**Career Path Recommendation System Using Content-Based Filtering on Linkedln Profile Dataset**

**Akbar Muhammad Sadat\*1), Dimas Febrianto2), Akmal Eggy Terea3) , Marcel Hans4).**

1)Informatics Department, Telkom University Surabaya, Jombang, Indonesia

2)Informatics Department, Telkom University Surabaya, Sidoarjo, Indonesia

3)Informatics Department, Telkom University Surabaya, Bogor, Indonesia

4)Informatics Department, Telkom University Surabaya, Surabaya, Indonesia

[1\*](mailto:1*first@institution.ac.id)[akbarmuhammadsadat@telkomuniversity.ac.id](mailto:akbarmuhammadsadat@telkomuniversity.ac.id),

2 [dimasfebrianto@telkomuniversity.ac.id](mailto:dimasfebrianto@telkomuniversity.ac.id),

3 [akmaleggyte@telkomuniversity.ac.id](mailto:akmaleggyte@telkomuniversity.ac.id),

4 [marcelhans@telkomuniversity.ac.id](mailto:marcelhans@telkomuniversity.ac.id)

|  |  |  |
| --- | --- | --- |
| **A R T I C L E I N F O** |  | **A B S T R A C T** |
| ***Article history:***  Received 7 November 2023  Revised 21 December 2023  Accepted 31 February 2024 Available online 07 May 2024 | A succinct and informative paragraph is indispensable, encompassing key components such as the rationale for the study, the proposed solution, the methodology, proposed contributions, research objectives, obtained results, and conclusion. Moreover, it is essential to accentuate the distinctive advantages that our approach affords in comparison to existing methods. Do not include procedure steps and citation sources in this part. It should be concise, ranging from 100 to 250 words, using single spacing. As an article advertisement, it must be crafted with an emphasis on both allure and comprehensibility, providing a compelling overview of the study's elements. |
| ***Keywords:***  Maximum  Five  Important  Key  Words |
| *This is an open access article under the* [*CC BY-SA*](https://creativecommons.org/licenses/by-sa/4.0/) *license.* | | |

1. **Introduction**

In the current era of Industry 4.0, the integration of Big Data and Artificial Intelligence (AI) has significantly changed various sectors, including Human Resource Management. Linkedln, as one of the largest professional networking platforms, holds millions of profiles and job listings, generating huge amounts of unstructure data[1]. For fresh graduates and job seekers, navigating this vast amount of information of find a career path that matches their specific skills can be overwhelming. As a result, the mismatch between job seekers qualifications and industry requirements remains a critical issue, creating a need for intelligent tools to bridge this gap effectively[2].

To handle this information overload, recommendation systems have become a vital tool for filtering and prioritizing relevant items. Previous studies generally categorize job recommendation approaches into Collaborative Filtering (CF) and Content-Based Filtering (CBF). While Collaborative Filtering is widely used, it often suffers from the "cold start" problem for new users who lack historical data[3] On the other hand, Content-Based Filtering focuses on the attributes of the items and the user profile. However, existing research often relies on limited datasets or simple keyword matching, neglecting the semantic relationships within the text. Many current systems lack precision in parsing complex resume data found in LinkedIn profiles, highlighting the need for more robust content analysis methods.[1]

Addressing these challenges, this study proposes a Career Path Recommendation System utilizing Content-Based Filtering on a LinkedIn Profile Dataset. The rationale behind this approach is to use Natural Language Processing (NLP) techniques—specifically Term Frequency-Inverse Document Frequency (TF-IDF) and Cosine Similarity—to analyze profile content such as skills, summaries, and experiences, and match them against job requirements. By using Content-Based Filtering, the system aims to provide personalized career recommendations even for users with minimal interaction history, directly solving the limitations found in collaborative approaches.[4]

The primary objective of this research is to develop a recommendation model that can accurately predict suitable career paths based on the textual content of a user's LinkedIn profile. Specifically, this study aims to preprocess unstructured LinkedIn data, implement the Content-Based Filtering algorithm to calculate the similarity between user profiles and career paths, and evaluate the system's performance. This research contributes to the field by demonstrating the effectiveness of content-based filtering on professional social media datasets, providing a practical solution for students and job seekers to identify career opportunities that statistically match their competencies.[2]

1. **Research Methods**

This study uses a quantitative experimental approach to design and evaluate the Career Path  Recommendation System. The main methodology relies on Content-Based Filtering techniques to process unstructured text data from LinkedIn profiles[5] In essence, the system works by transforming text attributes into vector representations and calculating the semantic similarity between a user's profile and various career specifications. The following sections describe the system architecture, data collection, the preprocessing pipeline, feature extraction, and how similarity is measured.

**2.1. System Architecture**

The system architecture consists of several modules working together to turn raw data into actionable recommendations. The process starts with data collection, followed by a strict text preprocessing phase to clean and standardize the input. After that, the data is transformed into a structured format using the Vector Space Model (VSM). Finally, the system calculates similarity scores to rank potential career paths. Figure 1 shows the complete workflow, from the initial data input to the final recommendation output[6]

A diagram of a step-by-step process

AI-generated content may be incorrect.

Figure 1 The Architecture of the Proposed Recommendation System

**2.2. Datasets Description**

This research utilizes the "LinkedIn Profiles and Jobs Data," which is publicly accessible via the Kaggle repository. Curated by the contributor killbot, this dataset contains approximately 10,000 anonymized user profiles featuring employment history records up to January 2018.

The dataset offers a detailed look at professional backgrounds, including essential attributes such as job titles, company names, and the duration of employment. For the purpose of building this content-based recommendation system, the study focuses on extracting and combining textual features from the job history (specifically job titles and descriptions) to build a semantic representation of each career path. To ensure the quality of the model, any profiles containing incomplete text information or missing job records were filtered out before the analysis began.[7]

**2.3. Text Preprocessing**

Because the raw data was gathered through web scraping, it contains a significant amount of noise and unstructured elements that need to be cleaned to maximize the performance of the TF-IDF algorithm[8] The preprocessing pipeline used in this study involves the following steps:

* Handling Missing Values : First, the system identifies null or NaN values in key text columns, such as Job Titles. Rows with significant missing data are removed to prevent errors or bias in the recommendation model.[9]
* Data Concatenation : To create a single, unified document vector for each profile, relevant text columns (such as current Job Title and previous roles) are merged into one continuous string.[8]
* Case Folding : All text is converted to lowercase. This ensures uniformity, for example, treating "Manager" and "manager" as the same word.[10]
* Noise Removal : Using Regular Expressions (Regex), the system removes non-alphanumeric characters, including punctuation, special symbols (like /, @, #), and numbers. This ensures that only meaningful words remain for analysis.[10]
* Stop-word Removal : Common English words that carry little semantic meaning (such as "and", "the", "of", "in") are removed using the NLTK library to focus on relevant keywords.[8]
* Tokenization & Stemming : Finally, the cleaned text is split into individual tokens. A stemming process (using the Porter Stemmer) is then applied to reduce words to their root forms (e.g., changing "developing" to "develop"), which helps reduce the complexity of the feature space.[9]

**2.4. Feature Extraction using TF-IDF**

Following  the preprocessing phase, the text data—which is now a combined string of job titles and descriptions—needs to be converted into a numerical format that the machine learning algorithm can understand. To do this, this study uses the Term Frequency-Inverse Document Frequency (TF-IDF) vectorization technique.

In the context of this research, the entire collection of LinkedIn profiles acts as the "corpus," while each individual user profile is treated as a single "document."[11]The TF-IDF algorithm assigns a specific weight to each term (word) based on its statistical importance:

* Term Frequency (tf): This measures how often a specific term t appears in a profile d. A higher frequency indicates that the term is significant to that specific user's professional identity.
* Inverse Document Frequency (idf): This measures how rare a term is across the entire dataset. Terms that appear in almost every profile (generic words like "work" or "employee") are given lower weights, while specific skill keywords (such as "TensorFlow" or "Epidemiology") receive higher weights because they are more distinctive.

The formula for calculating the weight is defined as follows:

|  |  |
| --- | --- |
|  | 2.1. |

The IDF measures the rarity of a term across the entire corpus. Words that appear in nearly every document will have a low weight, while words that appear infrequently will have a higher weight. Logarithmic and smoothing factors are used to maintain the stability of the weight values.

|  |  |
| --- | --- |
|  | 2.2. |

TF-IDF weights are obtained by multiplying TF and IDF. A high weight value will occur if a word appears frequently in one document but rarely in another. This representation is effective for distinguishing user profiles based on text content and is widely used in content-based recommendation systems[12]

Here, N represents the total number of profiles in the dataset. The result of this process is a sparse matrix where rows represent user profiles and columns represent the vocabulary of unique professional terms.

**2.5. Similarity Measurement using Cosine Similarity**

To generate career path recommendations, the system calculates the semantic similarity between the active user's input profile (the Query Vector) and the existing profiles in the dataset (the Target Vectors). We utilize Cosine Similarity as the distance metric for this calculation.[12]

We chose Cosine Similarity over Euclidean distance because it measures the cosine of the angle between two vectors rather than their magnitude (length). This property is particularly useful for this dataset because the length of employment history varies significantly among users. For example, a user with a short but highly relevant summary should still result in a good match with a longer career path if the key terms align[13]

The similarity score ranges from 0 (completely different) to 1 (identical). The similarity between the user vector *A* and a dataset profile vector *B* is calculated using Equation (2.3.):

|  |  |
| --- | --- |
|  | 2.3. |

Brief Explanation of Each Componen :

* *A*:the TF-IDF vector representing the user profile (query)
* *B*:the TF-IDF vector representing a profile in the dataset
* *:* the dot product, which measures the diractional similarity between the two vectors
* : the vector magnitudes (lengths), used for normalization
* *:* the total number of features (unique terms)[14]

The system computes this score for every profile in the database. Finally, the results are sorted in descending order, and the profiles (or associated job titles) with the highest cosine similarity scores are presented to the user as the Top-N Career Recommendations.

1. **Results and Discussions**

This section explains some of the results of the research conducted. However, not all research results need to be displayed in this section. Just a few important research results. It is necessary to discuss in detail the interesting results. Discussion can also be carried out by comparing the results obtained with the results from previous research. To make it easier for writers to describe research results, tables and figures can be used in this section.

1. **Tables and Figures**
2. **Writing Rules for Tables and Figures**

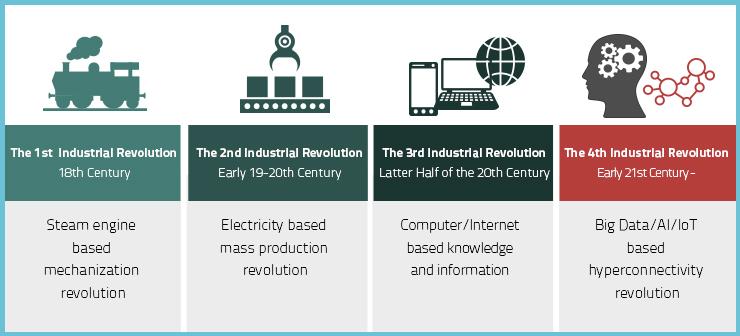
Tables and figures are numbered sequentially according to the order in which they appear. The table title/name is placed above the table, while the image title/name is placed below the image. Every table or figure displayed in an article must be cited in the body of the article. The font size for tables and figures is 10 pt. If the table or figure is taken from another source, then include the name of the source below the table or figure.

Vertical lines in tables are not allowed. Meanwhile, horizontal lines are only allowed on the first row of the table and the last row, as well as to separate certain parts/groups of tables (for example table titles). Example:

**Table 1.** Definition of Notations

|  |  |
| --- | --- |
| **Notation** | **Definition** |
| *v* | velocity (m/s2) |
| *s* | distance (m) |
| *t* | time (s) |

Ensure the image has a sufficient level of resolution. If there is writing on the image, make sure it is legible. Example:



**Figure 1.** The 4th Industrial Revolution

Each mathematical equation must be assigned a number. Each notation is written in an italic. The equation number is written in a parenthesis and placed to the right of the equation. Example:

|  |  |
| --- | --- |
|  | (1) |

**4. Conclusion**

The conclusion section must be written in a paragraph, and cannot be written in bullet points. Conclusions must be able to answer the research objectives. Explain the important results obtained related to the research objectives. Apart from that, this section also explains the limitations of the research conducted and the future research agenda.

**Acknowledgement**

This section is optional. In this part, the author can express his or her thanks to parties (individuals or institutions) who helped with research and writing articles.

[1] S. Nayak, “CareerCue: A Survey on Intelligent Job Recommendation Systems Using Skill Extraction and Conversational AI Harnessing Skill Extraction and AI Chatbots for Next-Generation Career Assistance.” [Online]. Available: www.ijsdr.org

[2] F. Fakhriyah, H. D. Adiyanti, R. Almutazan, and M. A. Saryatmo, “INDUSTRIAL REVOLUTION 4.0: THE PROCESS OF RECRUITING QUALIFIED HUMAN RESOURCES,” *International Journal of Application on Economics and Business*, vol. 2, no. 4, pp. 796–803, Nov. 2024, doi: 10.24912/ijaeb.v2i4.796-803.

[3] G. Vinodhini and R. Suban, “A State Of The Art Survey On Cold Start Problem In A Collaborative Filtering System”, [Online]. Available: www.ijstr.org

[4] R. Alana and A. Hartanto, “Sistemasi: Jurnal Sistem Informasi Implementasi Algoritma Content Based Filtering dalam Sistem Rekomendasi Komik Implementation of Content Based Filtering Algorithm in Comic Recommendation System.” [Online]. Available: http://sistemasi.ftik.unisi.ac.id

[5] A. Rianti, N. W. A. Majid, and A. Fauzi, “MACHINE LEARNING JOURNAL ARTICLE RECOMMENDATION SYSTEM USING CONTENT BASED FILTERING,” *JUTI: Jurnal Ilmiah Teknologi Informasi*, pp. 1–10, Jan. 2024, doi: 10.12962/j24068535.v22i1.a1193.

[6] V. Meida Hersianty, E. Larasati Amalia, D. Puspitasari, and D. Wahyu Wibowo, “PENERAPAN ALGORITMA TF-IDF DAN COSINE SIMILARITY DALAM SISTEM REKOMENDASI LOWONGAN PEKERJAAN,” 2025.

[7] L. Almalki, N. Al-Alsheikh, F. Bahareth, M. Abdullah, and R. Alsahafi, “BERT-based Job Recommendation System Using LinkedIn Dataset,” 2024. [Online]. Available: https://www.jisem-journal.com/

[8] R. Budiarto Hadiprakoso, H. Setiawan, R. Novita Yasa, and P. Siber dan Sandi Negara, “Text Preprocessing for Optimal Accuracy in Indonesian Sentiment Analysis Using a Deep Learning Model with Word Embedding.”

[9] H. Ar Rosyid, A. Y. H. Putra, M. I. Akbar, and F. A. Dwiyanto, “Can Multinomial Logistic Regression Predicts Research Group using Text Input?,” *Knowledge Engineering and Data Science*, vol. 5, no. 2, p. 150, Dec. 2022, doi: 10.17977/um018v5i22022p150-159.

[10] Z. Jianqiang and G. Xiaolin, “Comparison research on text pre-processing methods on twitter sentiment analysis,” *IEEE Access*, vol. 5, pp. 2870–2879, 2017, doi: 10.1109/ACCESS.2017.2672677.

[11] M. A. F. Fitria and E. B. Setiawan, “Hybrid Deep Learning with GloVe and Genetic Algorithm for Sentiment Analysis on X: 2024 Election,” *Scientific Journal of Informatics*, vol. 11, no. 3, pp. 693–704, Aug. 2024, doi: 10.15294/sji.v11i3.8467.

[12] M. Siino, I. Tinnirello, and M. La Cascia, “Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers,” *Inf Syst*, vol. 121, Mar. 2024, doi: 10.1016/j.is.2023.102342.

[13] C. P. Chai, “Comparison of text preprocessing methods,” *Nat Lang Eng*, vol. 29, no. 3, pp. 509–553, 2023, doi: DOI: 10.1017/S1351324922000213.

[14] Z. Jing, Q. Yang, M. Qin, J. Li, and K. S. Kwak, “Long-Term Max-Min Fairness Guarantee Mechanism for Integrated Multi-RAT and MEC Networks,” *IEEE Trans Veh Technol*, vol. 70, no. 3, pp. 2478–2492, Mar. 2021, doi: 10.1109/TVT.2021.3059944.