Vol. 06, No. 4 (2024) 1473-1480, doi: 10.24874/PES06.04.006

Proceedings on Engineering Sciences

www.pesjournal.net

AI-BASED RESUME SKILLS EXTRACTOR AND RECOMMENDER MODULES FOR STATE UNIVERSITY HUMAN RESOURCE ANALYTICS SYSTEM

Paul Nigel S. Abalos[1]
Thelma D. Palaoag

Received 04.06.2024.
Received in revised form 01.11.2024.
Accepted 10.11.2024.
UDC – 004.62

Keywords:

Recommender System, Skills Extraction, Job Recommendation, Apriori, Cosine similarity

A B S T R A C T

This quantitative-developmental research primordially seeks to develop two artificial intelligence (AI) agents, namely Skills Extractor and Job Recommender AI Agents, that will be involved in the skills extraction from a portable document format resumé (PDF) file and offer job recommendations via a larger web application-based data-driven state university human resource analytics system being concurrently developed also by the researchers. Dataset for this study which covers ___ resumes/applicants and ____ job posts in ___ job sectors came from the resumés and job posts from the largest state university in the Ilocandia Region of the Philippines. The researchers explored and compared Apriori data mining association rule algorithm and content-based filtering (CBF) approach to match extracted words and phrases from a resumé to skills banks and job posts and generate list of soft and hard skills using support, confidence, and lift metrics for the Apriori algorithm and cosine similarity score for the CBF algorithm, and from there generate job or applicant recommendations. For the validation, the researchers employed offline evaluation method by using relevancy approach through decision support metrics (accuracy, precision, recall, and F1-score), and ranking-based metrics (average precision or AP@k). Experimental results of the study have shown that the CBF algorithm has outperformed the Apriori algorithm which obtained mean accuracies of 92.30% for skills recommendation and 91.82% for job across ___ test job sectors. Meanwhile, significant mean accuracy differences of 4.28% in skills recommendations and 5.38% in job recommendations, respectively, were measured between the 2 algorithms in favor of the CBF algorithm.

1. INTRODUCTION

Skills extraction and job recommendations have been a critical part of the recruitment process for decades. However, according to Wang and Chen (2021), the advent of artificial intelligence (AI) has revolutionized the field, enabling more sophisticated and accurate methods for both tasks. In the early 1980s, skills extraction and job recommendations were primarily

[1] Corresponding author: Paul Nigel S. Abalos
Email: paulnigelabalos@gmail.com

manual processes. Human resource experts reviewed resumes, job postings, and other documents to identify relevant skills and match candidates to jobs (Jatoba, Santos, Guitierriz, Moscon, Fernandes, & Teixeira, 2019). This approach was time-consuming and error-prone, particularly for large organizations with a high volume of resumes and job postings.

In recent years, AI and data mining have been used to develop more sophisticated skills extraction and job recommendation systems that can learn from large datasets of text and data to identify skills and make recommendations which led to significant improvements in the accuracy and scalability of both tasks (Jatoba, et al., 2019; Nain & Shyam, 2024). AI-based skills extraction systems use a variety of techniques to identify skills in pure plaintexts from resumé files and among these techniques is to use natural language processing (NLP) to extract words and phrases and associate their relationship with established association rules or models to come up with recommendations of skillsets (Chen, Wang, Sun & Wang, 2020). Once skills have been extracted, AI-based systems can use a variety of techniques to rank and prioritize the skills. One common technique is to use the frequency of skills in a given dataset to determine their importance while another common technique is to use a machine learning algorithm to learn the importance and similarity of skills based on their relationship to job postings (He, Chua, & Guan, 2019). At this point, the purpose of AI has shifted from skills extraction to job recommendation.

In general, recommender systems commonly divide tasks into two main stages: candidate generation and scoring(Chen, et al., 2020) in which candidate generation is provision of a list relevant to the user's input in relation to a dataset of items and corresponding importance while scoring is primarily ranking the list of candidates based on their matched relationship of the user's input to each item's importance in the dataset (He, et al, 2019). In the human resource sphere, candidate generation vary based on hiring or applicant entity: hiring entity views candidate generation as being provided by the recommender system with list of applicants eligible to respective job postings based on applicant's profile, skillsets, and work experience (Liu, Cao & Song, 2020).

On the other hand, in the view of the applicant, it expects the recommender system to provide a list of job postings that are closely related to the applicant's profile, skillsets and work experiences or based on list of job posts viewed, rated, or recommended by other users to be relevant to their profile, skillsets and work experiences which are similar to the current user (Liu, et al., 2020). These views are based on the three most common strategies behind recommender systems: (1) global strategy in which job titles or job posts can be relevant to all users by providing users with the most popular; (2) contextual strategy which relies on relationship of job posts to applicants' attributes belonging to a specific group, industry, job sector, or location; and (3) personalized strategy which uses not only personal user attributes and job features but even overall user clicks, recommendations, search hits, clicks, etc. (Deutschman, 2023, Shivaji Godawale 2024).

AI-based job recommendation systems use a variety of techniques to match candidates to jobs based on their skills. One common technique is to use a collaborative filtering algorithm to recommend jobs to candidates based on the jobs that similar candidates have applied for or been accepted to (Deutschman, 2023). Another common technique is to use a content-based filtering algorithm to recommend jobs to candidates based on their profile, skills, work experience, and interests and based on job post detailed contents (Wang & Chen, 2021).

This study focused on the comparative utilization of content-based filtering approach and Apriori algorithm, which is a rule-based association approach, to empower contextual recommendations of Skills Extractor AI Agent and Job Recommender AI agent.

The output of this study is deemed relevant to the larger multi-agents Data-Driven Human Resource Analytics System (DDHRAS) concurrently being developed by the researchers which will in turn significantly help the subject institution in minimizing human errors during hiring process and increasing workplace efficiency and at the same time providing relevant information among stakeholders with significant accuracy and precision

2. METHODS

2.1 Research Design

The Quantitative-Developmental method of research was utilized by the researchers for this study taking into consideration that artificial intelligence uses quantitative research methods with experimental research design being the de facto research approach (Abu-El-Haija & Al-Khateeb, 2022).

The overall framework of this study is divided into 5 phases: data preparation and modelling, skills extraction, matching and scoring, job recommendation, and algorithm evaluation as displayed on Figure 1.

Phase 1 corresponds to data preparation and modelling in which the respective words and phrases utility matrices are prepared to be utilized by the content-based filtering (CBF) and Apriori algorithms. These data models can be made by extracting unique words and phrases from job post details and skills bank.

Once the data models are prepared, the Skills Extractor AI agent can start skills extraction by mining unique words and phrases per resumé which will then be matched word-by-word and phrase-by-phrase to each entry in the skills bank resulting to unique hard and soft skills lists. During the matching and scoring phase, these skills lists will then be processed with the corresponding utility matrix of each algorithm taking into account the similarity score (for the CBF algorithm) and support, confidence and lift metrics (for the Apriori algorithm). Scores and skills frequencies will be aggregated, and a list of skills ranked based on highest score and frequency will be prepared and passed to the recommender AI agent to start the recommendation phase.

With a list of recommended skills, under the job recommendation phase, the Job Recommender AI agent will either recommend list of job applicants or list of relevant job postings depending on system user type and intent: if system user is from the employer side, the intent is to list matched job applicants, on the other hand, list matched job postings for the applicant. Matchings will be done by utilizing the recommended skills given by the Skills Extractor AI Agent, user profile and job post contents.

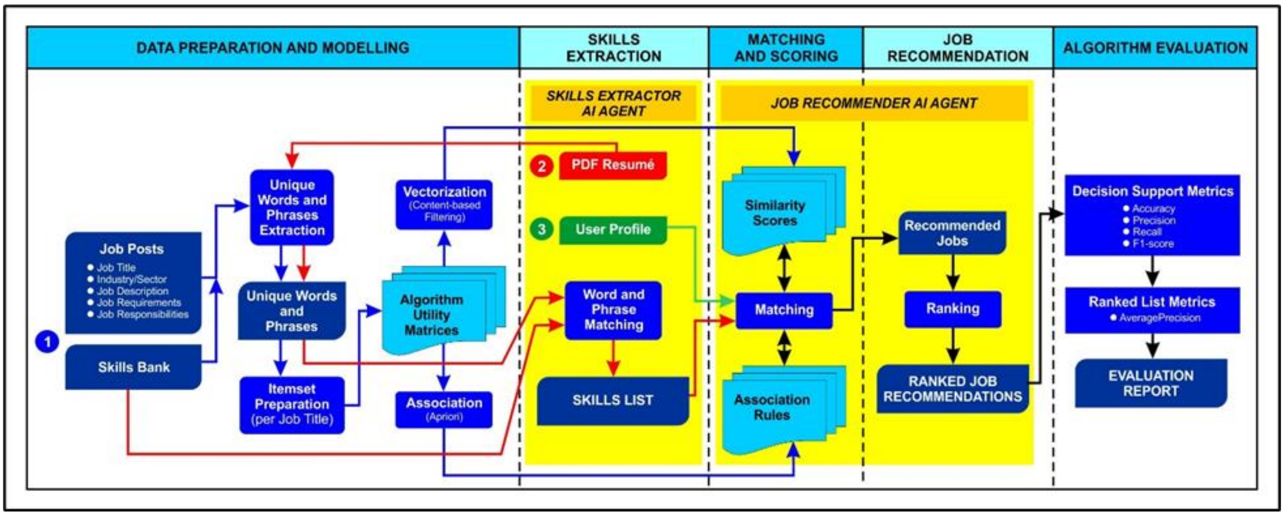


Figure 1. Conceptual Framework

The final phase relates to the evaluation of the recommendation performance of the two algorithms used in terms of decision support metrics and ranked list metrics to identify which algorithm performs better with the other.

2.2 Data Collection and Pre-Processing

The data for this study came from the main Human Resource department of Pangasinan State University situated at Lingayen, Pangasinan, Philippines on which one of the researchers is currently employed. Job applicant resumés and previous job postings that details job title, requirements, qualifications, and responsibilities were gathered via a formal letter and were collated to be part of development data. Meanwhile, related hard and soft skills for specific job sectors and job titles were collected from online resources.

A front-end user interface for job postings, skills, user profiles, job sectors, and resumé extractor were developed using PHP, HTML and Javascript programming languages and corresponding records were stored in separate tables in a MySQL database.

Since both the CBF and Apriori algorithms require words and phrases, no data normalization or transformation were made. The corresponding job title ID from the respective job titles table was utilized as identifier for the CBF algorithm's utility matrix. Meanwhile, for the Apriori algorithm, since each record of job posts, job titles, user profiles, and skills are organized and kept in their corresponding tables with their respective unique numeric identifier, such numeric identifiers were used for the Apriori's utility matrices.

A PHP script was created by the researchers to prepare the final dataset by iterating to every job post, fetching each job post's related job title and skills from corresponding tables, and extracting all unique words and their corresponding frequencies. Typically, the PDFparser library, which was used by the researchers as PHP word extractor function, would extract words and arrange them in an array as they appear on the PDF file. Hence, adjacent words appear in the array elements adjacently also. Unique words and their corresponding frequency were saved in a database table with a unique identifier.

After extracting the unique words, phrases were then extracted by iterating to every unique words of each job post. Length of phrases were set from 2 words to 5 words. Figure 2 shows an example of how phrases were extracted/formed using unique words records.

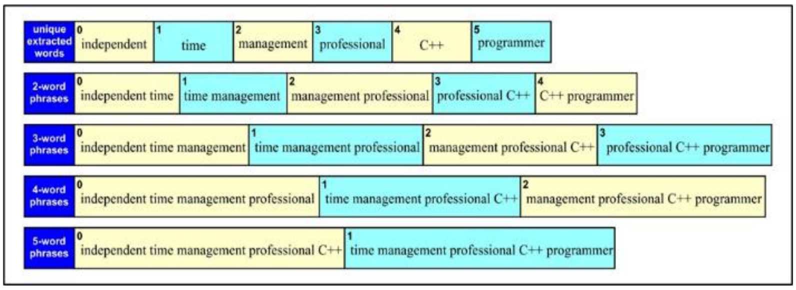
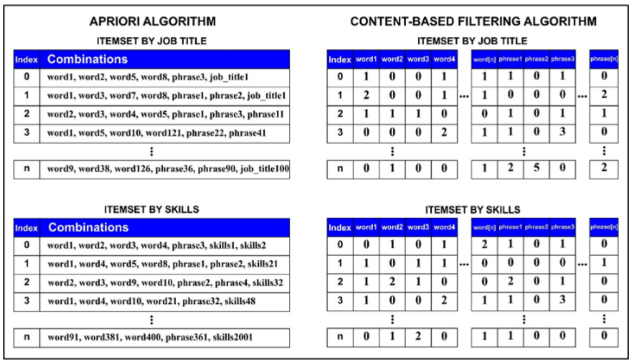


Figure 2. Phrases Extraction

Figure 2 illustrates that a phrase will be formed by joining 2 or more words from adjacent rows. Producing phrases is a better way of extracting skills, especially skills with 2 or more-word length. This also capture contiguous words appearing on the resumé file rather than depending on unique words which may result to frequent errors as there are words that can be match a phrased skill, but such words are located distantly on the resumé.

After extracting unique words and phrases, itemset preparation was conducted to organize the 4 final datasets which embodies the algorithm utility matrices consisted of unique words and phrases by skills and by job title. Figure 3 shows the table composition diagram of the final datasets for content-based filtering and Apriori algorithms.



APRIORI ALGORITHM

ITEMSET BY JOB TITLE

Index	Combinations
0	word1, word2, word5, word8, phrase3, job_title1
1	word1, word3, word7, word8, phrase1, phrase2, job_title1
2	word2, word3, word4, word5, phrase1, phrase3, phrase11
3	word1, word5, word10, word121, phrase22, phrase41
⋮	
n	word9, word38, word126, phrase36, phrase90, job_title100

ITEMSET BY SKILLS

Index	Combinations
0	word1, word2, word3, word4, phrase3, skills1, skills2
1	word1, word4, word5, word8, phrase1, phrase2, skills21
2	word2, word3, word9, word10, phrase2, phrase4, skills32
3	word1, word4, word10, word21, phrase32, skills48
⋮	
n	word91, word381, word400, phrase361, skills2001

CONTENT-BASED FILTERING ALGORITHM

ITEMSET BY JOB TITLE

Index	word1	word2	word3	word4		word[n]	phrase1	phrase2	phrase3		phrase[n]
0	1	0	0	1		1	1	0	1		0
1	2	0	0	1	...	1	0	0	0	...	2
2	1	1	1	0		0	1	0	1		1
3	0	0	0	2		1	1	0	3		0
⋮						⋮					⋮
n	0	1	0	0		1	2	5	0		2

ITEMSET BY SKILLS

Index	word1	word2	word3	word4		word[n]	phrase1	phrase2	phrase3		phrase[n]
0	0	1	0	1		2	1	0	1		0
1	1	0	1	1	...	0	0	0	0	...	1
2	1	2	1	0		0	2	0	1		0
3	1	0	0	2		1	1	0	3		0
⋮						⋮					⋮
n	0	1	2	0		1	1	0	0		0

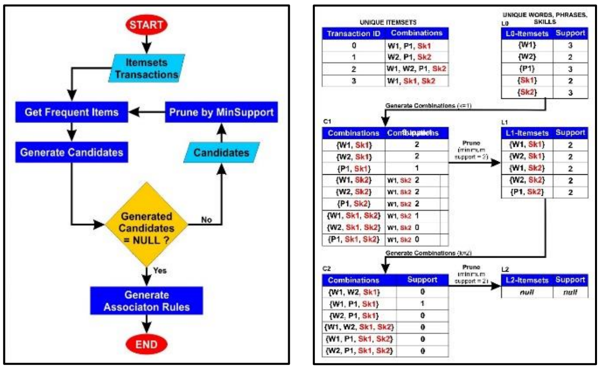
Figure 3. Utility Matrices or Final Datasets Composition

2.3 Data Modelling

The researchers used the PHP programming language to develop the models in Microsoft's Visual Studio Code as the integrated development environment (IDE) because it will be integrated in a web app. No data splitting was done as __ resumé files will be utilized to test the developed models. Overall, the final datasets were composed of ___ unique words, ___ phrases which will be linked to __ skills and __ job titles.

The researchers chose the Apriori and CBF algorithms to experiment which will provide a better ranked list of skills or job recommendations Based on the works of Latifah, Akhriza, and Adistia (2019), the Apriori algorithm is a widely-used data mining technique that is simple and easy to implement, explain and interpret as the process and output rules are human-readable. Aside from flexibility in terms of customization, the researchers chose Apriori since it performs well even on unlabeled data, thus, saving a lot of time in data preparation. On the other hand, Fatourechi (2019), the CBF algorithm is a popular machine learning algorithm used currently even by the largest companies such as Netflix, IMDB, Rotten Tomatoes, and Pandora as it can narrow down decision-making processes via a shortlist based on empirical relevant values. The researchers chose CBF as it fits the recommendation problem at hand and its simplicity to be implemented in the PHP programming language.

Figure 4 displays the flowchart and sample illustration of Apriori algorithm as modified by the researchers to pair words (*W*) and phrases (*P*) to specific skills (*Sk*) or job title (*JT*).



(a) (b)

Figure 4. Apriori Algorithm Flowchart and Itemset Association Illustration

For the Apriori algorithm to make an association, it requires to iterate to every rows of combinations to pair unique words and phrases to a specific skill or job title by *k* elements candidate set (*C*) and increase k until no more other pairings can be done. Usually, the initial value of k is always set to 1 to map all unique items. Since there is a need to associate the combination of words (*W*) and phrases (*P*) to a skill (*Sk*) or job title (*JT*), a modified pairing is introduced by the researchers to ensure that there exists one or more skills or job titles for every itemset pairings.

Apriori's objective is find the association rules that satisfy a minimum support and minimum confidence thresholds. Support quantifies the frequency of an item set in a dataset. It is calculated as the proportion of transactions containing the item set to the total number of transactions.

$$Support(\{X\} \rightarrow \{Y\}) = \frac{Transactions\ containing\ both\ X\ and\ Y}{Total\ number\ of\ transactions} \quad (1)$$

Confidence is the number of times the pairing of items *X* and *Y* divided by the total number of transactions with *Y* item. Confidence is the degree of the strong relationship of the antecedent (transactions with *X* and *Y*) and the consequent (transactions with *X*). Its formula is as follows:

$$Confidence(\{X\} \rightarrow \{Y\}) = \frac{Transactions\ containing\ both\ X\ and\ Y}{Transaction\ containing\ X} \quad (2)$$

Lift can be used to relate to the confidence or likelihood of having item *Y* with or without pairing with it item *X*. The formula for lift is as follows:

$$Lift(\{X\} \rightarrow \{Y\}) = \frac{Confidence(\{X\} \rightarrow \{Y\})}{Fractions\ of\ transaction\ containing\ Y} \quad (3)$$

Aside from support, confidence, and lift, the researchers followed the works of Latifah, et al. (2019) in ranking the results of the association rules which is expressed as:

$$Score(R) = Sup(R) \times Conf(R) \times Lift(R) \times Length(R) \quad (4)$$

where Score is the total score obtained by a rule *R*, Sup is for the rule support, *Conf* is the rule confidence, *Lift* is the rule lift value, and *Length* is the number of skills or job titles in the rule.

Meanwhile, according to Roy (2020), the CBF algorithm is a powerful machine learning technique that puts a numerical value to plaintexts based on contextual features or attributes of an item through word vectorization known as Term Frequency - Inverse Document Frequency (TF-IDF). The researchers computed for the *TF* and *IDF* of each frequency in the utility matrices using the following formulas:

$$TF(i \rightarrow D) = 1 + \log_{10} f(i \rightarrow D) \quad (5)$$

$$DF(i) = \sum_{n=1}^{N} f\ (i \rightarrow D) \quad (6)$$

$$IDF(i) = \log_{10}\left(\frac{N}{DF(i)}\right) \quad (7)$$

$$LV(D) = \sqrt{\sum_{t=1}^{T} DF(i)^2} \quad (8)$$

where *TF* is the term frequency, *i* is the item, *D* is the document, *f* is frequency, *DF* is the document frequency, *N* is the total number of documents, n is the starting index of documents, *IDF* is the Inverse Document Frequency, *LV* is the length of the document vector, *T* is the total number of items or columns, and t is the starting index of item columns.

When the *TF-IDF* of the algorithm utility matrices have been computed for each skill or job title vector, each vector can be related and differentiated among other skills or job titles through distance metrics to determine how far or near is a particular skill or job title to the rest. The distance metrics can be Euclidian, Manhattan, Jaccard or Cosine distance (Makwana, 2022). In this study, the researchers employed the cosine distance or similarity scoring which is mathematically formulated as:

$$cos(x, y) = \frac{x \cdot y^T}{\|x\| \cdot \|y\|} = \frac{\sum_{i=1}^{n} x_i \cdot y_i^T}{\sqrt{\sum_{i=1}^{n}(x_i)^2}\sqrt{\sum_{i=1}^{n}(y_i)^2}} \quad (9)$$

where *x* is the vectorized item being compared from the matrix of items, *y* is the list of all vector items in the matrix except for *x* and is indexed by *i* as the initial row identifier and n as as the total number of rows in the matrix while *T* is the total number of columns which corresponds to the total number of rows in a single-columned matrix forming the vector. In this study, *x* and *y* are the vectors of a skill or job title being compared.

According further to Makwana (2022), the cosine similarity score would produce a float value between -1.0 to 1.0 in which having a near 1.0 cosine similarity score means the more similar the items being compared are while having low score means being dissimilar. Hence, the ranking of recommendations will be based on the highest cosine similarity score among suggested items.

The output of computing the cosine similarity score of each vectorized item in a utility matrix is a similarity score matrix of shape (*N x N*) where *N* is the total number of documents (skills or job titles). The similarity score matrix corresponds to the finished model. In this study, the researchers aimed to develop 2 similarity score matrices, 1 for the job titles and 1 for skills.

2.4 Performance Evaluation

Accuracy, precision, recall and F1-score metrics were used to evaluate the performance of the models. The number of accurate and inaccurate list of recommendations made by the similarity score and rules association models in relation to the expected recommendation list (target value) are monitored by the researchers.

Accuracy refers to the proportion of the total number of predictions that were correct and is mathematically defined as:

$$Accuracy = (TP + TN)/n \quad (10)$$

where *TP* refers to True Positives, *TN* represents True Negatives and n is the total number predictions, in this case, the total elements of the recommendation list.

Meanwhile, precision, also known as the positive predictive quantity, is the number of correct predictions out of the combined positive correct predictions and negative incorrect predictions and is attributed to hold a high value when negative incorrect prediction are low. The formula of Precision is:

$$Precision = \frac{TP}{TP+FP} \quad (11)$$

Recall, also referred to as the true-positive rate or sensitivity, is the portion of positive correct predictions over successful predictions and is also distinguished to maintain a high value with fewer negative incorrect guesses. Its formula is written as:

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

Finally, F1-score is the congruent mean of Precision and Recall with a direct similar effect on Precision and Recall such that it has a higher value when either Precision or recall is high and F1-score becomes low if either Precision or Recall is also low. The formula for F1-score is:

$$F1 - score = 2\frac{Precision * Recall}{Precision + Recall} \quad (13)$$

Meanwhile, in terms of evaluating the relevance of each item in the recommended list, the researchers employed the Mean Average Precision at *K* or *MAP@K* accuracy metrics which, according to Rink (2023), is one of the most popularly used metrics to test the overall relevance of the ranking results of a recommender system and is mathematically defined as:

$$MAP@K = \sum_{k=1}^{K}\left(\frac{1}{m}\sum_{k=1}^{K} P(k) \cdot rel(k)\right) \quad (14)$$

where *MAP@K* is the mean average precision of recommended *K* items, *K* is the total number of recommendation items in the list, k is the index of the item, *P(k)* is the precision at *k*, and *rel(k)* is the relevance of the kth item which is either 1 or 0 for relevant or not relevant, respectively. The Precision at *k* can be expressed as:

$$P@K = \frac{\sum_{k}^{K} rel(i)}{K} \quad (15)$$

where *P@K* is the precision at *K*, *rel(i)* is the number of relevant items in top *K* results, and *K* is the total number of recommended items.

Since the offline evaluation approach will be utilized, it will be researchers themselves who will check for the relevance and correctness of each recommended skill or job title for each query or recommendation request.

3. RESULTS AND DISCUSSION

Table 1 highlights the comparative evaluation performance of the Apriori and CBF algorithms in extracting relevant soft and hard skills among 40 PDF resumés used as test data during the integration testing of the Skills Extractor AI Agent in the Data Driven Human Resource Analytics System developed by the researchers.

Table 1. Comparative Evaluation Performance of Apriori and CBF Algorithms in Recommending Relevant Skills

#	Metrics	Apriori	CBF	Diff
1	Accuracy	87.61	92.30	4.69
2	Precision	86.59	91.57	4.98
3	Recall	87.13	90.81	3.68
4	F1-Score	87.21	90.64	3.43
5	MAP@K	85.74	90.38	4.64
	Mean Diff			**4.28**

Legend: **Diff** = difference; **CBF** = content-based filtering algorithm; **MAP@K** = mean average precision at K; **Mean Diff** = Mean Difference

Performance data displayed on Table 1 have shown that both Apriori and content-based algorithm performed well in recommending list of relevant skills based on words and phrases extracted from test resumé files surpassing the usual 85% accuracy baseline which is a threshold often used as a benchmark for comparing the performance of different machine learning models (Wilson, et al., 2019).

Shown on Table 2 is the comparative evaluation performance of the Apriori and CBF algorithms in recommending relevant job titles based on skills extracted among 40 PDF resumés which were used as test data during the integration testing of the Job Recommender AI Agent in the Data Driven Human Resource Analytics System developed by the researchers.

Table 2. Comparative Evaluation Performance of Apriori and CBF Algorithms in Recommending Relevant Job Titles

#	Metrics	Apriori	CBF	Diff
1	Accuracy	85.77	91.82	6.05
2	Precision	85.06	90.79	5.73
3	Recall	84.89	90.04	5.15
4	F1-Score	85.01	89.91	4.90
5	MAP@K	84.71	89.76	5.05
	Mean Diff			5.38

Legend: **Diff** = difference; **CBF** = content-based filtering algorithm; **MAP@K** = mean average precision at K; **Mean Diff** = Mean Difference

Table 2 provides experimental results which show that both Apriori and content-based algorithm obtained a significant accuracy and precision in suggesting list of relevant job titles based on extracted skills from test resumé files. Accuracy performance of the two utilized algorithms also exceeded the 85% accuracy threshold.

Data presented on Tables 1 and 2 reveal that the content-based filtering algorithm consistently outperformed the Apriori algorithm across all metrics with a mean difference of 4.28% for the skills recommendation and 5.38% for the job title recommendation. These findings are parallel to the works of Sankarasetty, et al. (2022) which also conducted a comparative study of utilizing Apriori and CBF algorithms for job recommendations and found out that the CBF algorithm always surpassed the Apriori algorithm in terms of accuracy, precision, recall, and F1-score measures because of CBF's ability to capture semantic relationships between job descriptions and user profiles unlike Apriori which relied purely on word-by-word matching and frequency.

Interestingly, data from Table 2 indicates that the Apriori algorithm performed better in recommending relevant skills than in suggesting relevant job titles due to the fact that extracted skills have fewer words and phrases to compare with the Apriori's utility matrices than with extracted words and phrases from the PDF resumés which are more plentiful, hence, higher word-for-word matchings and frequency values. Further, the researchers investigated the instances when Apriori struggled to provide relevant listings of job titles and found out that it has obtained low overall scores on cross-industry or multi-sectoral skills as shown on Figure 5.

Data on Figure 5 revealed that the Apriori algorithm scored low on associating soft skills and some hard skills that are common core skills that employees must have regardless of job sector.

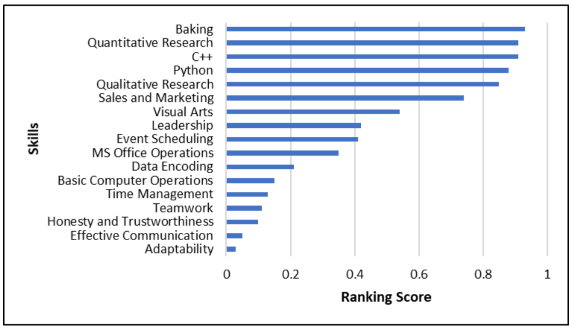


Figure 5. Sample Apriori Ranking Performance on Cross-Sectoral Skills

Guided by the experimental results of the comparative performance of the Apriori and content-based filtering algorithms, the researchers chose and sticked with the latter algorithm in the further and optimal development of the Resumé Skills Extractor and Job Recommender AI Agents for the Data-Driven Human Resource Analytics System.

4. CONCLUSION

Resumé Skills Extractor and Job Recommender AI agents were developed by the researchers in this study utilizing content-based filtering approach and Apriori algorithm to provide relevant list of skills based on extracted words and phrases from a PDF resumé file and list of relevant job titles based on input skills. The developed AI agents were successfully integrated into a much larger Data-Driven Human Resource Analytics System which the researchers are simultaneously developing.

References:

Abu-El-Haija, S., & Al-Khateeb, A. (2022). "Research methods in machine learning: A content analysis." *International Journal of Computer and Information Technology, 11*(1), 1-12.

Latifah, B.B., Akhriza, T.M., & Adistia, L.D. (2019). Constructing Recommendation about Skills Combinations Frequently Sought in IT Industries Based on Apriori Algorithm. *Advances in Computer Science Research, volume 95* doi:10.2991/miseic-19.2019.12

Chen, Y., Wang, Z., Sun, J., & Wang, L. (2020). A deep learning approach for skills extraction from job postings. *IEEE Transactions on Knowledge and Data Engineering, 33(*1), 194-206.

Deutschman, Z. (2023). Recommender Systems: Machine Learning Metrics and Business Metrics. In MLOps Blog, Neptune.ai. Retrieved: 2023 October 08 from https://neptune.ai/blog/recommender-systems-metrics.

Fatourechi, M. (2019). The Evolving Landscape of Recommendation Systems. In Media & Entertainment, TechCrunch. Retrieved 2023 October 18 from https://techcrunch.com/2015/09/28/the-evolving-landscape-of-recommendation-systems/..

He, R., Chua, J. Y., & Guan, Z. (2019). DeepRec: A deep interest-aware recommendation framework for personalized job recommendation. In Proceedings of the 2019 ACM International Conference on Information and Knowledge Management (CIKM) (pp. 2275-2284).

Jatoba, M., Santos, J., Gutierriz, I., Moscon, D., Fernandex, D. O., & Teixeira, J. P. (2019). Evolution of Artificial Intelligence *Research in Human Resources. Procedia Computer Science, 164,* 137-142. doi: 10.1016/j.procs.2019.12.165

Liu, Y., Cao, Y., & Song, Q. (2020). A machine learning approach for skills extraction from resumes. *IEEE Transactions on Human-Machine Systems, 50*(3), 254-267.

Makwana, A. (2022). Building a Content-Based Recommendation System. In Data Science Blogathon, Analytics Vidhya. Retrieved 2023 October 18 from https://www.analyticsvidhya.com/blog/ 2022/08/building-a-content-based-recommendation-system/.

Nain, V., & Shyam, H. S. (2024). Empirical Analysis of The Role of Artificial Intelligence in Human Resources Recruitment and Selection. *Proceedings on Engineering Sciences*, *6*(2), 817-826. doi: 10.24874/PES06.02A.008

Rink, K. (2023). Mean Average Precision at K (MAP@K) clearly explained. Towards Data Science. Retrieved 2023 October 18 from https://towardsdatascience.com/mean-average-precision-at-k-map-k-clearly-explained-538d8e032d2.

Roy, A. (2020). Introduction to Recommender Systems- 1: Content-Based Filtering And Collaborative Filtering. Towards Data Science. Retrieved: 2023 Oct 18 from https://towardsdatascience .com/introduction-to-recommender-systems-1-971bd274f421.

Shivaji Godawale, Prashant Deshmukh (2024). IMPACT OF BLOCKCHAIN TECHNOLOGY IN THE E-COMMERCE SECTOR. *Journal of Trends and Challenges in Artificial Intelligence*, *4*(2), 127-132. doi: 10.61552/JAI.2024.04.002

Sankarasetty, S., Kakaraparti, S., Munadi, S.H., & Jonnalagadda, S.K., (2022). A Comparative Study of Content-Based Filtering and Apriori Algorithm for Job Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *34*(12), 2793-2805

Wang, X., & Chen, Y. (2021). A hybrid approach for job recommendation based on machine learning and collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, *34*(6), 2662-2675.

Wilson, R.C., Shenhav, A., Straccia, M. & Cohen, J.D. (2010). The Eighty Five Percent Rule for Optimal Learning. *Nature Communications 10*, 4646 (2019). doi: 10.1038/s41467-019-12552-4

Paul Nigel S. Abalos
Pangasinan State University,
San Carlos City,
Philippines
paulnigelabalos@gmail.com
ORCID 0000-0003-1866-9274

Thelma D. Palaoag
University of the Cordilleras,
Baguio City,
Philippines
tdpalaoag@uc-bcf.edu.ph
ORCID 0000-0002-5474-7260