Word2vec-based Latent Semantic Indexing (Word2Vec-LSI) for Contextual Analysis in Job-Matching Application

Sukri Sukri*[1], Noor Azah Samsudin[2], Ezak Fadzrin[3], Shamsul Kamal Ahmad Khalid[4], Liza Trisnawati[5]
Faculty of Computer Science and Information Technology,
Universiti Tun Hussein Onn Malaysia (UTHM),Johor, Malaysia[1, 2, 3, 4, 5]
Department of Informatics Engineering, Universitas Abdurrab, Pekanbaru, Indonesia[1, 5]

***Abstract*—Job-matching applications have become a technology that provides solutions for making decisions about accepting and looking for work. The contextual analysis of documents or data from job matching is needed to make decisions. Some existing studies on the analysis of job-matching applications can use the Latent Semantic Indexing (LSI) method, which is based on word-to-word comparisons in the text. LSI has the advantage of contextual analysis. It can analyze amounts of data above 10,000 words. However, the conventional LSI method has limitations in contextual analysis because it uses the exact words but different meanings. Therefore, this paper proposes a new technique called word2vec-based latent semantic indexing (Word2vec-LSI) for contextual analysis, which is based on gen-sim as a multi-language word library. Then, modeling in text and wordnet and stopword as basic text modeling. We then used word2vec-LSI to perform contextual analysis based on the Irish (IE), Swedish (SE), and United Kingdom (UK) languages in the dataset (Jobs on CareerBuilder UK). The results of applying conventional LSI have an accuracy level of 79%, recall has a value of 79%, precision has a value of 62%, and Fi-Scor has a value of 70% with a similarity level of up to 50%. After implementing word2vec-LSI, it can increase accuracy, recall, and precision, and Fi-Scor both have 84% in contextual analysis, and the similarity level reaches up to 95%. Experiments confirm the usefulness of word2vec-LSI in increasing accuracy for contextual analysis applicable in natural language text mining.**

***Keywords*—Contextual; LSI; job-matching; text-base; word2vec**

I. INTRODUCTION

This research will develop latent semantic index (LSI) techniques that will be used to make job recruitment decisions by improving accuracy in contextual analysis on several job matching data. An LSI technique used before is analyzing job-matching data with comparisons based on words and sentences [1]. LSI techniques for job checking data analysis typically use position features and descriptions, while to obtain job information, text relationships use semantics through Single Value Decomposition models (SVD) [1].

So, in the context of LSI, SVD still refers to Singular Value Decomposition, a key method for reducing dimensions and analyzing semantic relationships between words in text.

According to LSI standards, comparing the exact words and sentences can only be carried out in job-matching data with many words in the features and similarity of words presented to obtain the accuracy and relevance of matching. Based on the matching results of job-matching data with the same word and different meanings, textual analysis of the text has not produced maximum relevance [2]. To overcome this, researchers propose extended LSI (eLSI) in contextual analysis on job matching applications (JMA) [3]. Therefore, job-matching data will be contextually analyzed using the description feature and compared with the recruitment feature.

Job matching is becoming increasingly popular, which is realized at different levels of the labor market and is associated with the overall situation of the national economy. High competition increases the need to make better use of work resources and creates a better fit between workers and workplaces [4] [5] [6]. A job matching model is used to identify suitable candidates for open positions based on skills, qualifications, and experience. The job matching model performs searches using keywords to match between job seekers and employers [3].

Previous research has applied the Latent Semantic Indexing (LSI) technique [1]. LSI is text indexing and an analysis method used to identify semantic patterns in documents that uses vector spaces to describe documents and terms. LSI cannot capture complex relationships or hidden contexts between words in text (linear representation) [7]. This model requires understanding of context, relationships between words, and deeper meanings [8] [9]. LSI often has to limit the number of dimensions (semantic concepts) used to represent documents that are difficult to interpret, Compute Scalability and Efficiency[7], Limitations of representation [10] [11]and sensitivity to document changes[12]. LSI tends better to understand concrete words and direct relationships between documents. However, in the analysis of texts for job matching, it is often necessary to understand abstract terms [13] [14] [15], Cognitive abilities [11] [16], and aspects of the prospective worker's personality [17].

LSI has implemented several text analysis models, including text grouping [15]. This technique cannot extract resume data[18]. In addition, LSI is weak in reading new synonyms in the document resume. LSI has limitations in contextual readability, so it needs to be extended by integrating contextual analysis with other algorithms such as Word2Vec.

*Corresponding Author

LSI is a method of indexing and analyzing text used to identify semantic patterns of documents. LSI uses vector spaces to describe documents and terms when analyzing text. LSI cannot capture complex relationships or contexts hidden between words in text (linear representations). In a job-matching model, understanding context, relationships between words, and deeper meanings are required [19][20][21]. The SI often has to limit the number of dimensions (semantic concepts) used to represent documents (difficult to interpret) [22][23][24][25], Sensitivity to changes in documents [14][26], and cognitive abilities [27][16], or aspects of the job candidate's personality.

Text Analysis Techniques are text mining or natural language processing (NLP) techniques used to analyze and extract information from text data. These techniques are important in converting unstructured text into structured data for various applications, including information retrieval, document classification, and more.

These are just a few of the many text analysis techniques available, and the choice of technique depends on the specific task and purpose of the analysis. Text analysis is important in extracting insights and information from large amounts of text data in various fields.

Contextual analysis is an approach or method used to understand, evaluate, or analyze an object, event, text, or situation by considering its context. This context can be environmental, social, cultural, historical, political, or other variables that can affect the understanding or interpretation of somethi [28]. In natural language processing (NLP), contextual analysis refers to understanding words, sentences, or text more deeply by considering the surrounding words or sentences. It is used in sentiment analysis and natural language understanding [29].

Contextual analysis for job-matching applications is an approach or method used in business and human resources to deeply understand the context in which the job-matching process occurs [28]. It involves carefully evaluating the various factors and variables that affect the matching between workers looking for work with available job openings. Contextual analysis in job matching applications aims to ensure that the matching between jobs and job seekers is done efficiently and effectively[30]. By understanding the deeper context, companies and human resource professionals can make better decisions in managing the job-matching process [30] [31].

Contextual analysis is a process that involves understanding and evaluating texts, data, or information in the broader context in which they are used. In this analysis, information is viewed in terms of words or sentences and by considering the external context that can affect the meaning or interpretation of the text [32]. Contextual analysis is very important in comparative research, as it investigates the importance of contextual conditions for causal relationships. Over the past few decades, many comparative studies have focused on how contextual conditions affect causal relationships [29].

Contextual Analysis in job matching is based on the understanding that conventional job matching methods that focus only on words or sentences have limitations in understanding the proper context of the job and the candidate's qualifications. Therefore, there is a goal to improve the accuracy of job matching by paying attention to the broader context in the process. In this context, a deeper understanding of the relationship between job descriptions and candidates' qualifications is required, including contextual aspects that may not be visible through word matching alone. An extended Latent Semantic Indexing (LSI) technique is used to extract meaning and semantic relationships between words in context. Thus, this contextual Analysis is expected to help produce more accurate and relevant job matching between candidates and job openings by considering the context better.

Job matching is a special collaborative recommendation system developed for an entertaining and commonly used job matching process to help users identify and select qualified applicants who meet the requirements required by any organization [33] [34] [35]. Job seekers and job recipients need job matching. Job-matching is also a platform to facilitate the recruitment process and is cost-effective and time-effective [36].

Job matching is controlling the right person with the right job based on the motivation and power inherent in the individual. This requires a thorough understanding of the job and the person under consideration [8][23]. This process is very beneficial in simplifying recruitment and improving cost efficiency and time effectiveness [37]. With job matching, job seekers can easily find job openings that match their qualifications, and employers can quickly find suitable candidates for the positions they need [23]. Many existing online recruitment platforms have developed a reliance on automated ways to match job seekers to job positions [38]. Intuitively, records of successful recruitment in the past contain important information that should be used for job matching of current people [23] [39] [40]. The following can be seen in Fig. 1 of the job matching search system framework.

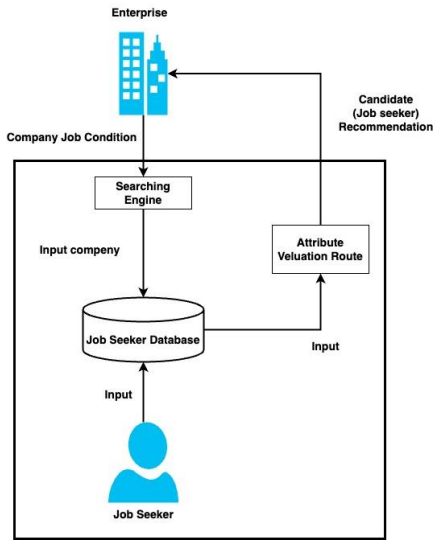


Fig. 1. Job-matching search system framework [17] [40].

Word2Vec is a powerful technique for word representation learning that captures semantic relationships between words based on patterns of their occurrence together. Word representations generated by the Word2Vec model have shown outstanding performance in various NLP tasks, such as sentiment analysis, named entity recognition, and machine translation. However, Word2Vec alone may not fully capture complex topic structures in text data.

In this research, we introduce a new approach, Word2vec-based Latent Semantic Indexing (Word2Vec-LSI), which combines the advantages of Word2Vec and LSI to improve the quality of topic modeling and contextual analysis in text documents. Word2Vec-LSI aims to bridge the gap between word representations and hidden semantic indexes by leveraging the semantic richness of Word2Vec representations while benefiting from the topic modeling capabilities of LSI. Our research explores the potential of Word2Vec-LSI in improving the accuracy and depth of topic modeling, especially in the context of contextual analysis. We evaluated this methodology on various text datasets from different domains, assessing its performance in capturing complex topics and contextual information in the text. The study contributes to developing cutting-edge text analysis techniques and promises many applications in information retrieval, content recommendation, and knowledge discovery. In the following sections, we will provide a detailed overview of the proposed Word2Vec-LSI methodology, outline the experiment setup, and present the results obtained.

Word2Vec is a vector representation algorithm that can understand the meaning of words based on their context in the text [41]. This technique allows the system to understand better the context of words in job descriptions and job seeker profiles. This is especially useful in addressing synonym and antonym problems, where Word2Vec can identify words with similar or opposite meanings, improving accuracy in matching [32]. Moreover, Word2Vec also helps understand the semantic hierarchy between words [41], so that the system can recognize that some words are subconcepts of more significant concepts.

In addition, Word2Vec can capture semantic relationships, such as the relationship between a subsidiary company and a central company or between junior and senior positions. With Word2Vec, job-matching systems can provide more accurate results by considering the context of the meaning of words, not just the similarity of words that align. This helps generate results that align with the criteria of job seekers and companies, which ultimately increases the efficiency and accuracy of the job-matching process. Word2Vec also helps overcome the challenges of matching more complicated jobs. For example, when keywords in a job description change or language variants are used, Word2Vec can help identify solid semantic relationships between those words. For instance, if a job posting searches for "software developer" and a candidate describes themself as a "programmer," Word2Vec will detect similarities in meaning and match them effectively.

In addition, Word2Vec also allows personalization in the job-matching process. By analyzing broader text such as CVs, cover letters, and candidates' employment history, Word2Vec can create unique vector representations for each candidate. This allows for a more tailored job search to an individual's abilities and experience, which often cannot be achieved with traditional keyword-based matching.

Lastly, Word2Vec also helps in reducing human errors in the recruitment process. Using this technology, companies can minimize bias in candidate selection and ensure that each candidate is assessed based on their suitability for the job. This contributes to creating a more fair and efficient recruitment environment, benefiting both the company and the job seekers. Thus, Word2Vec has great potential to improve the quality and accuracy of job-matching in the world of recruitment.

LSI has limitations in analyzing contextual resume documents [32]. LSI can only do the process of comparing the same words and sentences [1]. Based on the results of matching work data with the same word and different meanings, textual analysis of the text has not produced maximum relevance. As a search technique in the context of application matching jobs, the "extended" method aims to improve the matching accuracy in this application. This research introduces the extension of LSI Techniques aimed at understanding the context of job-matching. An approach is integrating Word2Vec to manage synonyms, antonyms, semantic hierarchies, and semantic relations. This integration results in the representation of data in dimensions used to measure similarity.

The main objective of this study is to optimize LSI techniques into extended LSI from contextual analysis using integration techniques with word2vec and evaluate using precision, recall, and F1-score. The testing process uses the Jobs on CareerBuilder UK dataset (description and resignation) and development using Python programming language on the Google collaboration platform. This research can contribute to the development of LSI Engineering. The Extended LSI technique will be one of the contextual analysis techniques in job-matching applications. This research can overcome conventional LSI's limitations that rely only on word frequency in text. More advanced extended LSI techniques can account for document contextual analysis to generate relevance from job matching applications.

II. MATERIALS

Fig. 2 is the data collection and preprocessing process used to perform text modeling in the job-matching context. For job-matching data analysis in the database, we collect words in employees' curriculum vitae (CV). Then, the words are processed by selecting the default word as comparison data using StopWord, Stemmer, and Tokenization. After obtaining the standard words, a comparison of meanings is carried out using gensim to obtain the corpus data set. The result of the comparison will get up to 10000 words [41].

Fig. 2 is explained that the data collection and pre-processing process in the context of job matching begins by retrieving data from various related sources, such as job search websites or internal company databases. The data consists of job descriptions that include details about the responsibilities, qualifications, and requirements for each job position. Once

the data is collected, the first step is to process it into individual words. This involves dividing text into separate tokens or words. Next, the text data is prepared through pre-processing, where steps such as removal of common words (stop words), text normalization, tokenization, and stemming or lemmatization are performed. Once the data is cleaned and prepared, a corpus is formed using tools such as Gensim, which allows the creation of theme modeling models. This corpus is a collection of documents or texts that have been processed and are ready for further analysis. The final step involves the formation of a final vocabulary, which is a collection of unique words from the entire corpus. Each word in this vocabulary has a numerical representation that can be used in subsequent natural language processing models. Thus, this process provides an important foundation for advanced analysis in the context of job matching, enabling the application of various natural language processing techniques to gain deeper insights from existing text data.

Fig. 3 shows the number of jobs that underwent displacement each year from 2001 to 2021. It can be seen that job movements have increased sharply, especially in the period from 2019 to 2021. On the graph, it can be seen that the number of job moves significantly increased during the period. This reflects the changing dynamics of the labor market over time, where workers have more opportunities to change jobs or find new jobs. Economic growth, industrial development, and changes in worker preferences may have influenced this job movement trend. Therefore, a deeper understanding of this data can provide valuable insight into changes in the labor market structure over the past two decades.

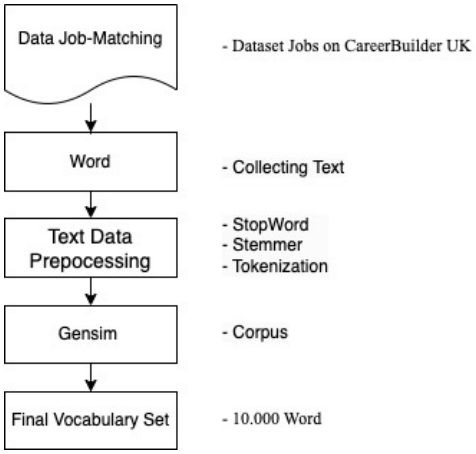
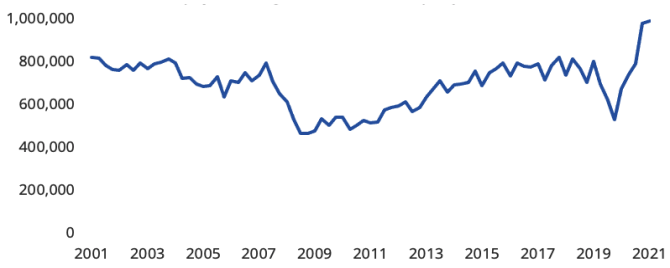


Fig. 2. Process of data collection and preprocessing.



Source : ONS Labour Force Survey [41]

Fig. 3. Job maching data with skills.

III. Proposed Methodology

Conventional latent semantic indexing (LSI) methods can only compare text in sentences. However, because this method only searches and compares the same text, it is unlikely to be able to carry out contextual analysis in job-matching applications that have many words with different meanings. Usually, contextual data in job-matching has different language and meaning, making it difficult to match between jobs and job recipients. So it can reduce the accuracy of contextual analysis in job-matching. To overcome this problem, we propose Word2Vec-based latent semantic indexing (Word2Vec-LSI) to improve contextual analysis and use Gensim as a library used to create a vector representation model of words in sentence [42]. So you can increase the recommendation area while maintaining as much accuracy as possible. This method is suitable for solving problems that can be contextually analyzed with various words and different languages having the same meaning Fig. 4. This is expected to provide significant benefits for job-matching applications.

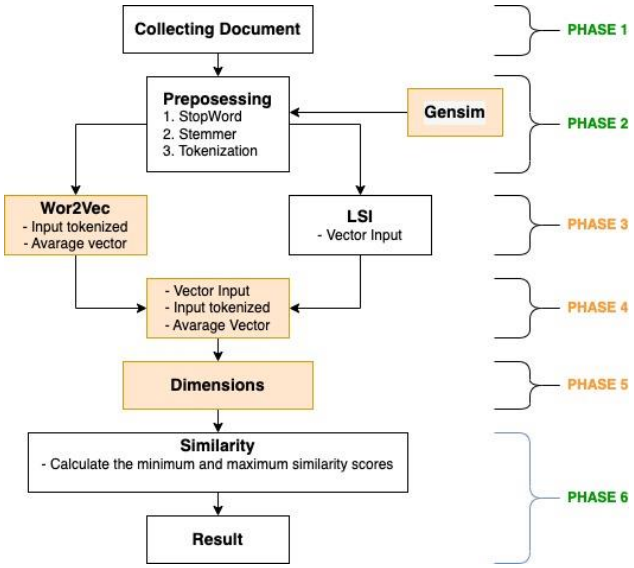


Fig. 4. Framework improved LSI.

The first phase, document collection, is an important step in the process of information processing and research that involves collecting relevant or necessary documents for a specific purpose. The first step in this process is to identify the sources of documents to use, whether they come from internal sources such as corporate databases or external sources such as the internet, digital libraries, or general data repositories. Next, the relevant documents are selected based on specific criteria such as topic, date, or document type.

Then, the documents are retrieved or downloaded from the source, often using tools or technology appropriate to the document type and its source. After collection, these documents often require a processing stage, including cleaning and preprocessing to remove irrelevant data and indexing to facilitate data search and management. Quality and accuracy in document collection have a significant impact on the final results of research, data analysis, or information system development that is being carried out.

Second phase, in the context of Latent Semantic Indexing (LSI), there are three important stages in text processing involving stemming, removal of stopwords, and tokenization.

1) Stemming is the process by which words in a text are transformed into their basic form or base words. The main goal of stemming is to address variations of words that have the same root. In other words, words with similar meanings but written with different variations will be identified as the same word. A simple example would be the words "run", "run", and "run around" which would be transformed into the basic form "run".

2) Removal of stopwords. Stopwords are common words that appear frequently in text but do not provide high semantic information. Examples of stopwords in English are "the", "and", "in", and the like. Removing stopwords helps focus on more informative and specific words in semantic analysis, so LSI results are more accurate.

3) Tokenization, in which text is divided into smaller units called "tokens". These tokens can be words, phrases, or even sentences, depending on the level of granularity required in the analysis. Tokenization allows text to be broken down into separate entities that can be counted in a document-term matrix representation within an LSI.

These three stages in text processing are important in word2vek and LSI, as they help reduce the dimensionality of words in document representations, eliminate less relevant information, and ensure that semantic analysis can be performed more effectively. By performing stemming, removal of stopwords, and tokenization, document text is well prepared for a more accurate and informative LSI process. Then, to get a collection of sentences, there needs to be a library using genisms. Gensim is a library for text modeling and natural language processing (NLP). The library is known for its ability to develop Word2Vec models in modeling various word vector techniques and other text processing.

Third phase, the integration between Word2Vec, Demension, and Latent Semantic Indexing (LSI) creates a more sophisticated approach to contextual analysis.

a) Word2Vec

- Word2vec is used as a representation of a word in a low-dimensional space that understands the context and semantics of words.
- Word2Vec generates a vector of words that represent the meaning of the word in its context. It describes the meaning of words in vector spaces and can be used for tasks such as meaning-based matching, classification, and sentiment analysis. However, Word2Vec has the disadvantage of not understanding the relationships between words in larger documents or underlying topics
- Word2Vec can be used to replace words in a document, which improves understanding of word context.
- Word2vec functions as a tokenization and average vector analysis, because it can get maximum results in reading words in sentences and average words that often appear

b) LSI

- LSI to analyze the document as a whole to identify latent patterns or topics.
- LSI can be used as an input vector that serves to vector words that will be used as word matching to be included in dimensions that will be applied in the same unity of meaning.
- LSI, on the other hand, is used to identify latent patterns or underlying topics in documents. It helps in a deeper understanding of document context and can be used for topic-based grouping and semantic search. However, LSI may be less accurate in representing individual word meanings

Fourth phase, vector Input: The first step is to generate a vector representation of the word using Word2Vec. This is done by training a Word2Vec model on a corpus of relevant texts. Once training is complete, the Word2Vec model will have a word vector representation for each word in the corpus. For example, if you have the sentence “I like machine learning”, each word (“I”, “like”, “machine”, “learning”) will have a word vector that explains its meaning in context. Input Tokens: Once you have a vector representation of words from Word2Vec, you need to parse the text document you want to analyze into words or tokens. This process is called "tokenization" and allows understanding the structure of the text and detailing each word in the document. For example, if you have the sentence "Natural language processing is very interesting", tokenization will decompose this sentence into individual words: "Natural", "language", "processing", "is" and "interesting." Average Vector: After parsing the words in a document, it can calculate the average word vector from Word2Vec for all the words in the document. This is done by adding up the word vector of each word in the document and then dividing it by the number of words. The result is a mean vector representing the document in the Word2Vec vector space. This average vector can then be used as a document representation in LSI analysis. Using this average vector as input, we can then apply LSI analysis to identify latent topics in the data collection. This is one way to integrate Word2Vec's word vector representation into LSI analysis and leverage the power of both for deeper text understanding.

Fifth phase, using word vectors, Word2Vec can integrate Word2Vec's advantages in understanding word meaning in context with LSI analysis that identifies latent topics in documents. This combination allows for deeper text analysis and a better understanding of the content of the document.

Sixth phase, in terms of maximum and minimum similarity is used to measure similarities or differences between sentences or concepts in sentences. Similarity maximum is used to identify sentences or concepts that are most similar to a particular reference document or reference concept. This concept represents the highest cosine similarity considered to be the most similar to the reference, the concept that is most different or not similar to the reference sentence. Just like the similarity maximum, it also involves calculating the similarity

of cosines, but this time the sentence or concept with the lowest cosine similarity value is considered to be the most different. Both maximum and minimum similarity play an important role in various text analysis applications, depending on the purpose of the analysis and the context.

IV. Experimental Results and Discussion

In the Results section, we compare the accuracy and relevance of words in each language Languages such as Irish (IE), Swedish (SE), and United Kingdom (UK).

A. Testing using LSI

Based on Fig. 5, trials using evaluation of the results of applying conventional LSI to matching using accuracy have a value of 79%, recall has a value of 79%, precision has a value of 62%, and Fi-Scor has a value of 70%. Based on Fig. 6, it shows that applying LSI to 500 documents has a similarity level of up to 50%. While previous studies have increased Accuracy to 82.5% [19]. So, it is necessary to increase the accuracy value in contextual reading using conventional LSI.

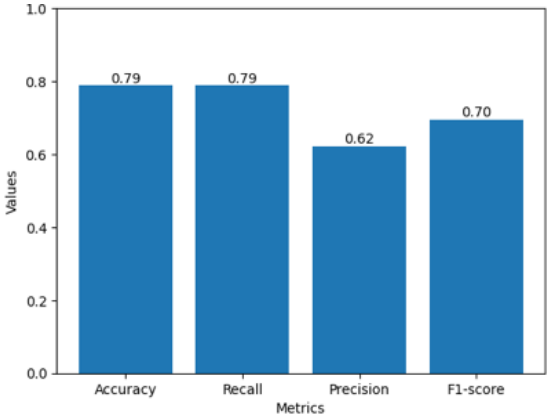


Fig. 5. Classification metrics LSI.

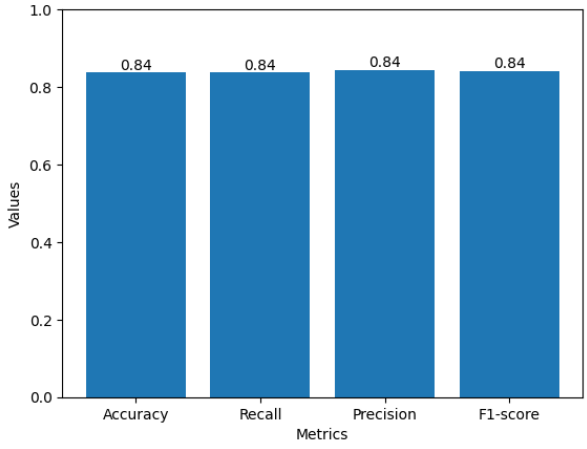


Fig. 6. Classification metrics for Word2Vec-LSI.

B. Testing using Word2Vec-LSI

Fig. 7 shows that the application of word2vec-LSI to 500 documents has a similarity level of up to 95%. In this study, we refer to Table I, which displays the three languages used for testing the dataset as an implementation of word2vec-LSI. The results of this experiment provide a deeper understanding of how this method works in different contexts. From these results, we can conclude that Ireland (IE) has a ratio of 95%, Sweden (SE) 95%, and England (UK) 96.6%.

These findings show that the word2vec-LSI implementation performs well in all three languages tested, with the United Kingdom (UK) achieving the highest ratio. This is important information, as it can assist researchers or practitioners in selecting appropriate methods for natural language processing tasks in various contexts and environments. In addition, these findings also provide valuable insights into understanding the extent to which word2vec-LSI-based representations of words and documents can be used effectively in language-based analysis.

Based on Fig. 8, the evaluation of the results of applying word2vec-LSI to matching using accuracy, recall, precision, and Fi-Scor is 84%.

The evaluation results documented in Table II show that evaluation method, namely accuracy value, which is 63.4%. However, it should be noted that these same results may cause confusion and need to be re-examined. In addition, the results of the evaluation illustrate that the combination of the use of Word2Vec and LSI currently has low performance. This can be largely affected by the use of threshold = 0.7. In this context, it is necessary to clarify how threshold changes affect the performance of the model or system. Furthermore, there are indications that the evaluation results can be improved by increasing the threshold value to 0.5, as seen in Table III.

Based on Table III, the percentage of test results increased when the threshold value is raised. This means the system becomes stricter in classifying data as positive so that more data is typed correctly. Conversely, if the threshold is lowered, the results will decrease as the system becomes more tolerant in classifying data as positive, which can increase false positives. In other words, threshold changes affect the balance between precision and recall (the ability to identify all positive instances), and this is an important consideration in determining how a model or classification system performs in a given context.

The result of a document's vector construction is a numerical vector representation that encodes information about the meaning and context of the document. These vectors can be used in various analyses to understand and group documents based on their similarity in vector spaces, including topic modeling, contextual analysis, and information retrieval. With approaches like Word2Vec-LSI, we can combine the advantages of Word2Vec word representation with LSI to produce a richer understanding of text sentences.

Based on the results of this research, the results of applying conventional LSI have an accuracy level of 79%, recall has a value of 79%, precision has a value of 62%, and Fi-Scor has a value of 70% with a similarity level of up to 50%. After implementing word2vec-LSI, it can increase accuracy, recall, and precision, and Fi-Scor both have 84% in contextual analysis, and the similarity level reaches up to

95%. This research also succeeded in contextual analysis in several languages, such as Irish (IE), Swedish (SE), and the United Kingdom (UK). Based on a comparison between conventional LSI and Word2Vec-LSI, accuracy can be significantly increased to 84% from 50% and applied to contextual analysis in job-matching applications.

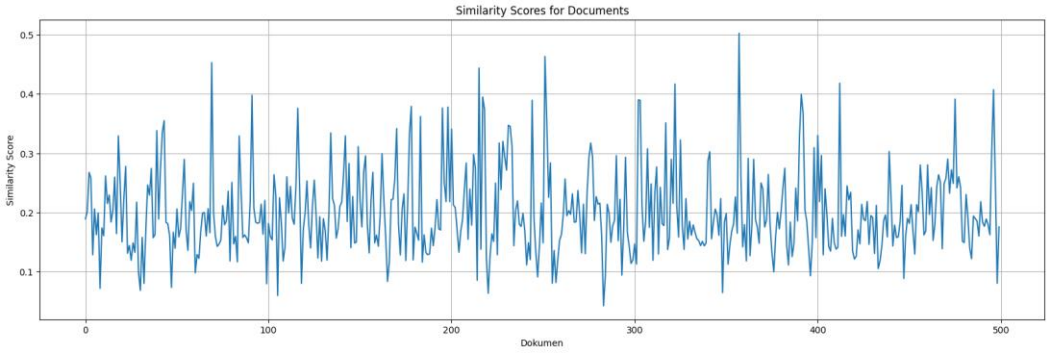


Fig. 7. Similarity score LSI.

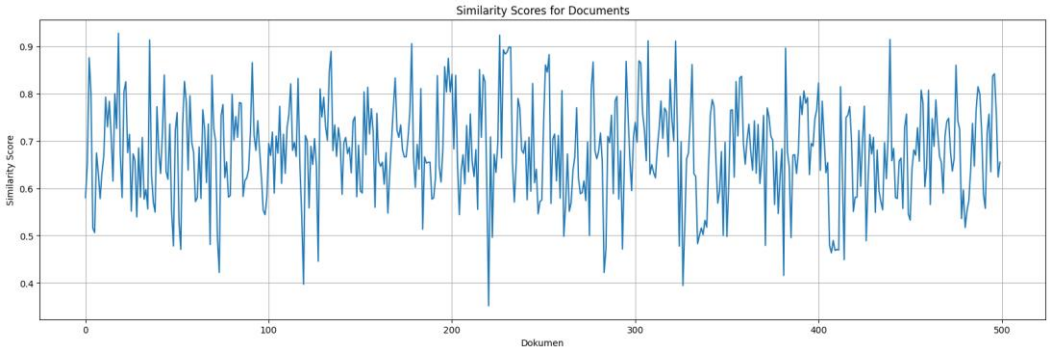


Fig. 8. Similarity score Word2Vec-LSI.

TABLE I. WORD2VEC-LSI-BASED CONTEXTUAL RESULTS FOR JOB MATCHING BASED ON THREE COUNTRIES IN THE DATA SET (JOBS ON CAREERBUILDER UK)

NO	WORD (Language)	Ratio (%)
1	Ireland (IE)	96
2	Sweden (SE)	95.5
3	United Kingdom (UK)	96.6

TABLE II. EVALUATION METRICS RESULTS USING ACCURACY WITH A THRESHOLD OF 0.7

Evaluation	**Score (%)**
Accuracy	0.634

TABLE III. RESULTS OF EVALUATION METRICS USING ACCURACY WITH THRESHOLD = 0.5

Evaluation	**Score (%)**
Accuracy	96.6

V. CONCLUSION

This research compares the performance of conventional Latent Semantic Indexing (LSI) and Word2Vec-LSI in analyzing text data across several languages, including Irish (IE), Swedish (SE), and British English (UK). The main findings of this research are as follows: Conventional LSI achieved an accuracy rate of 79%, recall of 79%, precision of 62%, and F1-Score of 70%, with a similarity rate of up to 50%. Meanwhile, Word2Vec-LSI succeeded in achieving a similarity level of up to 95% and increased accuracy, recall, precision, and F1-Score to 84%. This research also successfully analyzed text data in Irish (IE), Swedish (SE), and British English (UK), with British English achieving the highest ratio at 96.6%. Adjusting the threshold value also significantly affects the model performance, where a higher threshold value results in tighter classification and higher accuracy, while a lower threshold value leads to higher tolerance but lower accuracy. These findings highlight the importance of selecting appropriate methods for natural language processing tasks, especially in multilingual contexts, with Word2Vec-LSI offering deeper insight and higher accuracy in contextual analysis than conventional LSI. In conclusion, the combination of Word2Vec and LSI techniques proved effective in improving classification accuracy, particularly in job matching applications, with results that impact the consideration of threshold values in the performance evaluation of models and classification systems.

ACKNOWLEDGMENT

The author would like to thank the support provided by the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia (UTHM), especially all the tutors involved in completing this article, then Abdurrab University under the Abdurrab Pekanbaru Foundation which

has supported this research to obtain results which is satisfying.

REFERENCES

[1] F. G. Balazon, A. A. Vinluan, S. C. Ambat, and Q. City, "Job Matching Platform Using Latent Semantic Indexing and Location Mapping Algorithms," vol. 6, no. 4, pp. 1–8, 2018.

[2] F. Liang and X. Wan, "Job Matching Analysis Based on Text Mining and Multicriteria Decision-Making," Math Probl Eng, vol. 2022, 2022, doi: 10.1155/2022/9245876.

[3] I. V Mashechkin, M. I. Petrovskiy, D. S. Popov, and D. V Tsarev, "Automatic Text Summarization Using Latent Semantic Analysis," vol. 37, no. 6, pp. 299–305, 2011, doi: 10.1134/S0361768811060041.

[4] H. Jayadianti and R. Damayanti, "Latent Semantic Analysis (LSA) Dan Automatic Text Summarization (ATS) Dalam Optimasi Pencarian Artikel Covid," vol. 2020, no. Semnasif, pp. 52–59, 2020.

[5] R. C. Belwal, S. Rai, and A. Gupta, "Text summarization using topic-based vector space model and semantic measure," Inf Process Manag, vol. 58, no. 3, p. 102536, 2021, doi: 10.1016/j.ipm.2021.102536.

[6] F. Al-Anzi and D. Abuzeina, "Enhanced latent semantic indexing using cosine similarity measures for medical application," International Arab Journal of Information Technology, vol. 17, no. 5, 2020, doi: 10.34028/iajit/17/5/7.

[7] S. Singla and A. Eldawy, "Raptor Zonal Statistics: Fully Distributed Zonal Statistics of Big Raster + Vector Data," Proceedings - 2020 IEEE International Conference on Big Data, Big Data 2020, pp. 571–580, 2020, doi: 10.1109/BigData50022.2020.9377907.

[8] Y. Kino, H. Kuroki, T. Machida, N. Furuya, and K. Takano, "Text Analysis for Job Matching Quality Improvement," Procedia Comput Sci, vol. 112, pp. 1523–1530, 2017, doi: 10.1016/j.procs.2017.08.054.

[9] W. Kopp, A. Akalin, and U. Ohler, "Simultaneous dimensionality reduction and integration for single-cell ATAC-seq data using deep learning," Nat Mach Intell, vol. 4, no. 2, pp. 162–168, 2022, doi: 10.1038/s42256-022-00443-1.

[10] K. Kowsari, K. J. Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown, "Text classification algorithms: A survey," Information (Switzerland), vol. 10, no. 4, pp. 1–68, 2019, doi: 10.3390/info10040150.

[11] R. M. Suleman and I. Korkontzelos, "Extending latent semantic analysis to manage its syntactic blindness," Expert Syst Appl, vol. 165, Mar. 2021, doi: 10.1016/j.eswa.2020.114130.

[12] S. R. Vrana, D. T. Vrana, L. A. Penner, S. Eggly, R. B. Slatcher, and N. Hagiwara, "Latent Semantic Analysis: A new measure of patient-physician communication," Soc Sci Med, vol. 198, no. December 2017, pp. 22–26, 2018, doi: 10.1016/j.socscimed.2017.12.021.

[13] A. Kontostathis, "Essential dimensions of latent semantic indexing (LSI)," Proceedings of the Annual Hawaii International Conference on System Sciences, no. March, 2007, doi: 10.1109/HICSS.2007.213.

[14] L. Canete-Sifuentes, R. Monroy, and M. A. Medina-Perez, "A Review and Experimental Comparison of Multivariate Decision Trees," IEEE Access, vol. 9, pp. 110451–110479, 2021, doi: 10.1109/ACCESS.2021.3102239.

[15] T. Bi, P. Liang, A. Tang, and C. Yang, "A systematic mapping study on text analysis techniques in software architecture," Journal of Systems and Software, vol. 144, no. January, pp. 533–558, 2018, doi: 10.1016/j.jss.2018.07.055.

[16] M. S. Eldin et al., "Alterations in Inflammatory Markers and Cognitive Ability after Treatment of Pediatric Obstructive Sleep Apnea," Medicina (Lithuania), vol. 59, no. 2, 2023, doi: 10.3390/medicina59020204.

[17] S. Jung, J. Hyung Cho, and I.-W. Kim, "Corporations' and Job Seekers' Using Intention and WOM (Word-of-Mouth) of NCS-based Job Matching System," Adv Econ Bus, vol. 7, no. 5, pp. 194–201, 2019, doi: 10.13189/aeb.2019.070503.

[18] A. Barducci, S. Iannaccone, V. La Gatta, V. Moscato, G. Sperlì, and S. Zavota, "An end-to-end framework for information extraction from Italian resumes," Expert Syst Appl, vol. 210, no. October 2021, p. 118487, 2022, doi: 10.1016/j.eswa.2022.118487.

[19] F. S. Al-Anzi and D. AbuZeina, "Toward an enhanced Arabic text classification using cosine similarity and Latent Semantic Indexing," Journal of King Saud University - Computer and Information Sciences, vol. 29, no. 2, pp. 189–195, Apr. 2017, doi: 10.1016/j.jksuci.2016.04.001.

[20] N. Aqilah, P. Rostam, N. Hashimah, and A. Hassain, "Text categorisation in Quran and Hadith : Overcoming the interrelation challenges using machine learning and term weighting," Journal of King Saud University - Computer and Information Sciences, vol. 33, no. 6, pp. 658–667, 2021, doi: 10.1016/j.jksuci.2019.03.007.

[21] and L. P. Xiaowei Wang;Zhenhong Jiang, "A Deep-Learning-Inspired Person-Job Matching Model Based on Sentence Vectors and Subject-Term Graphs," Complexity, vol. 2021, 2021, doi: 10.1155/2021/6206288.

[22] D. R. Ghica and K. Alyahya, "Latent semantic analysis of game models using LSTM," Journal of Logical and Algebraic Methods in Programming, vol. 106, pp. 39–54, 2019, doi: 10.1016/j.jlamp.2019.04.003.

[23] Z. Wang, W. Wei, C. Xu, J. Xu, and X. L. Mao, "Person-job fit estimation from candidate profile and related recruitment history with co-attention neural networks," Neurocomputing, vol. 501, pp. 14–24, 2022, doi: 10.1016/j.neucom.2022.06.012.

[24] P. Donner, "Identifying constitutive articles of cumulative dissertation theses by bilingual text similarity. Evaluation of similarity methods on a new short text task," Quantitative Science Studies, vol. 2, no. 3, 2021, doi: 10.1162/qss_a_00152.

[25] S. Zhao, Y. Wang, Z. Yang, and D. Cai, "Region mutual information loss for semantic segmentation," Adv Neural Inf Process Syst, vol. 32, no. 1, pp. 1–11, 2019.

[26] W. V. Padula et al., "Machine Learning Methods in Health Economics and Outcomes Research—The PALISADE Checklist: A Good Practices Report of an ISPOR Task Force," Value in Health, vol. 25, no. 7, pp. 1063–1080, 2022, doi: 10.1016/j.jval.2022.03.022.

[27] R. M. Suleman and I. Korkontzelos, "Extending latent semantic analysis to manage its syntactic blindness," Expert Syst Appl, vol. 165, no. October 2020, p. 114130, 2021, doi: 10.1016/j.eswa.2020.114130.

[28] M. Mimura and T. Ohminami, "Using lsi to detect unknown malicious vba macros," Journal of Information Processing, vol. 28, pp. 493–501, 2020, doi: 10.2197/ipsjjip.28.493.

[29] T. Denk and S. Lehtinen, "Contextual analyses with QCA-methods," Qual Quant, vol. 48, no. 6, pp. 3475–3487, Oct. 2014, doi: 10.1007/s11135-013-9968-4.

[30] A. Solomon, B. Shapira, and L. Rokach, "Predicting application usage based on latent contextual information," Comput Commun, vol. 192, pp. 197–209, Aug. 2022, doi: 10.1016/j.comcom.2022.06.005.

[31] L. LaLonde, J. Good, E. Orkopoulou, M. Vriesman, and A. Maragakis, "Tracing the missteps of stepped care: Improving the implementation of stepped care through contextual behavioral science," J Contextual Behav Sci, vol. 23, pp. 109–116, Jan. 2022, doi: 10.1016/j.jcbs.2022.01.001.

[32] S. Kim, H. Park, and J. Lee, "Word2vec-based latent semantic analysis (W2V-LSA) for topic modeling: A study on blockchain technology trend analysis," Expert Syst Appl, vol. 152, Aug. 2020, doi: 10.1016/j.eswa.2020.113401.

[33] J. S. Mendez and J. D. Bulanadi, "Job matcher: A web application job placement using collaborative filtering recommender system," International Journal of Research Studies in Education, vol. 9, no. 2, pp. 103–120, 2020, doi: 10.5861/ijrse.2020.5810.

[34] S. Wulandari and M. Rahmah, "A Survey on Crowdsourcing Awareness in Indonesia Micro Small Medium Enterprises," IOP Conf Ser Mater Sci Eng, vol. 769, no. 1, 2020, doi: 10.1088/1757-899X/769/1/012016.

[35] E. Ma, J. Du, S. (Tracy) Xu, Y. C. Wang, and X. Lin, "When proactive employees meet the autonomy of work—A moderated mediation model based on agency theory and job characteristics theory," Int J Hosp Manag, vol. 107, no. June, p. 103326, 2022, doi: 10.1016/j.ijhm.2022.103326.

[36] W. Wang, K. Zhang, H. Ren, D. Wei, Y. Gao, and J. Liu, "UULPN: An ultra-lightweight network for human pose estimation based on unbiased data processing," Neurocomputing, vol. 480, pp. 220–233, 2022, doi: 10.1016/j.neucom.2021.12.083.

[37] B. Zhao and H. Bilen, "Dataset Condensation with Distribution Matching." [Online]. Available: https://github.com/

[38] H. Nazif, "An effective meta-heuristic algorithm to minimize makespan in job shop scheduling," Industrial Engineering and Management Systems, vol. 18, no. 3, pp. 360–368, 2019, doi: 10.7232/iems.2019.18.3.360.

[39] J. Dhameliya and N. Desai, "Job Recommendation System using Content and Collaborative Filtering based Techniques," International Journal of Soft Computing and Engineering, vol. 9, no. 3, pp. 8–13, 2019, doi: 10.35940/ijsce.c3266.099319.

[40] Md. S. Hossain and M. Shamsul Arefin, "Development of an Intelligent Job Recommender System for Freelancers using Client's Feedback Classification and Association Rule Mining Techniques," Journal of Software, vol. 14, no. 7, pp. 312–339, 2019, doi: 10.17706/jsw.14.7.312-339.

[41] A. Sharma and S. Kumar, "Ontology-based semantic retrieval of documents using Word2vec model," Data Knowl Eng, vol. 144, Mar. 2023, doi: 10.1016/j.datak.2022.102110.

[42] Mofiz Mojib Haider, Automatic Text Summarization Using Gensim Word2Vec and K Means Clustering Algorithm. 2020.