Exploring the Efficiency of Text-Similarity Measures in Automated Resume Screening for Recruitment

Ahmad Alsharef
Yogananda School of AI, Computer & Data Science
Shoolini University
Solan, H.P, India
ahmadalsharef@shooliniuniversity.com

Sonia
Yogananda School of AI, Computer & Data Science
Shoolini University
Solan, H.P, India
soniacsit@yahoo.com

Hasan Nassour
School of Management Sciences and Liberal Arts
Shoolini University
Solan, H.P, India
hasannassour@shooliniuniversity.com

***Abstract*—With many online recruitment portals requesting job applicants to upload their resumes, the automated process of screening and shortlisting candidates can accelerate selection and decision-making. This study explores the use of text similarity measures as an alternative to experienced human hiring managers in processing resumes. Three text similarity measures: Cosine, Sqrt-Cosine, and Improved Sqrt-Cosine (ISC) similarity were utilized as computer programs in scanning resumes for the recruitment of a business development manager and a software engineer. The decisions of the algorithms were compared to those of an expert hiring manager within the same scenarios. The findings indicate that ISC and Sqrt-Cosine were closer to the expert-human decision than Cosine similarity. These specific text-similarity algorithms can also make acceptable decisions even when recruiting for high-level positions and can do so in seconds when executed as a program on a normal CPU processor. This study suggests that these algorithms can efficiently facilitate the process of decision-making in recruitment and shortlisting candidates and can be an effective alternative to expert hiring managers.**

***Keywords*— Natural Language Processing; Similarity measures; Cosine similarity; Sqrt-Cos similarity; ISC similarity; Resume-recommendation; Decision-making.**

I. Introduction

The recruitment process is a time-consuming and costly task for many organizations, and screening resumes is one of the most tedious tasks involved [1]. This has led to an increase in the use of automated screening tools, such as applicant tracking systems (ATS) [2] and text similarity measures [3]. However, the extent to which these tools can be an effective alternative to human experts in the recruitment process remains a topic of debate.

Many of these ATSs utilize text-document similarity measurements like Cosine similarity to compare the text content of resumes with the text content of job descriptions. Cosine similarity didn't prove itself as the best algorithm to calculate the similarity between texts [4]. For this reason, this work explores more advanced algorithms.

This study, aimed to compare the decisions of a hiring manager in ranking resumes based on fitness with machine-generated decisions using specific text-document similarity calculation methods. The objective was to provide highly accurate methods to compare text blocks and find similarities in order to solve the problem of automated recruiting. We sought to verify the potential of several text-similarity measures in sorting candidates based on their fitness to a job description, as presented in their resumes.

To achieve these goals, we compared the accuracy and performance of three text-similarity measures: Cosine, Sqrt-Cosine, and ISC, with the human-level performance (HLP) as a benchmark. Additionally, we compared the execution time of these measures when implemented as computer programs and executed on the same CPU processor to explore their performance efficiency.

Based on the Euclidean norm, the Cosine similarity measure [5] is currently the most widely used method in calculating the similarity between text documents. This measure and other similarity measures like Sqrt-Cosine and ISC similarity were implemented in this work and were found, according to the experimental results, indeed effective. This work added to the growing body of automated online recruiting by exploring the efficiency of new methods for locating best-fit resumes using modern text similarities techniques like Sqrt-Cosine and ISC similarity.

This work used real datasets based on real scenarios, collected from Human Resource (HR) recruiting managers of two companies, one in Europe and one in the U.S., after taking permission from the candidates. The HR managers were asked to provide a survey about the candidates and their fitness for jobs based on their resumes. These surveys were taken as a benchmark to compare the efficiency of different methods since this work strives to reach the HLP in finding the best candidates for a job.

II. Literature Review

The following paragraphs describe the different metrics of similarity between two blocks of text. It will also mention the most relevant works to this work.

A. Text Similarity Measures

Text similarity measures are functions that output a real number between 0 and 1 to a two-document input. The higher similarity between the two documents, the higher the value of the output. Text similarity measures gained significance in text-related research work and applications such as text classification, information retrieval, document clustering, machine translation, essay scoring, text summarization, and others [6]. In the following paragraphs, a discussion on three similarity functions including Cosine, Sqrt-Cosine, and ISC similarity, is provided.

1) Cosine similarity:

Cosine similarity is a significant and widely common approach to calculating the similarity between text documents irrespective of their sizes. Calculating cosine similarity between two text documents, x, and y, requires vectorizing them (converting each document into a vector of real numbers) and then normalizing these vectors into L2 norm (Euclidean norm), which is a positive value calculated as the square root of the sum of the squared vector values: $\|\mathbf{x}\| = \sqrt{\sum_{i=1}^{m} \mathbf{x_i}^2}$. After getting the normalized vector representations of the two text documents, x, and y, the Cosine similarity between them will be simply the dot product of them, as it is given in Equation (1) [7].

$$cos(x,y) = \frac{x \cdot y}{\|x\| \times \|y\|} = \frac{\sum_{i=1}^{m} x_i y_i}{\sqrt{\sum_{i=1}^{m} {x_i}^2}\sqrt{\sum_{i=1}^{m} {y_i}^2}} \qquad (1)$$

Where: $\|x\|$ *and* $\|y\|$ are the Euclidean norms of vectors $x(x_1, x_2, x_3 \ldots . x_m)$ and $y(y_1, y_2, y_3 \ldots . y_m)$, respectively, which, in turn, are defined as $\|\mathbf{x}\| = \sqrt{\sum_{\mathrm{i}=1}^{\mathrm{m}} {\mathrm{x_i}}^2} = \sqrt{{x_1}^2 + {x_2}^2 + .. + {x_m}^2}$ and $\|\mathbf{y}\| = \sqrt{\sum_{\mathrm{i}=1}^{\mathrm{m}} {\mathrm{y_i}}^2} = \sqrt{{y_1}^2 + {y_2}^2 + \ldots . + {y_m}^2}$, respectively.

The former cosine similarity equation is derived from the Euclidean distance, which is shown in Equation (2).

$$Dist_{Euclid}(x,y) = \sqrt[2]{\sum_i (x_i - y_i)^2} = \sqrt[2]{2 - 2\sum_i x_i y_i} \qquad (2)$$

However, information retrieved from high-dimensional data like resume documents that have many words might become problematic when using Euclidean distance which is seldom considered an efficacious distance measure in this case [4]. Aggarwal et al. [4] proved that the Euclidean (L2) norm is often inefficient for high-dimensional data. They explored the behavior of the different Lk norms in high-dimensional space and concluded that it is recommended to use a lower value of k when dealing with high-dimensional data. More specifically, L1 distance is more convenient than L2 (Euclidean) distance in high-dimensional data like resume documents which might have too many words. Moreover, this work proves that the ISC similarity that uses the L1 norm is more efficient.

Another efficient similarity measure that also uses the L2 norm like Cosine similarity was later proposed by Zhu et al. [8] and was called "Sqrt-Cosine similarity".

2) Sqrt-Cosine similarity:

Zhu et al. [8] 2012 suggested this similarity calculation approach which is based on the Hellinger distance. The formula of Sqrt-Cosine similarity is represented in Equation (3).

$$SqrtCos(x,y) = \frac{\sum_{i=1}^{m} \sqrt{x_i y_i}}{(\sum_{i=1}^{m} x_i)(\sum_{i=1}^{m} y_i)} \qquad (3)$$

The Sqrt-Cosine similarity was derived from the Hellinger distance represented in Equation (4).

$$Dist_{Hell}(x,y) = \sqrt{\sum_i (\sqrt{x_i} - \sqrt{y_i})^2} = \sqrt{2 - 2\sum_i \sqrt{x_i y_i}} \quad (4)$$

As Zhu et al. [1] believed that Euclidean distance is not a good metric and Sqrt-Cos similarity is more appropriate in judging query relevance and fitter for dealing with the word-document connotations, they conducted two sets of experiments to evaluate the effectiveness of the Sqrt-Cosine metric in solving information retrieval problems. In the first set, they compared the document clustering results, and in the second, conducted a query on a real IR system and compared the output with the output of the standard cosine similarity. Through these experiments, they concluded that the precision and recall were improved by using Sqrt-Cos similarity. For this reason, this work used the Sqrt-Cosine similarity to locate the most relevant candidates among a pool of resumes.

However, Sohangir et al. [9], in 2017, proved that Sqrt-Cosine similarity is not always a reliable similarity measure. They found, by experiments, that Sqrt-Cosine similarity between two equal documents input did not equal one, revealing some defects in the design, and a particular document was found to be more similar to another document than to itself. To address these problems, they proposed an improved similarity measure based on Sqrt-Cosine similarity, and they called it the "Improved Sqrt-Cosine (ISC) similarity".

3) ISC similarity:

Sohangir et al. [9] proposed the ISC similarity metric represented in Equation (5).

$$ISC(x,y) = \frac{\sum_{i=1}^{m} \sqrt{x_i y_i}}{\sqrt{(\sum_{i=1}^{m} x_i)}\sqrt{(\sum_{i=1}^{m} y_i)}} \qquad (5)$$

In this equation, the square root of the L1 norm was used. L1 norm (Manhattan norm) is calculated as the sum of the absolute vector values: $\|x\| = \sum_{i=2}^{m} |x_i|$.

Sohangir et al. [9] also conducted experiments to compare ISC similarity with other similarity measures in different application domains, including document classification, document clustering, and information retrieval. In all cases, ISC similarity outperformed Cosine similarity and other measurements. For this reason, this work selected the ISC similarity and implemented it expecting an acceptable efficiency. It was found accurate and effective.

B. Related Work

Amin et al. [10] designed a web application where resumes are compared with the company recruiter's job profile requirements using semi-supervised learning. Indira et al. [6] proposed a model that provided a list of applicants with appropriate experiences depending on their resumes using Natural Language Processing (NLP) to extract features from data and convert it to a structured format with the required extracted features. Craven et al. [11] proposed a system that extracts information from resumes using XML tags to recognize key attributes including email, name, address, ..etc. Jiang et al. [12] explored extracting information from Chinese resumes by using regular expressions and fuzzy logic. Saxena et al. [13] proposed a model that uses keyword matching and normalization to map job requirements with candidates. Lu et al. [14] surveyed different protocols used in resume recommendation systems and discussed how resume-recommendation systems are commonly used in real-time applications. Wei et al. [15] discussed the types of resume recommendation methods with their working base. Al-Taibbi et al. [16] surveyed job recommendation systems and the steps involved in the process of online recruiting. Golem & Kahya [17] proposed a fuzzy-based model to evaluate applicants to a posted job description. Roy et al. [18] proposed a content-based resume recommendation system that used cosine similarity and k-nearest neighbors algorithm (k-NN) to locate the CVs nearest to the provided job description.

Recent studies have shown that the use of text similarity measures, such as word embeddings and cosine similarity, can provide an efficient and reliable method for screening resumes. For instance, Chen et al. [19] proposed a robust and adaptive text mining system that utilizes semantic similarity and classification algorithms for resume screening. Meanwhile, Tran et al. [20] evaluated the performance of different word embeddings and text similarity metrics for automatic resume screening. Furthermore, Mishra and Misra [21] developed an intelligent resume screening system using word embeddings and machine learning techniques.

This work differs from the growing body of proposed systems, as it uses HLP as a benchmark. It compares the ranking of candidates generated by automated methods based on the content of job descriptions and resumes with the expert human ranking of the same candidates corresponding to the same job description. In other words, this work took the HLP, which has been a goal of academic research in machine learning, as a benchmark. Also, it went beyond the Cosine similarity state-of-art algorithm for similarity and explored the efficiency of using Sqrt-Cosine and ISC similarity measures. The suggested algorithms are characterized by simplicity and high performance and accuracy.

III. Materials and Methodology

The experiments used the following materials and concepts:

- ATS: ATS [2] is an automated system used by organizations to manage the recruitment process. The ATS system was used in this study to help automate the process of screening resumes and finding the best-fit candidates.

- Text similarity measures: Text similarity measures [3] are algorithms used to measure the similarity between two text documents. Several measures were implemented in this work and were found, according to the experimental results, to be effective in sorting candidates based on their fitness to a job description.

- Python programming language: Python [22] is a widely used high-level programming language. It was used in this study to implement the text similarity measures and algorithms used to sort the candidate resumes.

- Scikit-learn library: Scikit-learn [23] is an open-source machine-learning library for Python. It was used in this study to implement the Cosine, Sqrt-Cosine, and Improved Sqrt-Cosine text similarity measures.

- NLTK: NLTK [24] stands for Natural Language Toolkit, which is a library in Python that provides tools and resources for NLP. In the work, NLTK was used to preprocess the text data of resumes and job descriptions. This involved various tasks such as tokenization, stopword removal, stemming, and part-of-speech tagging. After preprocessing the data with NLTK, text-similarity measures such as Cosine, Sqrt-Cosine, and Improved Sqrt-Cosine were applied to calculate the similarity between resumes and job descriptions.

- Pandas library: Pandas [25] is a Python library used for data manipulation and analysis. It was used in this study to read and process the candidate resumes and job descriptions.

The methodology used in this study involves the following steps:

1. Collection of real datasets based on real scenarios from two companies, one in Europe and one in the U.S., after taking permission from the candidates.
2. Preparation of the datasets for analysis.
3. Implementation of Cosine, Sqrt-Cosine, and ISC similarity measures.
4. Comparison of the accuracy and performance of these measures in terms of accuracy and execution time.
5. Comparison of the results with the HLP as a benchmark.
6. Analysis of the results to determine the efficiency of the different methods.

This work has explored Cosine, Sqrt-Cosine, and ISC similarity measures in two case studies:

- First, match the job description of the position of a “business development manager" with the resumes of 40 applicants with 10-20 years of working experience.
- Second, matching the job description of the position of a "software engineer" with the resumes of 30 applicants with 5-10 years of working experience.

These candidates (40 in the first case and 30 in the second) were in both experiments real applicants to the positions. For this reason, their resumes were selected as samples aiming to explore the efficiency similarity techniques in a real-life scenario.

This work utilized datasets of real case studies collected from two companies, after taking authorization from the candidates on using their resumes in our experiments. The HR managers of the two companies were requested to provide a survey containing the ranking of the candidates based on their fitness for the job. These two surveys were taken as a benchmark in the experimental work where the ranking provided by the automated methods was compared to compare the ranking of the HR managers in each experiment.

The methodology followed in this experimental work is illustrated in figure 1.

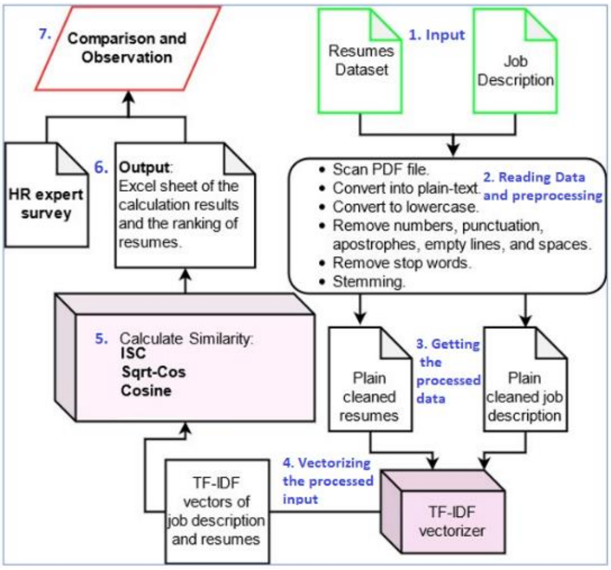
7. Comparison and Observation
1. Input
Resumes Dataset
Job Description
HR expert survey
6. Output: Excel sheet of the calculation results and the ranking of resumes.
• Scan PDF file.
• Convert into plain-text.
• Convert to lowercase.
• Remove numbers, punctuation, apostrophes, empty lines, and spaces.
• Remove stop words.
• Stemming.
2. Reading Data and preprocessing
5. Calculate Similarity: ISC Sqrt-Cos Cosine
Plain cleaned resumes
3. Getting the processed data
Plain cleaned job description
4. Vectorizing the processed input
TF-IDF vectors of job description and resumes
TF-IDF vectorizer

Fig 1. The methodology used in the experimental work

We, first, collected a human ranking survey on the candidates' eligibility, conducted by a hiring manager. Then, compare this survey with the ranking generated by ISC, Sqrt-Cosine, and Cosine similarity measures to figure out the efficiency of these methods in comparing job descriptions with resumes. As mentioned earlier, the experiments were on real-life recruitment case studies of recruiting a business development manager and recruiting a software engineer.

The following paragraphs illustrate the steps followed in the experiments to get the results.

A. *Experiment I*

1) *Data collection:*

The dataset contained 40 resumes of applicants for a role with a job description of a business development manager of one of the leading platforms for SME Compliance Management in Europe.

2) *Data conversion and cleaning:*

With the use of NLTK library [22] which provides methods for tokenization, stemming, and text cleaning and normalization methods, Scikit-learn library [23] which provides methods for converting texts to vectors and calculating the similarity between texts, and Python programming language, the resumes, and job descriptions were converted into plain text and cleaned as the following:

1. The PDF files of job descriptions and resumes were converted into plain-text bodies.
2. Text data cleaning included:
   - Removing empty lines from documents.
   - Removing stop words from sentences. Stop words are words that are most common in the language and don't affect the whole meaning like "the", "a", "is", etc.
   - Removing numbers and special characters
   - Converting the whole text into lowercase letters.
   - Removing punctuations and apostrophes.

3) *Stemming:*

Stemming is the normalization task concerned with removing word affixes (prefixes and suffixes) from each word. For example, stemming would trunk the word "going" to "go". Stemming was conducted to enhance the matching reliability between words of different documents. For example, the words "developer" and "development" were considered identical after stemming each of them into "develop".

4) *Vectorization:*

Since similarity measures calculate the similarity between vectors, vectorization is needed to convert the bodies of texts (resumes in our case) into a vector, and this is called "vectorization". One common technique of vectorization is "count vectorizer" which allows frequent words to dominate. Frequent words are the words that occur multiple times in a document and across the majority of documents. The count vectorizer concludes that documents containing these frequent words are more similar to each other as it overweighs the frequent words, although these words may sometimes be insignificant in the context. On the other hand, the term frequency-inverse document frequency (TF-IDF) vectorizer, places more emphasis on rare words, words that aren't frequently included across the majority of resumes but only included in a few documents. TF-IDF vectorizer concludes that these documents are more similar to each other.

This work used the TF-IDF vectorization method rather than the count vectorization which is a widely used method since TF-IDF outweighs the less frequent words and doesn't only focus on the frequency of words allowing less frequent words like skills and keywords to dominate in matching text documents.

IV. Experiments and Results

The experiment process involved collecting real datasets from two companies, one in Europe and one in the US, for the roles of business development manager and software engineer. The datasets were then prepared for analysis by following the same steps of data conversion and cleaning, stemming, and vectorization. The performance of Cosine, Sqrt-Cosine, and ISC similarity measures was compared with the HLP as a benchmark within two metrics: execution time and accuracy.

This work compared the potential of the similarity measures within two metrics: execution time and accuracy.

a) *Execution time:*

The total time needed for calculating the similarity between each of the 40 resumes and the job description in the first experiment (role of business development manager on the processor " Intel® Core™ i5-7200U CPU @ 2.50GHz (4 CPUs), ~2.7GHz", in seconds, is given in table 1:

TABLE 1. TIME NEEDED FOR CALCULATING SIMILARITY MEASURES OF 40 RESUMES IN EXPERIMENT 1.

Similarity measure	Cosine	Sqrt-Cosine	ISC

Execution time (in seconds)	0.45	*0.32*	0.36

On average, the Cosine similarity calculation for each resume took 0.011 seconds (0.45 seconds for the whole 40-resume sample).

On average, the Sqrt-Cosine similarity calculation for each resume took 0.008 seconds (0.32 seconds for the whole 40-resume sample).

On average, the ISC similarity calculation for each resume took 0.009 seconds (0.36 seconds for the whole 40-resume sample).

In terms of execution time, ISC similarity outperformed Cosine similarity, while Sqrt-Cosine similarity outperformed both ISC and Cosine similarities

b) Accuracy:

By consulting an expert hiring manager to rank the resumes without looking at the ranking generated by the automated similarity functions. The similarity score resulting from using the similarity measures was noted, the higher score means a higher similarity. We noted the ranking generated by the expert hiring manager and the ranking generated by the automated methods and then observed the results. It was found that the decision taken by the automated functions was close to the decision taken by the expert human. The following table 2 depicts a sample of the results obtained in the first experiment:

TABLE 2. COMPARISON OF THE RANKING OF RESUMES BETWEEN AN EXPERT HUMAN AND AUTOMATED MEASURES (ROLE OF BUSINESS DEVELOPMENT MANAGER).

Human Ranking of resumes	**Automated similarity measurement ranking and similarity score**		
	ISC	***Sqrt-Cosine***	***Cosine***
1	2 (0.38)	11 (28.545)	7 (43.02)
2	3 (0.345)	2 (32.732)	1 (48.44)
3	1 (0.391)	9 (28.87)	4 (44.29)
4	4 (0.344)	17 (26.749)	14 (38.5)
5	8 (0.302)	1 (32.733)	15 (38.2)
6	5 (0.34)	13 (27.744)	8 (41.67)
7	11 (0.296)	16 (27.023)	17 (37.55)
8	7 (0.303)	3 (30.537)	18 (37.35)
9	6 (0.314)	7 (29.147)	11 (39.38)
10	10 (0.29)	4 (30.53)	3 (44.56)
11	9 (0.301)	34 (22.973)	26 (35.25)
12	16 (0.28)	18 (26.478)	16 (38.08)
……..	……..	……..	……..
……..	……..	……..	……..
28	34 (0.248)	35 (22.827)	33 (32.78)
29	28 (0.256)	21 (25.985)	32 (32.97)
30	39 (0.226)	29 (24.47)	39 (26.29)
31	27 (0.253)	24 (25.206)	29 (34.19)
32	30 (0.253)	36 (22.554)	36 (31.65)
33	32 (0.245)	22 (25.836)	23 (36.3)
34	31 (0.245)	27 (24.845)	21 (36.73)
35	35 (0.244)	38 (21.384)	38 (28.88)
36	29 (0.25)	33 (23.747)	34 (32.54)
37	38 (0.225)	32 (24.113)	25 (35.36)
38	40 (0.223)	40 (19.277)	40 (22.81)
39	26 (0.25)	23 (25.396)	28 (34.83)
40	33 (0.243)	30 (24.214)	31 (33.86)

We observed that:

The **top 10** candidates, as reported by the **expert hiring manager**, were ranked among the **top 11** when using **ISC similarity**, among the **top 17** when using **Sqrt-Cos similarity**, and among the **top 18** when using **Cosine similarity**.

The **bottom 10** candidates, as reported by the **expert hiring manager**, were ranked among the **bottom 14** when using **ISC similarity**, among the **bottom 19** when using **Sqrt-Cos similarity**, and among the **bottom 20** when using **Cosine similarity**.

From the former observations, we concluded that ISC similarity outperforms, in terms of accuracy, the other two methods. This is because the L1 norm is proven to be a better fit in high-dimensional data compared to the L2 norm.

B. *Experiment II*

1) *Data collection:*

The dataset contained 30 resumes for applicants to the role job description of software engineer at an American multinational technology company.

2) *Data preprocessing:*

Data conversion and cleaning, stemming, and vectorization followed the same steps as the first experiment.

3) *Exploring the potential of similarity measures:*

a) Execution time:

The total time needed for calculating the similarity between each of the 30 resumes and the job description in the first experiment (role of business development manager on the processor " Intel® Core™ i5-7200U CPU @ 2.50GHz (4 CPUs), ~2.7GHz", in seconds, is given in table 3:

TABLE 3. TIME NEEDED FOR CALCULATING SIMILARITY MEASURES OF 30 RESUMES IN EXPERIMENT 1

Similarity measure	Cosine	Sqrt-Cosine	ISC
Execution time (in seconds)	0.36	*0.30*	0.33

On average, the Cosine similarity calculation for each resume took 0.012 seconds (0.36 seconds for the whole 30-resume sample).
On average, the Sqrt-Cos similarity calculation for each resume took 0.010 seconds (0.30 seconds for the whole 30-resume sample).
On average, the ISC similarity calculation for each resume took 0.011 seconds (0.33 seconds for the whole 30-resume sample).

According to the execution time, ISC similarity outperformed Cosine similarity. Whereas, and Sqrt-Cosine similarity outperformed the other two approaches.

b) Accuracy:

By consulting an expert hiring manager to rank the resumes without looking at the ranking generated by the automated similarity functions. The similarity score resulting from using the similarity measures was noted, the higher score means a higher similarity. We noted the ranking generated by the expert hiring manager and the ranking generated by the automated methods and then observed the results. It was found that the decision taken by the automated functions was close to the decision taken by the expert human. The following table 4 depicts a sample of the results obtained in the second experiment:

TABLE 4. COMPARISON OF THE RANKING OF RESUMES BETWEEN AN EXPERT HUMAN AND AUTOMATED MEASURES (ROLE OF SOFTWARE ENGINEER).

Human Ranking	**Automated similarity measurement Ranking and Score**		
	ISC	***Sqrt-Cos***	***Cosine***
1	8 (0.181)	6 (21.205)	11 (18.47)
2	1 (0.246)	4 (22.136)	1 (26.99)
3	4 (0.199)	1 (22.744)	4 (22.25)
4	3 (0.206)	2 (22.65)	10 (18.52)
5	6 (0.181)	14 (18.708)	3 (24.36)
6	10 (0.163)	5 (21.397)	6 (19.98)
7	13 (0.156)	12 (19.15)	8 (18.75)
8	9 (0.169)	7 (20.576)	13 (16.56)
9	2 (0.224)	8 (20.172)	2 (24.81)
10	5 (0.184)	10 (19.7)	14 (16.44)
11	7 (0.181)	11 (19.286)	9 (18.61)
12	12 (0.161)	16 (16.907)	5 (20.74)
....	....	....	....
....	....	....	....
18	17 (0.125)	17 (16.044)	22 (10.6)
19	19 (0.121)	13 (18.954)	20 (11.55)
20	20 (0.118)	15 (18.692)	17 (13.54)
21	27 (0.097)	25 (14.604)	21 (11.17)
22	21 (0.117)	21 (15.762)	24 (9.99)
23	29 (0.079)	26 (14.362)	30 (8.05)
24	22 (0.112)	29 (13.611)	25 (9.74)
25	26 (0.103)	24 (14.901)	19 (12.05)
26	25 (0.108)	19 (15.912)	26 (9.64)
27	28 (0.091)	28 (13.931)	28 (9.19)
28	30 (0.078)	27 (14.361)	29 (8.06)
29	24 (0.111)	22 (15.077)	27 (9.36)
30	23 (0.111)	30 (12.966)	18 (12.45)

We observed that:

The **Top 10** candidates, as reported by the **expert hiring manager**, were ranked among the **top 10** when using **ISC similarity**, among the **top 14** when using **Sqrt-Cos similarity**, and among the **top 14** when using **Cosine similarity**.
The **Bottom 10** candidates, as reported by the **hiring manager expert**, were ranked among the **bottom 10** when using **ISC similarity**, among the **bottom 12** when using **Sqrt-Cos similarity**, and among the **bottom 13** when using **Cosine similarity**.

From the former observations, we concluded that ISC similarity outperforms, in terms of accuracy, the other two methods. This is because the L1 norm is proven to be a better fit in high-dimensional data compared to the L2 norm.

The findings of this study are significant as they demonstrate the potential of automated recruiting using text similarity measures. This approach can help recruiters filter the most suitable candidates for a particular job and nominate the best.

V. CONCLUSION

In conclusion, this study aimed to evaluate the potential of specific text similarity methods in automated recruiting by comparing their performance with expert human decision-making. The ISC similarity measure was found to be more accurate than Cosine and Sqrt-Cos similarity measures and closely matched human decision-making. All three methods were efficient in ranking resumes within seconds. The results suggest that automated processes can be effective in finding the best-fit candidates for a particular job, helping recruiters filter suitable candidates and nominate the best. Future research can explore the potential of more text similarity methods and enhanced hybrid techniques to improve the effectiveness of automated recruiting.

REFERENCES

[1] Dittrich, J., Busch, P. A., & Rieder, K. (2021). The Effect of Social Capital on Job Search and Recruitment Processes: A Systematic Literature Review. International Journal of Environmental Research and Public Health, 18(5), 2527. https://doi.org/10.3390/ijerph18052527

[2] Bharadwaj, S., Varun, R., Aditya, P. S., Nikhil, M., & Babu, G. C. (2022, July 20-22). Resume Screening using NLP and LSTM. Paper presented at the 2022 International Conference on Inventive Computation Technologies (ICICT), pp. 238-241. IEEE.

[3] Fauzan, R., Labib, M. I. A., Johannis, J. O. T., Herlinawati, Saifulah, S. Noor. (2022, June 29-July 1). Semantic similarity of Indonesian sentences using natural language processing and cosine similarity. Paper presented at the 2022 4th International Conference on Cybernetics and Intelligent Systems (ICORIS), Prapat, Indonesia, pp. 1-4. doi: 10.1109/ICORIS56080.2022.10031439.

[4] Aggarwal, C. C., Hinneburg, A., & Keim, D. A. (2001). On the surprising behavior of distance metrics in high dimensional space. In International conference on database theory (pp. 420–434). Springer.

[5] Kirişci, M. (2023). New cosine similarity and distance measures for Fermatean fuzzy sets and TOPSIS approach. Knowledge and Information Systems, 65(4), 855–868. https://doi.org/10.1007/s10115-022-01776-4

[6] Indira, M. D., & Kumar, R. K. (2016). Profile screening and recommending using natural language processing (NLP) and leverage Hadoop framework for big data. International Journal of Computer Science and Information Security (IJCSIS), 14(6).

[7] Korenius, T., Laurikkala, J., & Juhola, M. (2007). On principal component analysis, cosine and Euclidean measures in information retrieval. Information Sciences, 177(22), 4893-4905.

[8] Zhu, S., Liu, L., & Wang, Y. (2012, July 29-August 1). Information retrieval using Hellinger distance and sqrt-cos similarity. Paper presented at the 2012 7th International Conference on Computer Science & Education (ICCSE), pp. 925–929. IEEE.

[9] Sohangir, S., & Wang, D. (2017). Improved sqrt-cosine similarity measurement. Journal of Big Data, 4(1), 1–13.

[10] Amin, S., Jayakar, N., Sunny, S., Babu, P., Kiruthika, M., & Gurjar, A. (2019, April 4-6). Web application for screening resume. Paper presented at the 2019 International Conference on Nascent Technologies in Engineering (ICNTE), pp. 1–7. IEEE.

[11] Ciravegna, F., & Lavelli, A. (2004). Learning Pinocchio: adaptive information extraction for real-world applications. Natural Language Engineering, 10(2), 145–165.

[12] Jiang, Z., Zhang, C., Xiao, B., and Lin, Z. (2009, January 8-11). Research and implementation of intelligent Chinese resume parsing. Paper presented at the 2009 WRI international conference on communications and mobile computing, vol. 3, pp. 588–593. IEEE.

[13] Saxena, C. (2011). Enhancing the productivity of recruitment process using data mining & text mining tools.

[14] Lu, J., Wu, D., Mao, M., Wang, W., & Zhang, G. (2015). Recommender system application developments: a survey. Decision Support Systems, 74, 12–32.

[15] Wei, K., Huang, J., & Fu, S. (2007, June 25-28). A survey of e-commerce recommender systems. Paper presented at the 2007 international conference on service systems and service management, pp. 1–5. IEEE.

[16] Al-Otaibi, S. T., & Ykhlef, M. (2012). A survey of job recommender systems. International Journal of Physical Sciences, 7(29), 5127–5142.

[17] Golec, A., & Kahya, E. (2007). A fuzzy model for competency-based employee evaluation and selection. Computers & Industrial Engineering, 52(1), 143–161.

[18] Roy, P. K., Chowdhary, S. S., & Bhatia, R. (2020, January 9-11). A Machine Learning Approach for automation of Resume Recommendation system. Paper presented at the Procedia Computer Science, vol. 167, pp. 2318-2327.

[19] Chen, S. S., Chen, H. C., & Lin, C. C. (2021, November 19-21). A Robust and Adaptive Text Mining System for Recruiting Process. Paper presented at the 2021 IEEE 16th International Conference on Intelligent Systems and Knowledge Engineering (ISKE), Chengdu, China, pp. 545-550. doi: 10.1109/ISKE52549.2021.00097.

[20] Tran, T. H., Tran, H. M., Nguyen, T. M., Nguyen, T. H., & Nguyen, V. T. (2021, January 7-10). Evaluating Word Embeddings and Text Similarity Metrics for Automatic Resume Screening. Paper presented at the 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, pp. 0533-0538. doi: 10.1109/CCWC51732.2021.9371274.

[21] Mishra, V., & Misra, A. (2021, March 18-19). Intelligent Resume Screening Using Word Embeddings and Machine Learning. Paper presented at the 2021 8th International Conference on Signal Processing and Integrated Networks (SPIN), Noida, India, pp. 311-315. doi: 10.1109/SPIN51788.2021.9400117.

[22] Van Rossum, G., & Drake Jr, F. L. (1995). Python tutorial (Vol. 620). Amsterdam, The Netherlands: Centrum voor Wiskunde en Informatica.

[23] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. the Journal of Machine Learning Research, 12, 2825-2830.

[24] Bird, S. (2006). NLTK: the natural language toolkit. Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions.

[25] W. McKinney et al., "Data structures for statistical computing in python," in Proceedings of the 9th Python in Science Conference, vol. 445, Austin, TX, 2010, pp. 51-56.