

Text Mining of Job Vacancy Data for Skill Extraction

1. Introduction

Online job vacancies published on platforms such as LinkedIn and Indeed contain large amounts of unstructured text. These job descriptions describe required skills, technologies, and responsibilities, but they are written in free text and lack a common structure. Text mining and basic Natural Language Processing (NLP) techniques are commonly used to analyze this data and extract meaningful information about skill demand and job roles.

2. Text Mining of Job Vacancies in Previous Work

Previous research shows that a common approach to analyzing job vacancies is skill extraction using predefined skill lists or taxonomies. In this method, job descriptions are scanned for known skill keywords. This approach is widely used because it is simple, transparent, and easy to implement.

A practical industry example of this approach is presented by SDK (2025), who analyzed LinkedIn job postings using keyword-based skill extraction combined with clustering to identify job role groups. This work is especially relevant because it applies simple and interpretable techniques similar to those used in this project.

Some studies apply more advanced NLP techniques, such as Named Entity Recognition or transformer-based models, to identify skills in context. While these methods can improve extraction accuracy, they require labeled data and additional complexity, which is not always suitable for small applied projects.

Data provisioning and feature engineering are important in job-vacancy text mining, because the raw postings are noisy (HTML fragments, duplicated postings, missing fields, inconsistent titles) and models depend strongly on how text is cleaned and represented.

In the survey by Senger et al. (2024), many datasets and approaches rely on a skill base (e.g., ESCO or O*NET) to define labels and to standardize extracted skill mentions. This means the “data provisioning” step is often: collect job postings, decide the granularity (whole posting, sentence, or span), and map extracted spans to a predefined taxonomy so that different spellings or synonyms end up as the same skill.

Tzimas et al. (2024) describe a full processing pipeline before any NLP model is applied. Their methodology includes selecting multiple reputable sources, extracting postings, then performing cleansing, normalization, and deduplication. They also highlight handling missing values as part of preprocessing, and treat information extraction (skills, occupation, employer, location, experience) as a separate final phase after data cleaning.

Boselli et al. (2018) and related WoLMIS work focus on collecting large-scale web vacancies from heterogeneous sources and then converting them into a consistent text classification

dataset. A key feature engineering choice is to represent postings using bag-of-words / n-gram text features (often using title words and short text fields, because they are highly predictive). This representation is then used for supervised classification into a standard occupation taxonomy.

In the practical pipeline by SDK (2025), feature engineering is kept lightweight and interpretable: the job-description text is cleaned by removing clutter such as links and punctuation, and extracted skills are normalized so that abbreviations (e.g., "ML") map to a single canonical name. This improves clustering quality because similar roles share the same standardized skill tokens.

3. Skill Normalization and Job Role Analysis

After skill extraction, normalization is applied to reduce duplication caused by synonyms, abbreviations, and different spellings of the same skill. This step improves data consistency and allows better comparison between job postings.

To analyze job roles, researchers often cluster vacancies based on their extracted skill sets or use classification models to assign vacancies to predefined job categories. These methods help identify groups of similar roles and understand how professions are defined by combinations of skills.

4. Application in This Project

This project follows a practical applied approach. A predefined skill dictionary is used to extract skills from job descriptions collected from multiple datasets. Extracted skills are normalized to ensure consistency. In this project, a similar lightweight preprocessing and normalization approach is used. The focus is not on complex NLP pipelines, but on making the data clean, structured, and understandable, so that the models remain transparent and suitable for an applied student project where explainability and practical usability are important.

Vacancies are clustered based on their skill sets, and a classification model is used to predict the most relevant skill group from a job title. As a result, the system can output the top required skills for a given profession.

5. Conclusion

Existing research confirms that text mining of job vacancies is an effective way to analyze skill demand and job role structure. The methods used in this project are consistent with previous work but focus on simplicity and practical implementation, making them suitable for an applied sciences bachelor project.

References

- [1] Senger, E. et al. (2024). Deep learning-based computational job market analysis: a survey on skill extraction and classification from job postings. NLP4HR Workshop.
<https://aclanthology.org/2024.nlp4hr-1.1/>
- [2] Tzimas, G. et al. (2024). From Data to Insight: Transforming Online Job Postings into Labor-Market Intelligence. *Information*, 15(8), 496. <https://www.mdpi.com/2078-2489/15/8/496>
- [3] Boselli, R. et al. (2018). WoLMIS: a labor market intelligence system for classifying web job vacancies. *Journal of Intelligent Information Systems*, 51(3), 477–502.
<https://link.springer.com/article/10.1007/s10844-017-0488-x>
- [4] SDK, E. (2025). Analyzing LinkedIn Job Postings: Skill Extraction & Clustering.
<https://dev.to/esdk/analyzing-linkedin-job-postings-skill-extraction-clustering-17c7>