Course Data Lab 1

Final Report

Job and Course Recommendation System for LinkedIn Users

Submitted by:

Itzhak Danan, Yehezkel Nikop , Refael Levi

1. Project Introduction

As part of our final project, we developed an intelligent recommendation system designed to help LinkedIn users discover jobs and courses that match their personal profiles. The system operates in three main stages:

1. **Company Similarity Identification**: This component receives the user's most recent employer and returns the three most similar companies. The similarity is determined using a combination of features, including textual company descriptions processed with TF-IDF, numerically encoded company size, and number of followers. A labeled dataset of company pairs (marked as similar or non-similar) was constructed to train a classification algorithm that predicts the probability of similarity between companies. All data processing and feature engineering were performed using **Apache Spark**, including the use of Spark DataFrames, column transformations, and joins. The methodology closely follows the techniques introduced in **Homework 3** of the course, allowing for scalable and efficient processing of the LinkedIn dataset.
2. **Relevant Job Matching**: Given the similar companies, we scraped open job listings and filtered them based on their similarity to the user's "about" section, salary range, and geographic distance. A composite score was calculated for each job to rank relevance.
3. **Course Recommendation**: For every relevant job, a set of courses was matched based on textual similarity between job descriptions and course texts. Clustering was performed using KMeans to group courses and improve recommendation focus.

The overall goal of the system is to provide data-driven, personalized career and learning recommendations even for users without extensive employment history. The system integrates NLP, machine learning, and semantic matching to bridge user profiles with job opportunities and upskilling resources.

---

2. Data Collection and Integration

Throughout the project, we utilized multiple internal and external data sources to build our recommendation system:

- **Primary Data Sources**: We used two LinkedIn datasets – linkedin/people and linkedin/companies, which provided detailed profile and company descriptions.
- **Open Job Listings**: In addition to the internal data, we used a publicly available dataset from Kaggle (linkedin_124k_kaggle.csv) containing job descriptions, requirements, locations, salaries, etc.
- **Course Scraping from FutureLearn**: We scraped course data using Selenium, BeautifulSoup, and pandas from professional categories like "AI and Machine Learning," "Cyber Security," and "Project Management." To bypass access restrictions, we utilized Bright Data's proxy service. Data collected included course title, description, institution, and URL, stored in a file named course_listings_2.pkl.
- **Data Integration**: After identifying similar companies (stage 1), job listings were retrieved (from Kaggle), and relevant courses were matched for each job using textual similarity.
- **Item Definition and Enrichment Size**: Each item is defined as a job listing. On average, 8 new columns were added to each job post, including normalized salary, location distance, profile similarity score, and recommended courses.
- **Visualizations of the data, column structures, and datasets** are included in the appendix.

---

3. Data Analysis

- **Analysis Techniques**: TF-IDF was used for text vectorization of jobs and courses. KMeans clustering was applied to segment the courses. Jobs were scored using cosine similarity, salary, and distance.
- **Feature Selection**: Features were selected based on domain expertise and statistical correlation: job title, description, skills, salary, and location.
- **Visualizations**: Visual outputs included cluster distribution graphs, heatmaps for course-job similarities, and relevance score distributions. Main visualizations appear in the report; additional plots are in the appendix.

---

4. AI Methodologies

- **Company Similarity Model**: Logistic Regression was trained on a labeled dataset of company pairs using features like TF-IDF-based similarity, company size (numeric), and follower count (with missing data handled). The model was validated using 5-fold cross-validation, achieving **F1 = 0.9074** and **ROC-AUC = 0.9626**. The final function accepts a user ID and returns the top 3 most similar companies.
- **Job Matching Algorithm**: A hybrid scoring model was developed using:
    - BERT for semantic similarity between user profile text and job description
    - Geopy for calculating geographic distance
    - Normalized salary and job type (full-time preferred)
    - Weighted formula: 60% text similarity, 20% distance, 10% job type, 10% salary
    - Challenges: heavy computation with BERT, API rate limits in Geopy, missing data, and outlier influence from high salaries
    - Suggested improvements: caching, adaptive weight tuning, using lightweight NLP models, and including more profile-based features
- **Course Matching Model**: TF-IDF vectors were computed for both job and course texts. KMeans with K=15 was used to cluster courses (K chosen via elbow and silhouette methods). Jobs were compared to their nearest cluster centroid, and cosine similarity was used to rank course relevance.

Additionally, a separate module allows users to input specific skills and receive relevant course recommendations. This addresses users who lack employment history or are targeting a specific skill.

The system performed efficiently, offering scalable and relatively accurate textual recommendations. Limitations include shallow textual analysis and the lack of deep contextual understanding, which could be mitigated by using models like BERT for embedding-based comparisons.

---

5. Limitations and Reflection

- **Technical Constraints**: BERT required heavy computation; limited GPU access slowed progress.

- **Partial and Missing Data**: Some users and jobs lacked location, salary, or descriptions.
- **Dependency on Proxies**: Occasional scraping issues due to Bright Data limitations.
- **Weight Sensitivity**: Scoring sometimes skewed by outliers (e.g., extreme salary).
- **No Ground Truth**: Lack of labeled outcomes for recommendation accuracy evaluation hindered supervised training.
- Despite these, independent tests showed the models produced highly relevant outputs.

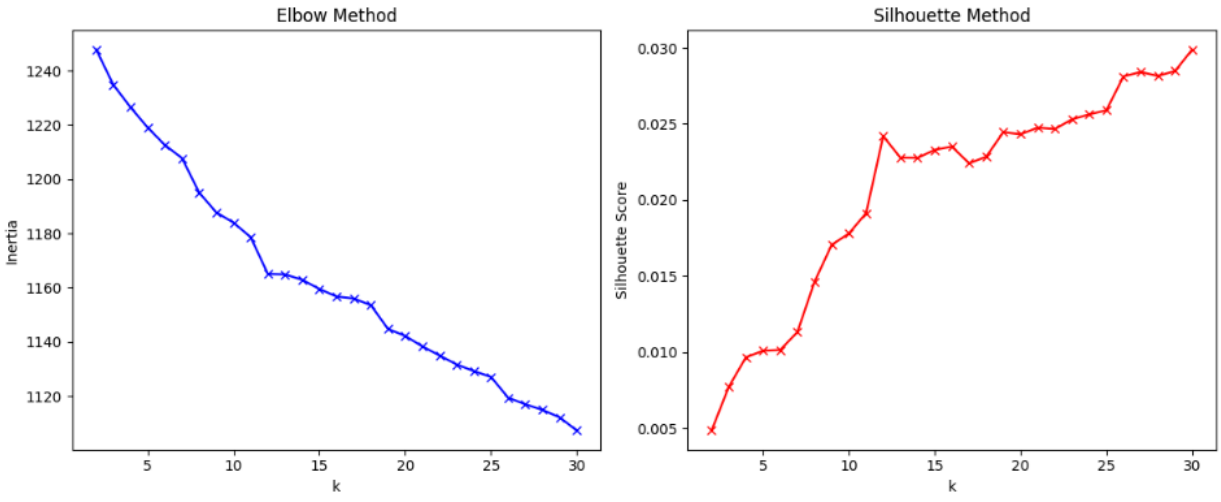
---

6. Conclusions

Our system offers a complete, intelligent, and scalable solution for job and course recommendations tailored to LinkedIn users. It performs well even with partial user data, and future improvements may include deep learning, emotional inference, and profile-personalized matching.

Appendix

- **Demo Videos**:
  - [User Journey Video – with narration]
  - [Technical Overview Video]
- **Visualizations**:
  - KMeans cluster selection graph:



K האופטימלי לפי silhouette score: 30
K האופטימלי לפי elbow method: 12

- Heatmap: job-to-profile matching:

	job_title	job_description	job_location	salary	job_type	normalized_salary	proximity_score	normalized_proximity_score	full_job_text	cosine_similarity_score	job_similarity_score
0	Data Scientist	Looking for a data scientist with expertise in...	New York, NY	120000	Full-Time	0.800000	1.000000	1.000000	Data Scientist Looking for a data scientist wi...	0.857836	0.894701
3	AI Researcher	Researcher in AI and deep learning application...	Boston, MA	140000	Full-Time	0.933333	0.693913	0.693913	AI Researcher Researcher in AI and deep learni...	0.825797	0.827594
12	Data Architect	Designing and managing complex data architectu...	Toronto, Canada	125000	Full-Time	0.833333	0.448753	0.448753	Data Architect Designing and managing complex ...	0.835844	0.774590
11	AI Product Manager	Managing AI-based products for consumer applic...	San Jose, CA	145000	Full-Time	0.966667	0.000000	0.000000	AI Product Manager Managing AI-based products ...	0.811929	0.683824
4	Data Engineer	Expertise in building data pipelines and cloud...	Austin, TX	115000	Full-Time	0.766667	0.000000	0.000000	Data Engineer Expertise in building data pipel...	0.840487	0.680959
16	Cloud AI Engineer	Deploying AI solutions in cloud environments.	Tel Aviv, Israel	130000	Full-Time	0.866667	0.000000	0.000000	Cloud AI Engineer Deploying AI solutions in cl...	0.817911	0.677413
15	Big Data Engineer	Processing and managing large-scale datasets.	Sydney, Australia	120000	Full-Time	0.800000	0.000000	0.000000	Big Data Engineer Processing and managing larg...	0.827377	0.676426
1	Machine Learning Engineer	We need an ML engineer to build and deploy dee...	San Francisco, CA	130000	Full-Time	0.866667	0.000000	0.000000	Machine Learning Engineer We need an ML engine...	0.805243	0.669812
9	Deep Learning Engineer	Focus on training neural networks for NLP tasks.	Paris, France	135000	Full-Time	0.900000	0.000000	0.000000	Deep Learning Engineer Focus on training neura...	0.799497	0.669698
8	Quantitative Analyst	Analyzing financial data and building predicti...	Hong Kong	150000	Full-Time	1.000000	0.000000	0.000000	Quantitative Analyst Analyzing financial data ...	0.775121	0.665072

- Distribution of match scores

- **References**:
  - Academic papers on TF-IDF, clustering, recommender systems
  - HuggingFace Transformers, scikit-learn, geopy, BrightData, spark

- **Key Resources**:
  - GitHub repository with full code and README:

    https://github.com/chezi-nikop/DataLab

  - Google Drive with cleaned datasets and visualizations: https://drive.google.com/drive/folders/1OnR9EMFlNNsNN3u2Hl7QEkor_m_Q8sGV?usp=drive_link

  - project demonstration video:

    https://drive.google.com/drive/folders/1ZnQMmPHOd507nDV-H_7NN7YzFmiRJQHa?usp=drive_link