-occurrence and semantic similarity network.

Observation results showed that topics discovered from the hybrid network have lower quality than those from the co-occurrence network. This was due to the use of a semantic network removing the community structure within the network overall, which could be seen from the decrease in modularity score. It was also shown that the use of author-generated keywords resulted in topics with better quality, in contrast to the extracted keywords with lower thematic values [9].

To discover groups within the documents, the Louvain community detection is applied to the keyword network. This method automatically detects the number of topics, eliminating the need to determine the number of topics beforehand. The algorithm is applied recursively to create a topic hierarchy until the intended granularity.

The profile of an author a is determined by observing each publication p associated with the author a to each topic t. The score of a publication p to topic t is calculated by the ratio of overlapping keywords from publication p to keywords from t. An author's profile is then described as the average score of publications p their corresponding topic t.

2) AUTHOR-PUBLICATION-KEYWORD NETWORK

A hierarchical topic structure could also be created by applying a community detection algorithm to a heterogeneous graph [1]. In this approach, a HIN is constructed from features such as publication, author, keyword, and ISI field. In this network, the last three entities act as attribute nodes to the former entity. Thus, the constructed HIN could be seen as having a star schema topology with publications that act as star nodes.

Before applying the Louvain algorithm, we transform the author-publication-keyword graph into a homogeneous similarity graph. This homogeneous graph represents the similarity between publications based on their keywords and patterns of collaboration. In this graph, the edge weight of a pair of publications is the sum of the edge weights of adjacent nodes (author, keyword, ISI field) between the two publications. The Louvain algorithm is applied recursively to discover hierarchical relationships within the topics. This approach also relies on the nature of the Louvain algorithm to automatically determine the number of topics. After the partitioning process, the ranks of entities (authors, keywords, ISI fields) within each group are ranked by using the PageRank centrality measure.

To measure the quality of topics, evaluation is performed using the HPMI (Heterogeneous Pointwise Mutual Information) metric. This metric is a modification of the PMI metric [18] with additional handling for heterogeneous vertex types, as seen in (1).

$$HPMI\left(v^x, v^y\right) = \begin{cases} \dfrac{2}{k(k-1)} \sum_{1 \le i \le j \le k} \log\left(\dfrac{p(v_i^x, v_j^y)}{p(v_i^x)p(v_j^y)}\right) & x = y \\ \dfrac{1}{k^2} \sum_{1 \le i,j \le k} \log\left(\dfrac{p(v_i^x, v_j^y)}{p(v_i^x)p(v_j^y)}\right) & x \neq y \end{cases} \tag{1}$$

The variables x and y in this case indicate the vertex type being compared (author, keyword, or ISI field). As proven in (1), HPMI handles the case of measuring the quality of two different vertex types ($x \neq y$). The calculation is done for k number of samples of the most relevant vertex in each topic group.

The publications are automatically associated with a topic group from community detection. Thus, the profile of an author could be defined as the average of topics associated with every of that author's publications. The weight of each topic indicates the relevance of those topics to the author.

B. AUTOMATIC KEYWORD EXTRACTION (AKE)

AKEs are useful for discovering phrases that perfectly represent the content of a document [19]. Generally, the keyword extraction task is considered an unsupervised problem since it determines whether a portion of text gives significant relevance to the entire content [20]. In information retrieval systems, keyword extraction is useful to construct indexes that assist document searches, as well as to classify individual documents.

Keyword extraction tasks are commonly viewed as an unsupervised task that selects a subset of terms from a longer text, although some supervised methods are also explored to complete such tasks [20]. Several keyword extraction methods commonly utilize a co-occurrence network to find candidate keyphrases and rank them as vertices in the graph.

These methods include TextRank [15], RAKE [16], and MultipartiteRank [17]. Other methods involve a statistical-based approach, such as YAKE [21] which utilizes terms statistical metrics to capture the context of terms within a document. Lastly, embedding-based methods are being developed to provide semantic context information in keyword extraction processes. The development of these methods is supported by the relevance that language models such as BERT [22] and GPT [23] are gaining. These pre-trained models are getting more involved in natural language tasks like text classification [24], text summarization [25], [26], and text retrieval [27]. Embedding-based approaches in extracting keywords could be seen in methods such as SIFRank [12], KBIR, and KeyBART [13] which involve embeddings from pre-trained language models, such as ELMo [28], RoBERTa [29], and BART [30] respectively, in determining the keywords of a document.

C. LOUVAIN COMMUNITY DETECTION

The Louvain algorithm [31] is a frequently used method to discover community structures within a network. This method utilizes a Greedy approach in grouping vertices by optimizing the average modularity score [32]. The modularity metric measures the quality of a partition by its connectivity within and outside of the community as notated by (2).

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \gamma \frac{k_i k_j}{2m} \right) - \delta(c_i, c_j) \tag{2}$$

Generally, each iteration of Louvain's algorithm consists of two phases. In the first phase, each vertex i in the graph is considered as its community. For each vertex i, each adjacent vertex j is examined with vertex i to observe the difference in modularity value if both vertices are in the same community. Vertex i is then assigned to the community that yields the highest positive modularity gain.

In the second phase, a weighted network is constructed with the vertices being the communities identified from the first phase and the edge weights are the sum of the weights between communities. The resulting graph will be utilized for the next iterations until the modularity value no longer increases.

D. PROFILE EVALUATION

Retrieval tasks aim to provide users with the most relevant documents for the given query. Added to that, this task also aims to rank documents in order of their relevance [33]. Thus, a few retrieval metrics are proposed to measure the quality of a retrieval system. Each of these metrics considered various aspects of retrieval results, such as ranking order or relevance scores.

As part of the expert retrieval task, expert profiling assessment methods measure the relevance of generated terms as well as the order of the terms [14]. Expert profiling tasks could be seen as a retrieval task with an author's name as its query and the results consist of term-score pairs. Retrieval metrics frequently used for expert profiles include mean reciprocal rank (MRR), mean average precision (MAP), and normalized discounted cumulative gain (nDCG) [34]. In this manuscript, these metrics measure the precision of generated topics concerning their order and relevance.

1. Mean reciprocal rank (MRR)
   MRR measures the relevance of query results by the rank of the first relevant result [35]. This value is measured by the inverse of the rank of a relevant item as shown in (3).
   $$MRR = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{rank(i)} \tag{3}$$
2. Mean Average Precision (MAP)
   MAP is the mean of Average Precision (AP) which assesses the relevance of each element in a sequential profile. Formally, the AP equation for R relevant elements can be denoted in (4). Relevance $rel(r)$ is a binary function with value 1 if element r is relevant, and 0 if it is considered irrelevant, while precision P(r) is the precision value up to the r-th element.
   $$AP(n) = \frac{\sum_{r=1}^{n} P(r) \cdot rel(r)}{R} \tag{4}$$
3. Normalized Discounted Cumulative Gain (nDCG)
   The nDCG metric measures the quality of ranking from query results. The value of this metric is the normalized result of the discounted cumulative gain (DCG) which utilizes a discount function that increases as the rank goes lower. The DCG function prioritizes elements with the highest relevance to be placed in a high rank. Normalization of DCG is carried out by determining the ideal DCG value (IDCG), which is the maximum DCG value that can be obtained. This IDCG value is calculated by determining DCG on elements whose relevance is ordered in a non-decreasing manner. The formal nDCG notation equation for the first n elements can be seen in (5) and (6).
   $$nDCG(n) = \frac{DCG(n)}{IDCG(n)} \tag{5}$$
   $$DCG(n) = \sum_{r=1}^{n} \frac{rel(r)}{log_2(r+1)} \tag{6}$$

III. PROPOSED METHOD

In this paper, we present a pipeline for expert profile discovery through heterogeneous network structure and keyword extraction methods. This section presents the detailed process of our proposed method as well as our experiment design.

A. EXPERT PROFILING PIPELINE

The pipeline to generate expertise profiles consists of two general stages: identifying relevant terms and topics, as well as mapping individual authors to their respective topics [4]. To identify topic groups within the dataset, a heterogeneous graph is constructed as seen in Figure 2. This graph illustrates the relationship between a document with its authors and keywords.

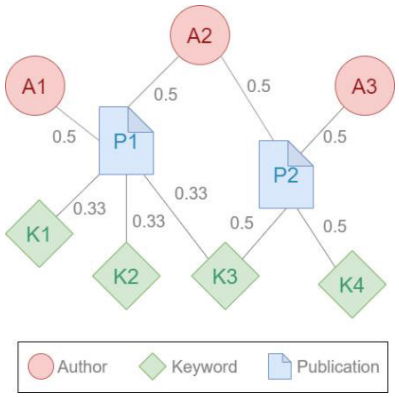

FIGURE 2. Heterogeneous graph illustration.

The keyword feature is useful in clustering the publications as well as in describing the generated topic groups. However, observations from the dataset exploration process showed that some publications did not have keywords by default. Thus, the keyword extraction method is then utilized to handle these missing keywords.

The overall pipeline for expert profiling in this paper is visualized in Figure 3. We utilized two datasets in this pipeline: a publication dataset containing the metadata of publications and a lecturer dataset containing the full names of lecturers within the faculty. The former dataset provides information for discovering topics and expert profiles, whilst the latter is used for author name disambiguation. We extract and disambiguate these keywords in the heterogeneous graph construction process.

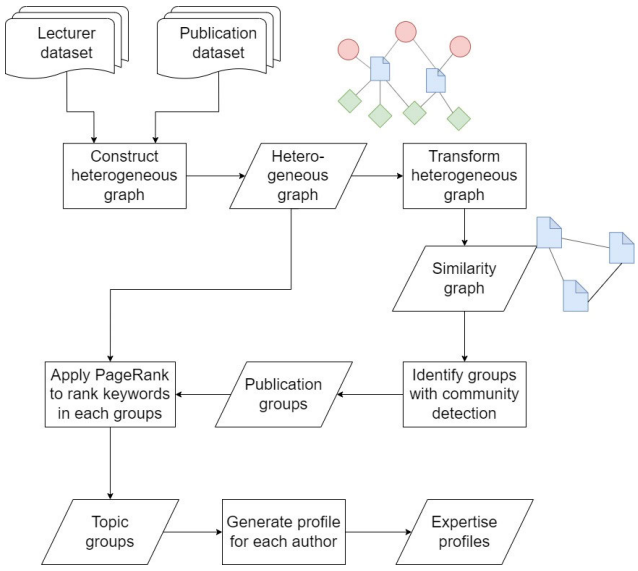

FIGURE 3. Expert profiling pipeline.

In this approach, several keyword extraction methods are employed to fill these features on the dataset. The methods used in this approach are a combination of methods employed in previous studies as well as a new keyword extraction method. This research seeks to involve these keyword extraction methods to build heterogeneous graphs.

The graph construction process follows the flowchart in Figure 4. We select features from our publication dataset such as author names, titles, abstracts, and default keywords. Keyword extraction is applied to the title and abstracts. The extracted keywords will then be combined with the default keywords as vertices to the graph. There were two combinations of keywords examined here: with and without disambiguation. After that, these keywords, along with a list of authors, are used as vertices to a heterogeneous graph. Author name disambiguation is also performed in this pipeline by examining the initials of an author's name. We use the lecturer dataset to provide the full name of an author as reference.

Keyword disambiguation is performed to reduce ambiguity between the extracted keywords. By measuring the string similarity of keywords with the longest common subsequence (LCS)-based metric, we aim to group the similar strings that refer to the same concept or entity.

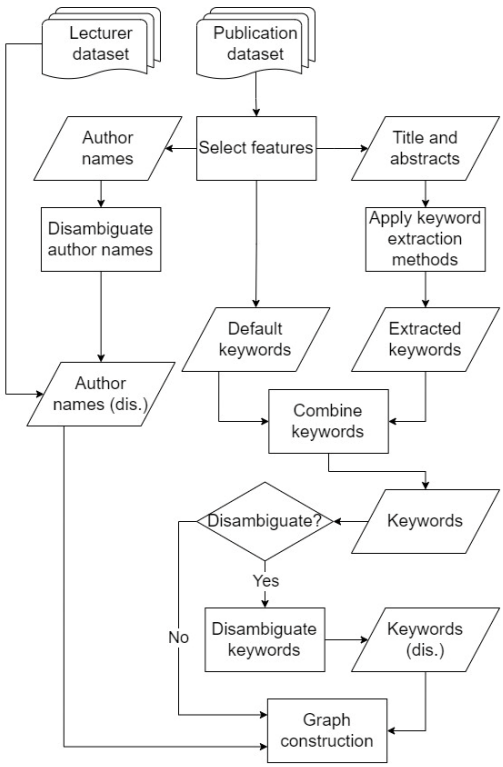

FIGURE 4. Graph construction flow.

The utilization of this lexical approach is chosen to keep the keywords associated with the dataset thematically consistent with the content in the text. Semantic similarity is not observed to provide more information between keywords in heterogeneous graphs like the previous work [9]. This is done to keep the keywords consistent with the contents of its source publication. As an illustration, the keywords *neural network* and *computer vision* are two terms that are semantically close

to each other, as *neural network* is one of the methods utilized in *computer vision* tasks [36]. However, publications that involve *neural networks* in their approach do not necessarily discuss *computer vision* research and vice versa.

Thus, the keyword disambiguation steps are as follows. First, a list of unique keywords is formed from the publication dataset. Then, for each pair of keywords contained in the list, the similarity value with the LCS metric will be measured. Then, the keywords will be grouped according to their similarity with a predetermined threshold. In this research, the experiment will utilize a similarity threshold value of 0.85. Figure 5 displays the string similarity measurement process in the keyword disambiguation process.

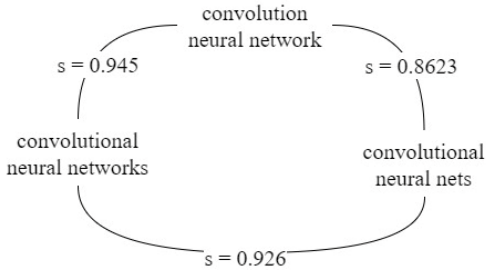

FIGURE 5. Keyword disambiguation process.

We adapt the method proposed in previous work to transform the heterogeneous network into a homogeneous similarity graph [1]. By transforming the graph, we could apply the Louvain method to group the publication vertices into groups. This method is performed recursively to discover a hierarchical relationship within the topics. In this approach, we group the topics into two hierarchy levels. The edge weights l between two publication vertices p_1 and p_2 in the similarity graph are measured according to (7). Variables α and β are variable weights that determine the proportion of author-publication edge and keyword-publication edge weights, respectively. Figure 6 displays an example of graph transformation between two publication vertices.

$$l_{p1,p2} = \alpha(l_{p1,A} + l_{p2,A}) + \beta(l_{p1,K} + l_{p2,K}) \tag{7}$$

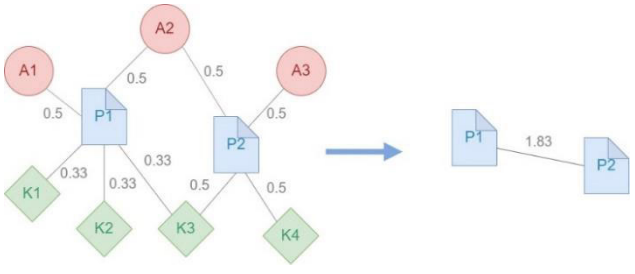

FIGURE 6. Graph transformation process.

The ranking step is performed after community detection is applied. To determine the ranking of authors and keywords in each community, PageRank centrality analysis [37] ranks the author and keyword features on the heterogeneous graph. This centrality measure is chosen as the ranking method because it could determine the importance of a vertex by its relations to neighboring vertices. It is also consistent with the intuition in co-authorship and co-occurrence graphs in which entities that are heavily involved with other entities generally have more significant influence.

An author's expertise profile is defined as the probability distribution of topic groups. At the end of the topic identification stage, each document in the dataset is associated to a topic group. The topic probability distribution is first determined by mapping the publications of an individual author with its corresponding topic group. Then, the topic probability is determined by calculating the frequency of occurrence of the topic of that author's publication. The probability distribution is represented as an n-dimensional vector. Figure 7 illustrates the discovery of expertise profiles from community detection results.

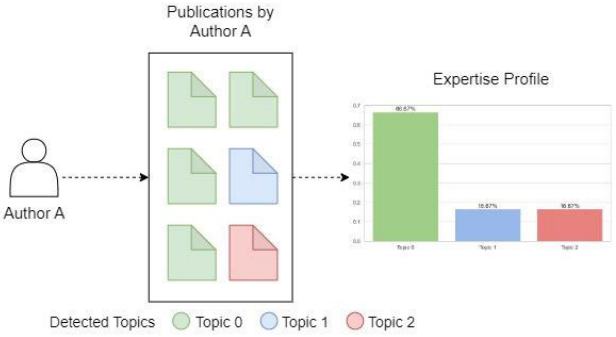

FIGURE 7. Individual expertise profile description.

B. EXPERIMENT DESIGN

In this research, experiments are conducted to determine which combination of observable factors produces the optimum quality topics. The observable variables that were compared in this experiment are as follows.

1. Keyword extraction method
   This experiment compares keyword extraction methods utilized in previous approaches [9] (TextRank, RAKE, MultipartiteRank) as well as newly proposed keyword extraction methods (YAKE, SIFRank, KBIR, KeyBART). A total of five keywords were extracted from each publication to represent the keywords in the graph, which follows the average number of default keywords within the dataset.
2. Keyword combination
   We conduct experiments to compare two keyword vertex representations in the heterogeneous graph. The first representation uses the combination of the default keyword publication with the extracted keywords whilst the second representation uses only the extracted keywords. This comparison determines the effectiveness of keyword extraction methods in discovering topics and expert profiles.

3. Keyword disambiguation
   In this experiment, we compare the constructed graph with or without keyword disambiguation. This experiment aims to discover the impact of keyword disambiguation on the quality of topics produced.
4. Edge weights transformation
   To observe the influence of keyword and author relationships, variables α and β are used to adjust the proportion of author edge weights and keyword edges respectively when transforming the initial heterogeneous graph into a homogeneous similarity graph. This comparison had been observed in previous study [1] to obtain the optimum topic quality. The variations of α and β pairs compared in this experiment are $\alpha = 1.0$, $\beta = 1.0$; $\alpha = 0.5$, $\beta = 1.0$; and $\alpha = 1.0$, $\beta = 0.5$.

IV. EXPERIMENTAL RESULTS

In this section, we present the steps taken to the implementation phase of the research. This includes graph construction, topic discovery, as well as individual profile evaluation.

A. DATASET

Our publication dataset contains 12,242 records of information which corresponds to scientific papers published by authors within ITB School of Electrical Engineering and Informatics (STEI) until July 2023. The records in this dataset are collected from the faculty publication database and publicly available scientific publication repositories such as Google Scholar and IEEE Xplore. Our dataset contains seven features that describe a publication's metadata as seen in Table 1.

TABLE 1. Publication dataset features.

No	Feature	Type
1	title	string
2	abstract	string
3	year	integer
4	authors	string
5	keywords	string
6	doi	string
7	venue	string

The initial publication dataset contains duplicated records, documents with missing abstracts, as well as nonrelevant publication documents, such as table of contents, author index, or welcome speech. Thus, we remove any such records in the dataset. This filtering process results in 7,145 publications to be represented as a heterogeneous network as a first step to identifying individual expertise profiles.

The distribution of the contributors group size per publication can be identified in Figure 8. From this figure, it could be seen that most publications were authored by two to four authors, with 30.76 percent of the collected publications

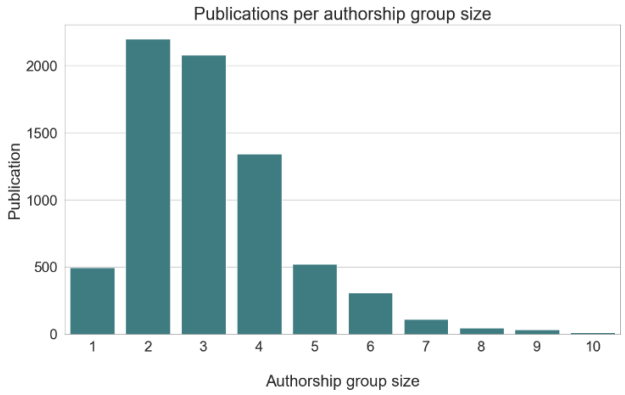

FIGURE 8. Authorship size distribution.

created by two authors (2,196 publications) and 29.06 percent created by three authors (2,077 publications).

In this paper, we use the title and abstract from each publication as the source document for keyword extraction. This is done since both these features combined already represent the overall content of a scientific publication [11]. Most text in this dataset consists of 100 to 200 tokens. The distribution of tokens in this dataset is displayed in Figure 9.

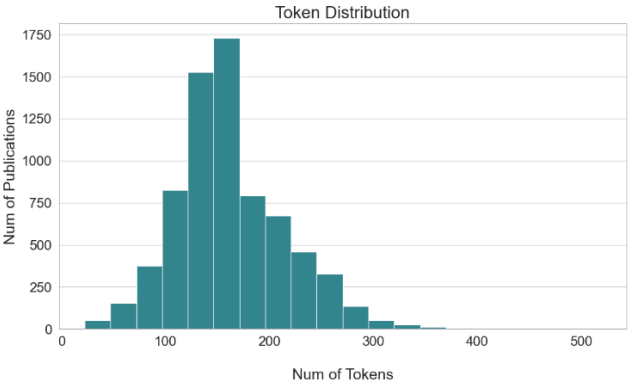

FIGURE 9. Publication token length distribution.

The keywords feature has a considerable number of missing values in this dataset, with 54.21 percent of the dataset (3,874 publications) being publications without providing keywords. To complete the keywords in the dataset, several keyword extraction methods will be compared at the heterogeneous graph construction stage. We can observe this in the graph in Figure 10.

In addition, we use another dataset that contains lecturer names and their assigned research groups within the faculty. This dataset would be used to disambiguate authors' names during the graph construction stage. There are nine general research groups within the faculty. From this dataset, we also discover the size distribution of each research group available within the faculty, as shown in Figure 11.

B. GRAPH CONSTRUCTION

Each heterogeneous graph constructed has 7,145 document vertices, 6,832 author vertices, and 22,527 document-author

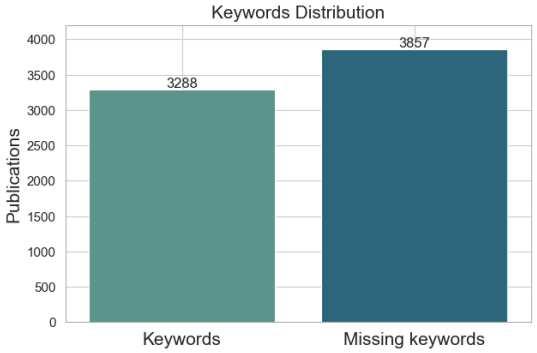

FIGURE 10. Missing keywords size.

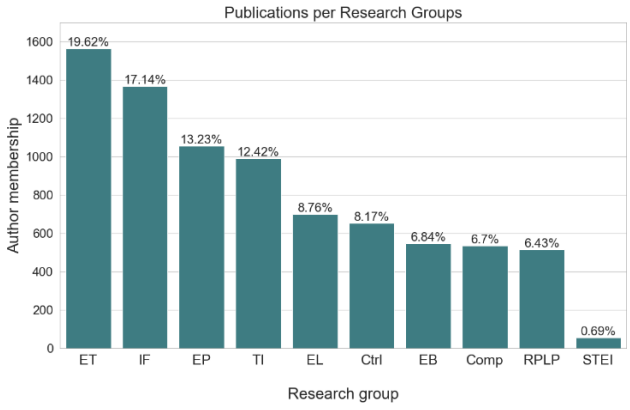

FIGURE 11. Research group size distribution.

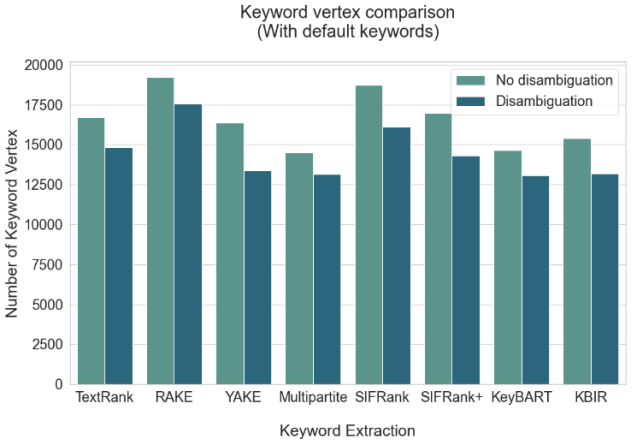

FIGURE 12. Keyword vertex size comparison (With default keywords).

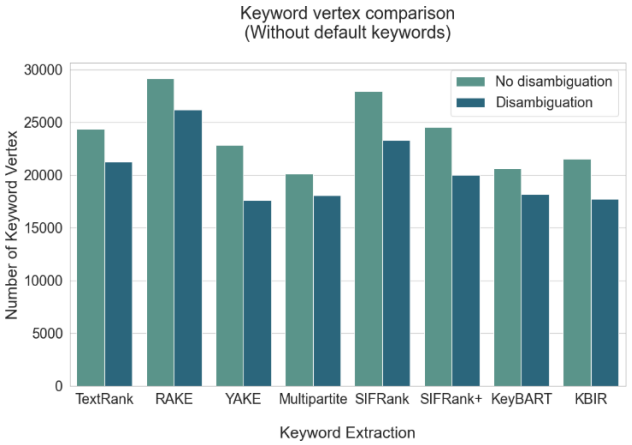

FIGURE 13. Keyword vertex size comparison (Without default keywords).

edges. The number of keyword vertices and keyword-document edges varies depending on the combination of other observation factors. In general, graphs that utilize default keywords have fewer keyword vertices compared to graphs that contain only extracted keywords alone. This is due to the default keywords having more occurrences in more than one publication. The common occurrences of these keywords could also be observed during the graph transformation phase into a similarity graph. Graphs that contain the default keywords have higher density values since they have more similarity edges.

The heterogeneous graphs with keyword disambiguation have lower keyword vertex count because the disambiguation process reduces the variation of keyword string that refers to the same entity. The vertex count comparison between the constructed graphs before and after keyword disambiguation graphs can be observed in Figure 12 for graphs with default keywords and Figure 13 for graphs without default keywords.

C. EXTRACTED KEYWORDS ANALYSIS

The keyword extraction methods utilized in this research are part of the extractive approach, which means that each keyword is a term that appears in the source document. From the observation of the entries in the dataset, it is known that the keywords provided by default are abstractive, meaning that some of these keywords never appear in the source document. Most of the default keywords in this dataset are found in publications from IEEE Xplore. The keywords in these publications are derived from expert annotations according to the entries in INSPEC. This illustration can be seen in Figure 14.

Through observation of the topic discovery results, most default keywords found were abstractive keywords. The abstractive keyword extraction approach has not been utilized in this research because of the limited number of abstractive methods available. The abstractive approach could be considered as a form of multilabel text classification task, so this approach requires a list of reference keywords as the initial labels of each text document.

To determine the initial label for all publications in this dataset, the identification of experts who are skilled and familiar with the two common fields within the scope of STEI publications, namely computing and electrical engineering, is required. Thus, an extractive approach was chosen to identify keywords from the existing text. Moreover, the extractive approach can be considered as an initial approach

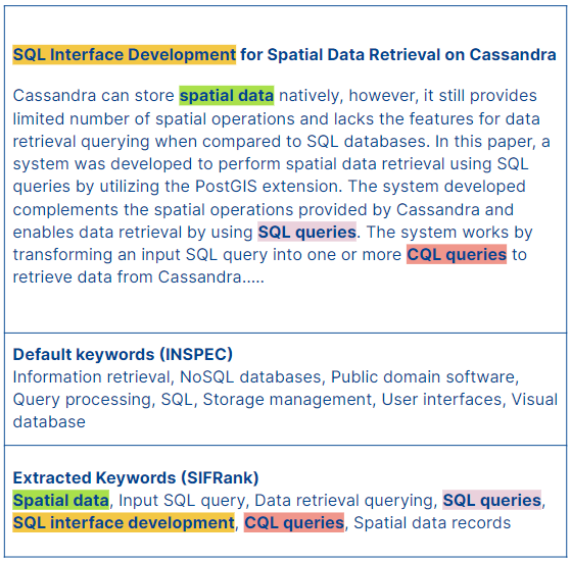

FIGURE 14. Default and extracted keywords comparison.

to unsupervised keyword identification. The findings in this experiment are expected to assist further development of expertise profiling solutions in the future.

D. TOPIC IDENTIFICATION ANALYSIS

This section discusses the topic identification results generated from the community detection method. In general, the quality of these topics would be compared by measuring their HPMI scores. We elaborate further on these results for each of the observable factors in the experiment.

1) KEYWORD EXTRACTION METHOD

The different keyword extraction methods affect the HPMI between keywords (KK) values significantly. On the comparison of graphs without keyword disambiguation, the SIFRank extraction method combined with the default keywords produces the best average HPMI value. Meanwhile, RAKE, YAKE, and SIFRank produce the best average HPMI value in graphs without default keywords for weights $\alpha = 1.0, \beta = 1.0$; $\alpha = 0.5, \beta = 1.0$; and $\alpha = 1.0, \beta = 0.5$ respectively.

On the graph with keyword disambiguation, the SIFRank keyword extraction method gives the best average HPMI value at the weight combination $\alpha = 1.0, \beta = 1.0$. In contrast, RAKE gives the best value at the weight combination $\alpha = 0.5, \beta = 1.0$ and $\alpha = 1.0$, $\beta = 0.5$ for graphs with default keywords. For graphs without default keywords, the RAKE method gives the best value for the weight combinations $\alpha = 1.0, \beta = 1.0$ and $\alpha = 0.5, \beta = 1.0$. Finally, the YAKE extraction method provides the best value for the weight combination $\alpha = 1.0, \beta = 0.5$.

From these observations, we found that RAKE and SIFRank extraction methods produce the most optimum quality of topics for a given combination of observation factors. This shows that these methods produce keywords that appear in more than one document, thus increasing the chance of token occurrence in the HPMI value calculation process.

The diagram in Figure 15 shows a comparison of the number of keywords that appear in more than five documents in the dataset with the extraction method used. It is shown that SIFRank and RAKE methods have the least number of keywords appearing in more than five documents compared to other extraction methods. The low occurrence rate would then increase the probability value of a pair of keywords appearing on the same topic.

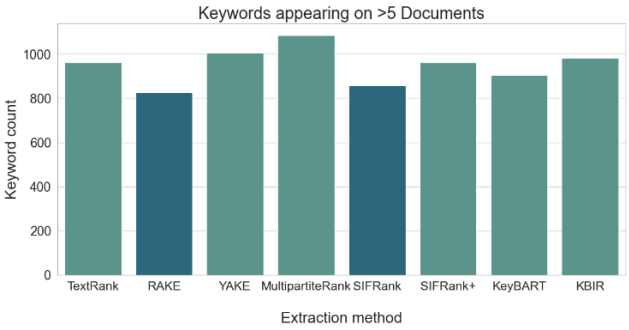

FIGURE 15. Keywords appearing in more than 5 documents.

2) KEYWORD COMBINATION

Topic groups generated from graphs with default keywords have a larger average HPMI value compared to graphs without default keywords. This is evidenced in Figure 16 which shows the comparison of average HPMI value with keyword combinations. The significant difference in average value is caused by the difference between HPMI for keyword vertices (KK). The graph that is constructed with the default keywords has more overlapping keywords that appear in more than one document. This increases the chance of a pair of keywords on the same topic appearing in the same document, thus increasing the HPMI score.

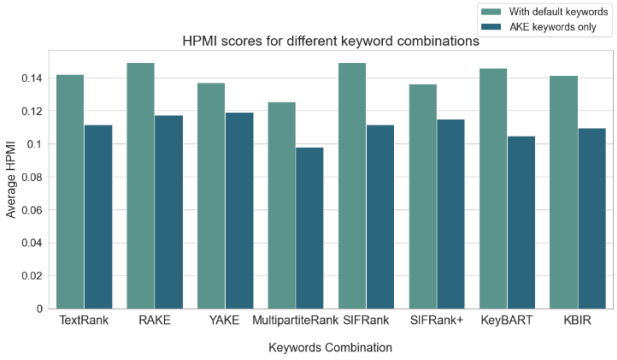

FIGURE 16. HPMI scores comparison for different keyword combinations.

3) KEYWORD DISAMBIGUATION

The utilization of keyword disambiguation provides different changes in the average HPMI value for each extraction

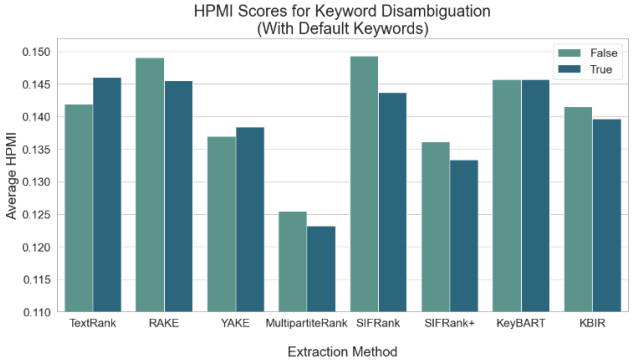

FIGURE 17. HPMI value comparison with disambiguation (graphs with default keywords).

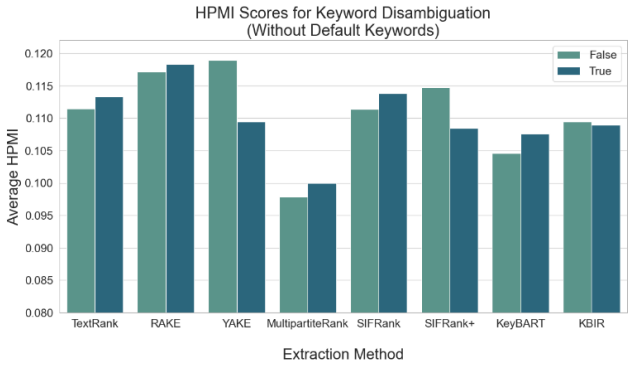

FIGURE 18. HPMI value comparison with disambiguation (graphs without default keywords).

method used. As an illustration, Figure 17 and Figure 18 present the graph comparison for the combination of author weight $\alpha = 0.5$ and keyword weight $\beta = 1.0$.

From the comparison we observed in Figure 17, the use of disambiguation increases the average value of HPMI for TextRank and YAKE extraction methods on the graph with default keywords. Meanwhile, using keyword disambiguation decreases the value for YAKE, MultipartiteRank, SIFRank, and SIFRank+ extraction methods.

The comparison in Figure 18 observes HPMI values for graphs without default keywords. This observation also proves the different effects of keyword disambiguation with the extraction methods. In this case, disambiguation increases the HPMI value for TextRank, RAKE, MultipartiteRank, SIFRank, and KeyBART extraction methods.

From our observation, we noticed that there are several error cases in the disambiguation process. While the string similarity measure could disambiguate keywords such as *convolution neural network* and *convolutional neural nets*, the method also produces some errors, with unrelated keywords such as *graph structure* and *graphene structure* being considered the same concept. These errors impact the quality of topics being produced depending on the extraction method being used.

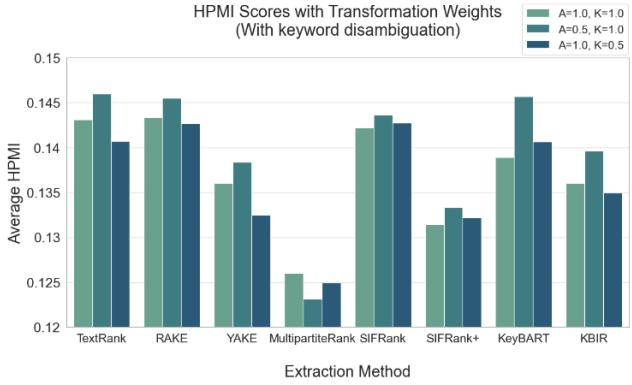

FIGURE 19. HPMI vs edge weights comparison on graphs with keyword disambiguation.

4) EDGE WEIGHTS TRANSFORMATION

The average value of HPMI scores can be seen in Figure 19 and Figure 20 for graphs without keyword disambiguation and with disambiguation. In both cases, the combination of variables $\alpha = 0.5$ and $\beta = 1.0$ gives a better average HPMI than the other two combinations. This is due to the lower author weight α increasing the influence of keyword similarity to weight edges in the similarity graph. These weighted edges are utilized in the topic identification process with the Louvain method. Thus, documents are clustered on the same topic because they have the same keywords, more so than having the same author. This then increases the chance of keywords in one topic group appearing in the same document. In general, the observation factors involved in this experiment significantly affect the HPMI value component for keywords (KK).

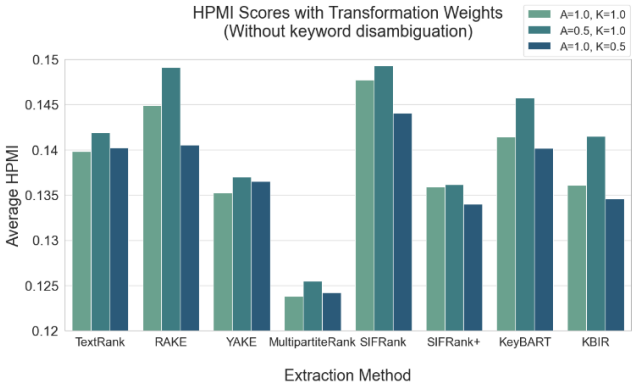

FIGURE 20. HPMI vs edge weights comparison on graphs without keyword disambiguation.

E. TOPIC EVALUATION

We compare HPMI values between each combination of factors globally to determine expert profiles that will be used to be evaluated by STEI lecturers. The average HPMI scores between combinations of observation factors are described in Table 2. All combinations shown in this table involve graphs

TABLE 2. Top combinations with best average HPMI.

No	Disambiguation	AKE	α	β	Avg. HPMI
1	**F**	**SIFRank**	**0.5**	**1.0**	**0.1493**
2	F	RAKE	0.5	1.0	0.1491
3	F	SIFRank	1.0	1.0	0.1477
4	F	KeyBART	0.5	1.0	0.1457
5	T	RAKE	0.5	1.0	0.1457

that use default dataset keywords. Through this observation, it is concluded that the combination of author weight $\alpha = 0.5$ and keyword weight $\beta = 1.0$ provides the best average HPMI value.

F. PROFILE GENERATION

The profiles identified from the optimum method are compared with keywords from the ATM model as a baseline since this method is a common approach utilized for topic detection with mapping to the authors. The optimum number of ATM model topics is determined by comparing the coherence value of NPMI to the number of topics. From this approach, we decided to use the ATM model with 16 topics as a baseline to construct the profiles.

The profile evaluation was conducted towards 22 STEI lecturer samples; 11 of whom were from the computer science (CS) branch of the faculty and another 11 from the electrical engineering (EE) branch of the faculty. This grouping ensures the sampling process provides as close a representation as possible to the expertise of the authors within STEI.

We use the mean average precision (MAP) metrics to measure how many relevant topics are in the resulting profile from both methods. In this study, the relevance value which is on a scale of 0-4 will be mapped into a binary label 0/1 with 2 as its threshold, meaning topics that score below 2 will be labeled as 0. Table 3 displays a comparison of the MAP@5 scores of the profiles produced by the two methods, namely the method of community detection, which we shortened here as CDT, and the method of the Author Topic Model (ATM). The CDT is measured to result in a better MAP score; thus, it is considered to produce profiles with topics that are more relevant to STEI authors. The CDT method provides a better MAP@5 score on both subsets of the samples.

TABLE 3. MAP@5 score comparison.

Method	MAP@5		
	EE	CS	Average
CDT	0.7081	0.7811	**0.7446**
ATM	0.6379	0.6045	**0.6212**

The MRR score comparison measures the quality of profiles by observing only the first occurrence of relevant topics.

TABLE 4. MRR score comparison.

Method	MRR		
	EE	CS	Average
CDT	0.9394	0.8939	**0.9167**
ATM	0.9091	0.7727	**0.8409**

Table 4 shows the comparison of MRR scores for both methods. In general, the CDT method provides better MRR scores in the samples of both disciplines. This observation matches the MAP measure. Both methods produce better MRR scores for profiles in the electrical engineering branch.

We compare nDCG metrics evaluation to measure the quality of topic rankings on profiles. Table 5 shows the comparisons between the nDCG@5 scores of the CDT method with the ATM method as baseline. In general, the nDCG score shows that both methods provide a good ranking quality to the actual expertise of the author. This is achieved because, for most of the profiles, both methods correctly place less relevant topics in a lower order. This evaluation also shows that the CDT method provides a better nDCG@5 score than ATM.

TABLE 5. nDCG@5 score comparison.

Method	nDCG@5		
	EE	CS	Average
CDT	0.9391	0.9260	**0.9326**
ATM	0.9264	0.9225	**0.9244**

Our observation shows that keywords within the same topic have varying scope sizes. As an example, one of the topics identified includes keywords with a broader scope such as *learning (artificial intelligence)* and *natural language processing,* as well as those with a more specific scope such as *support vector machine* and *sentiment analysis*. This makes keywords with a more general scope promoted more in the topic identification process, thus obstructing the discovery of more specific expertise. These more specific keywords help users determine the uniqueness of authors in the same scientific category. For example, keywords like *sentiment analysis* and *text summarization* describe a better detail of an author's expertise than the general term *natural language processing*. Domain-specific ontologies such as the ACM Computing Classification System (CCS) [38] or Computer Science Ontology (CSO) [39] could be utilized for further development in determining the level of granularity of keywords per topic. However, the graph-based expert profiling model in this study was still able to produce terms that were more meaningful than the collection of unigrams from the LDA-based topic modeling technique.

The MAP score indicates several irrelevant topics in the generated author's profile. One of the reasons is the nature of the Louvain algorithm which produces disjoint communities, so a publication could only be classified into one topic. This

grouping has limitations in handling multidisciplinary publications involving more than one scientific topic. For example, one of the publications on datasets involving hidden Markov model techniques in *software security* studies can only be classified under the topic of *learning (artificial intelligence)*. This makes the authors who contributed to the publication are considered to have more significant expertise on the topic of *learning (artificial intelligence)* compared to the topic of *software security*. Thus, future developments with overlapping community detection techniques could be adopted to handle multidisciplinary publications.

V. CONCLUSION AND FUTURE WORK

In this expertise profiling task, keyword extraction methods are utilized and compared for constructing author-publication-keyword heterogeneous graphs. Experiments are conducted to compare the quality of topics with various keyword methods. The use of string similarity method is also added to reduce ambiguity in the generated keyword list.

Experiment shows that extraction methods affect the HPMI value between keyword vertices (KK) significantly. This also applies to other observation factors in the experiments, which include keyword combination as well as transformation weights. We find that the topics provided from the graph with default *keywords* and extracted keywords by SIFRank without keyword disambiguation and transformed with weights $\alpha = 0.5$, $\beta = 1.0$ provided the optimum quality topics. This combination is utilized for further profile evaluation, comparing them to our baseline model. Moreover, the profile evaluation score shows that profiles from this approach are generally more relevant than ATM as the baseline method.

For future works, an abstractive keyword extraction method could be utilized with the keyword list being determined in advance. The keyword groups generated from this research can be utilized as one of the supplemental sources in determining the keyword list beforehand as classification labels. Other similarity measurement methods could be utilized to improve connectivity between keyword features. The use of other approaches would hopefully be able to improve the quality of document clustering. Further development of this approach could utilize other community detection methods, especially in grouping multidisciplinary publications.

REFERENCES

[1] J. Silva, P. Ribeiro, and F. Silva, "Hierarchical expert profiling using heterogeneous information networks," in *Proc. Int. Conf. Discovery Sci.*, 2018, pp. 344–360, doi: 10.1007/978-3-030-01771-2_22.

[2] L. Bornmann and R. Mutz, "Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references," *J. Assoc. Inf. Sci. Technol.*, vol. 66, no. 11, pp. 2215–2222, Nov. 2015, doi: 10.1002/asi.23329.

[3] X. Kong, Y. Shi, S. Yu, J. Liu, and F. Xia, "Academic social networks: Modeling, analysis, mining and applications," *J. Netw. Comput. Appl.*, vol. 132, pp. 86–103, Apr. 2019, doi: 10.1016/j.jnca.2019.01.029.

[4] K. Balog, "Expertise retrieval," *Found. Trends Inf. Retr.*, vol. 6, nos. 2–3, pp. 127–256, 2012, doi: 10.1561/1500000024.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[6] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," 2012, *arXiv:1207.4169*.

[7] A. Älgå, O. Eriksson, and M. Nordberg, "Analysis of scientific publications during the early phase of the COVID-19 pandemic: Topic modeling study," *J. Med. Internet Res.*, vol. 22, no. 11, Nov. 2020, Art. no. e21559, doi: 10.2196/21559.

[8] A. Glazkova, "Identifying topics of scientific articles with BERT-based approaches and topic modeling," in *Proc. Pacific-Asia Conf. Knowl. Discovery Data Mining*, 2021, pp. 98–105, doi: 10.1007/978-3-030-75015-2_10.

[9] P. C. Belém, (2018). *Temporal Research Interests Discovery Using Co-Occurrence Keywords Networks*. [Online]. Available: https://api.semanticscholar.org/CorpusID:134656482

[10] C. Wang, J. Liu, N. Desai, M. Danilevsky, and J. Han, "Constructing topical hierarchies in heterogeneous information networks," *Knowl. Inf. Syst.*, vol. 44, no. 3, pp. 529–558, Sep. 2015, doi: 10.1007/s10115-014-0777-4.

[11] S.-W. Kim and J.-M. Gil, "Research paper classification systems based on TF-IDF and LDA schemes," *Human-Centric Comput. Inf. Sci.*, vol. 9, no. 1, p. 30, Dec. 2019, doi: 10.1186/s13673-019-0192-7.

[12] 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.

[13] M. Kulkarni, D. Mahata, R. Arora, and R. Bhowmik, "Learning rich representation of keyphrases from text," in *Proc. Findings Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 891–906, doi: 10.18653/v1/2022.findings-naacl.67.

[14] R. Berendsen, M. de Rijke, K. Balog, T. Bogers, and A. van den Bosch, "On the assessment of expertise profiles," *J. Amer. Soc. Inf. Sci. Technol.*, vol. 64, no. 10, pp. 2024–2044, Oct. 2013, doi: 10.1002/asi.22908.

[15] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in *Proc. Conf. Empirical Methods Natural Lang. Process.* Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 404–411. [Online]. Available: https://aclanthology.org/W04-3252

[16] S. Rose, D. Engel, N. Cramer, and W. Cowley, "Automatic keyword extraction from individual documents," in *Text Mining*. Chichester, U.K.: Wiley, 2010, pp. 1–20, doi: 10.1002/9780470689646.ch1.

[17] F. Boudin, "Unsupervised keyphrase extraction with multipartite graphs," 2018, *arXiv:1803.08721*.

[18] D. Newman, J. H. Lau, K. Grieser, and T. Baldwin, "Automatic evaluation of topic coherence," in *Proc. Annu. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol.* New York, NY, USA: Association for Computational Linguistics, 2010, pp. 100–108.

[19] K. S. Hasan and V. Ng, "Automatic keyphrase extraction: A survey of the state of the art," in *Proc. 52nd Annu. Meeting Assoc. Comput. Linguistics*, vol. 1. Stroudsburg, PA, USA: Association for Computational Linguistics, 2014, pp. 1262–1273, doi: 10.3115/v1/p14-1119.

[20] E. Papagiannopoulou and G. Tsoumakas, "A review of keyphrase extraction," 2019, *arXiv:1905.05044*.

[21] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, and A. Jatowt, "YAKE! Keyword extraction from single documents using multiple local features," *Inf. Sci.*, vol. 509, pp. 257–289, Jan. 2020, doi: 10.1016/j.ins.2019.09.013.

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*.

[23] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving language understanding by generative pre-training," OpenAI, Tech. Rep., 2018. [Online]. Available: https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

[24] G. Danilov, T. Ishankulov, K. Kotik, Y. Orlov, M. Shifrin, and A. Potapov, "The classification of short scientific texts using pretrained BERT model," in *Studies in Health Technology and Informatics*. Amsterdam, The Netherlands: IOS Press, 2021, doi: 10.3233/SHTI210125.

[25] S. Bano and S. Khalid, "BERT-based extractive text summarization of scholarly articles: A novel architecture," in *Proc. Int. Conf. Artif. Intell. Things (ICAIoT)*, Dec. 2022, pp. 1–5, doi: 10.1109/ICAIoT57170.2022.10121826.

[26] S. Liu and C. G. Healey, "Abstractive summarization of large document collections using GPT," 2023, *arXiv:2310.05690*.

[27] X. Tian and J. Wang, "Retrieval of scientific documents based on HFS and BERT," *IEEE Access*, vol. 9, pp. 8708–8717, 2021, doi: 10.1109/ACCESS.2021.3049391.

[28] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," 2018, *arXiv:1802.05365*.
[29] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.
[30] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," 2019, *arXiv:1910.13461*.
[31] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech., Theory Exp.*, vol. 2008, no. 10, Oct. 2008, Art. no. P10008, doi: 10.1088/1742-5468/2008/10/p10008.
[32] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 69, no. 2, Feb. 2004, Art. no. 026113, doi: 10.1103/physreve.69.026113.
[33] S. Khalid, S. Wu, and F. Zhang, "A multi-objective approach to determining the usefulness of papers in academic search," *Data Technol. Appl.*, vol. 55, no. 5, pp. 734–748, Oct. 2021, doi: 10.1108/dta-05-2020-0104.
[34] M. Sanderson, "Test collection based evaluation of information retrieval systems," *Found. Trends Inf. Retr.*, vol. 4, no. 4, pp. 247–375, 2010, doi: 10.1561/1500000009.
[35] N. Craswell, "Mean reciprocal rank," in *Encyclopedia of Database Systems*. Boston, MA, USA: Springer, 2009, p. 1703, doi: 10.1007/978-0-387-39940-9_488.
[36] D. Bhatt, C. Patel, H. Talsania, J. Patel, R. Vaghela, S. Pandya, K. Modi, and H. Ghayvat, "CNN variants for computer vision: History, architecture, application, challenges and future scope," *Electronics*, vol. 10, no. 20, p. 2470, Oct. 2021, doi: 10.3390/electronics10202470.
[37] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Comput. Netw. ISDN Syst.*, vol. 30, nos. 1–7, pp. 107–117, Apr. 1998. [Online]. Available: http://www-db.stanford.edu/~backrub/google.html
[38] B. Rous, "Major update to ACM's computing classification system," *Commun. ACM*, vol. 55, no. 11, p. 12, Nov. 2012, doi: 10.1145/2366316.2366320.
[39] A. A. Salatino, T. Thanapalasingam, A. Mannocci, F. Osborne, and E. Motta, "The computer science ontology: A large-scale taxonomy of research areas," in *Proc. Int. Semantic Web Conf.*, 2018, pp. 187–205, doi: 10.1007/978-3-030-00668-6_12.

WILLIAM FU received the bachelor's degree in informatics from Bandung Institute of Technology, in 2022, where he is currently pursuing the master's degree in informatics. His research interests include data analytics, natural language processing, and software development.

SAIFUL AKBAR (Member, IEEE) received the bachelor's and master's degrees in informatics from the Department of Informatics, Bandung Institute of Technology, Indonesia, in 1997 and 2002, respectively, and the Ph.D. degree in engineering science from Johannes Kepler University Linz, Austria, in 2007. He was a Visiting Researcher with the Norwegian University of Science and Technology (NTNU), from 2009 to 2010. He is currently an Assistant Professor with the School of Electrical Engineering and Informatics, Bandung Institute of Technology. His research interests include data and knowledge engineering, multimedia database and similarity retrieval, information extraction, data analytics, and visualization.