Web Analytics Final Project (Group 13)

By: Marcos Sanson, Kelvin Martinez, Tapanga Witt

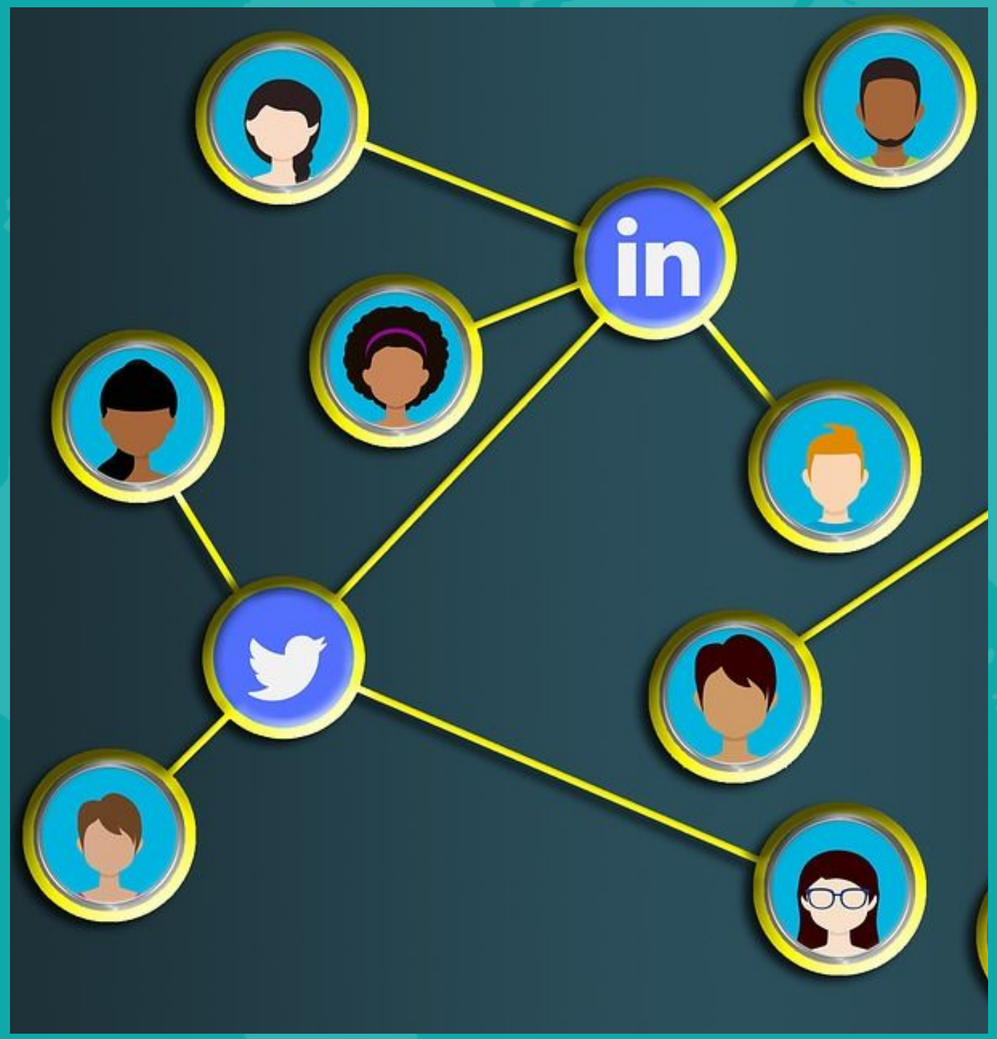
Project Idea

➔ **Create a network and career-growth platform for students and recent graduates**

- Offers personalized job matches and tracks application history
- Provides detailed feedback/advice to help users improve their chances for each applications (skills, keywords, other users related to job)

Goal and Motivation

- To encourage students and recent graduates to network
- To help students find jobs and opportunities based off their skills and profile
- Students struggle with gaining experience related to their major and skills
- Helps address the need for students to find relevant jobs and opportunities that fit their studies and skill set



State of the Art

→ **Existing Platforms and APIs:**

- **LinkedIn**
- **Indeed**
- **Glassdoor**

Key Features: filter by role, location, experience level; real time data

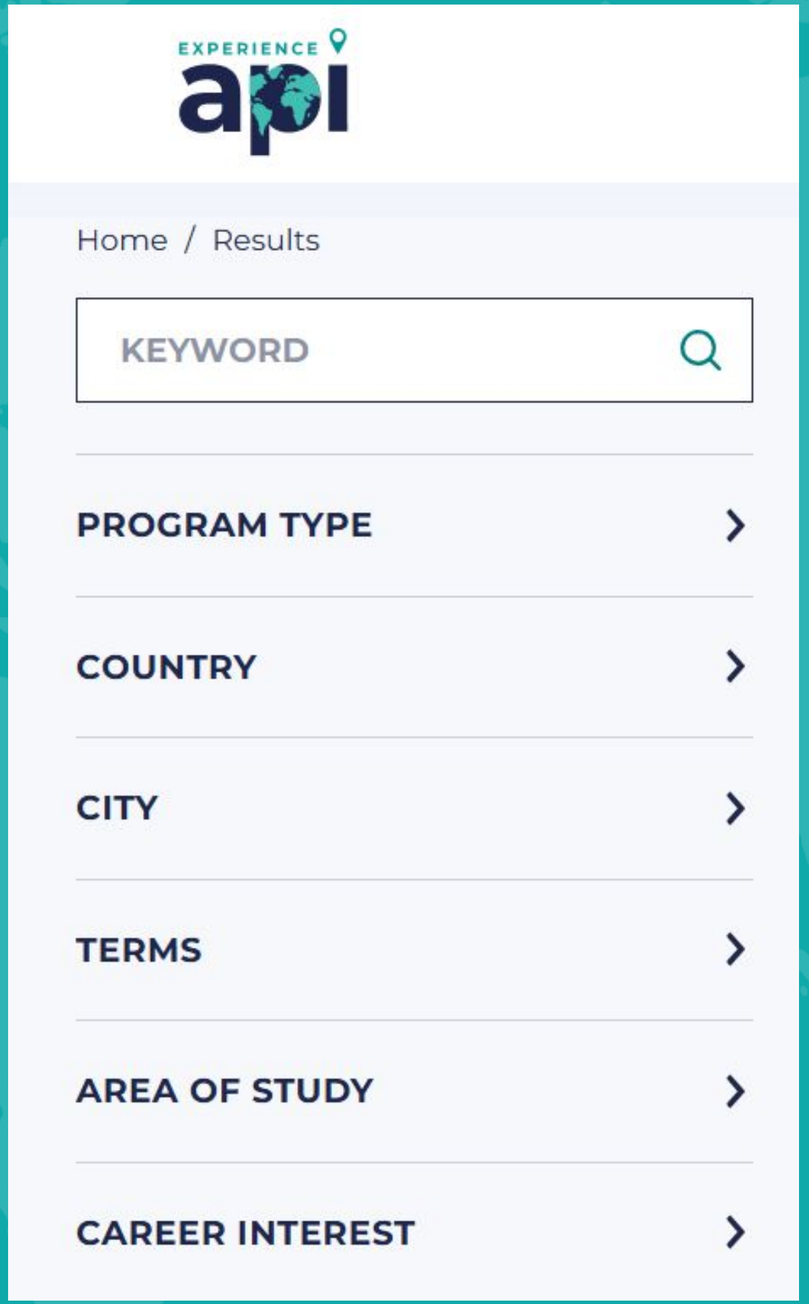
State of the Art

→ **Data analysis techniques**

- **Natural Language Processing (NLP):** Extract skills, keywords, and trends from job descriptions
  - **Example:** presence of the word “Python” in engineering job descriptions
- **Machine Learning:** Matches users to jobs; improves recommendations with time

State of the Art

- **Visualization and Insights**
  - **Interactive Dashboards:** Customizable filters for users preferences
  - **Company Insights:** Salary reports, interview experiences, total job openings
- **Emerging techniques**
  - **Graph Analysis:** Modeling relationships between job categories, companies, industries
  - **Chatbots:** Guide users, refine search, enhance user engagement



Our Data Source

→ **Adzuna API:** It provides real-time job listings

The data includes:

- Job title
- Location
- Salary information
- Company
- Job description

Data Gathering Techniques

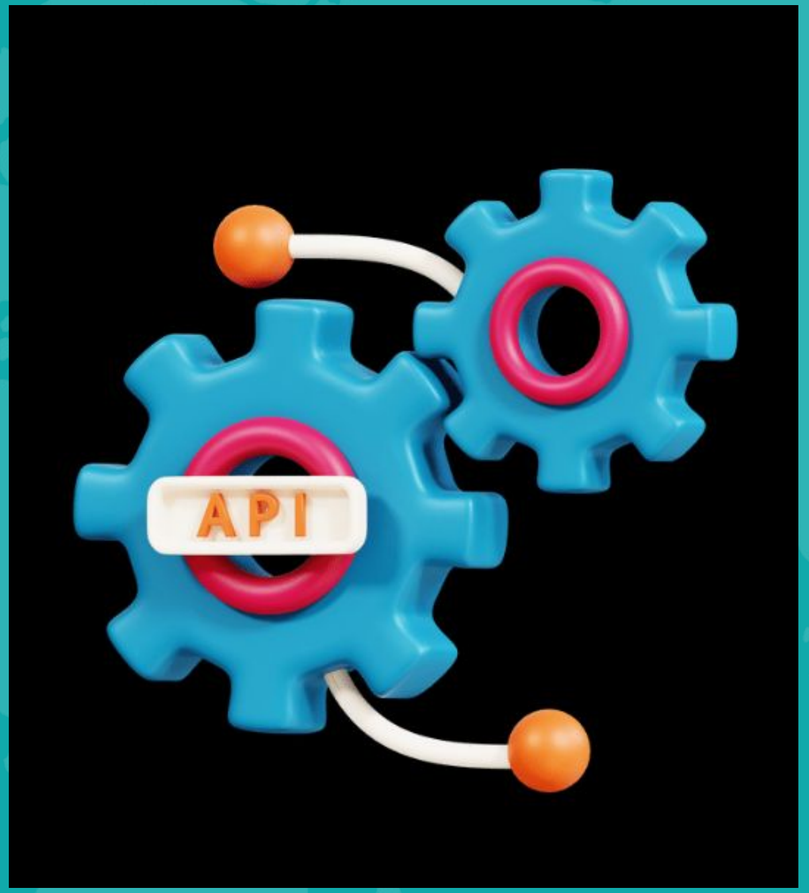
→ **API Requests**:

- **Base url:** https://api.adzuna.com/v1/api/jobs
- **Example of request:** gb/search/1?app_id=my_app_id&app_key=my_app_key&results_per_page=50&where=London&what=Engineer
- **We send a get request to the API**
  - Every time we make a query the response is returned as an object serialized using JSON

Data Gathering Techniques

→ **How it works:**

- **Base URL:** Specifies the endpoint for job data
- **Query Parameters:**
  - **app_id:** Your unique application ID
  - **app_key:** Your unique API key
  - **results_per_page:** Number of job results to retrieve per request
  - **where:** Location (e.g., London)
  - **what:** Job title or keyword (e.g., Engineer)



Data Gathering Techniques

→ Data Processing Workflow

1. Data Extraction:

- Extract the JSON element from the API response for further processing

2. Data Cleaning:

- Handle missing values
- Convert numeric fields (e.g., salaries) to integers or floats for accurate calculations

3. Data Storage:

- Use Pandas to save the cleaned data into CSV files


```
{
  "__CLASS__": "Adzuna::API::Response::JobGeoData",
  "locations": [
    {
      "__CLASS__": "Adzuna::API::Response::LocationJobs",
      "count": 10978,
      "location": {
        "__CLASS__": "Adzuna::API::Response::Location",
        "area": [
          "UK",
          "West Midlands",
          "Birmingham"
        ],
        "display_name": "Birmingham, West Midlands"
      }
    },
    .... more child locations here ...
  ]
}
```

Problems found/Possible issues

- → **API limitation:** 1000 requests/hour
- → **Inconsistent Data:** Some job listings lack salary information, or latitude and longitude information
- → **API errors:** Some errors (error 503) could occur, requiring retries
- → **Presence of duplicates**
- → **Jobs offers might not be available for certain regions**

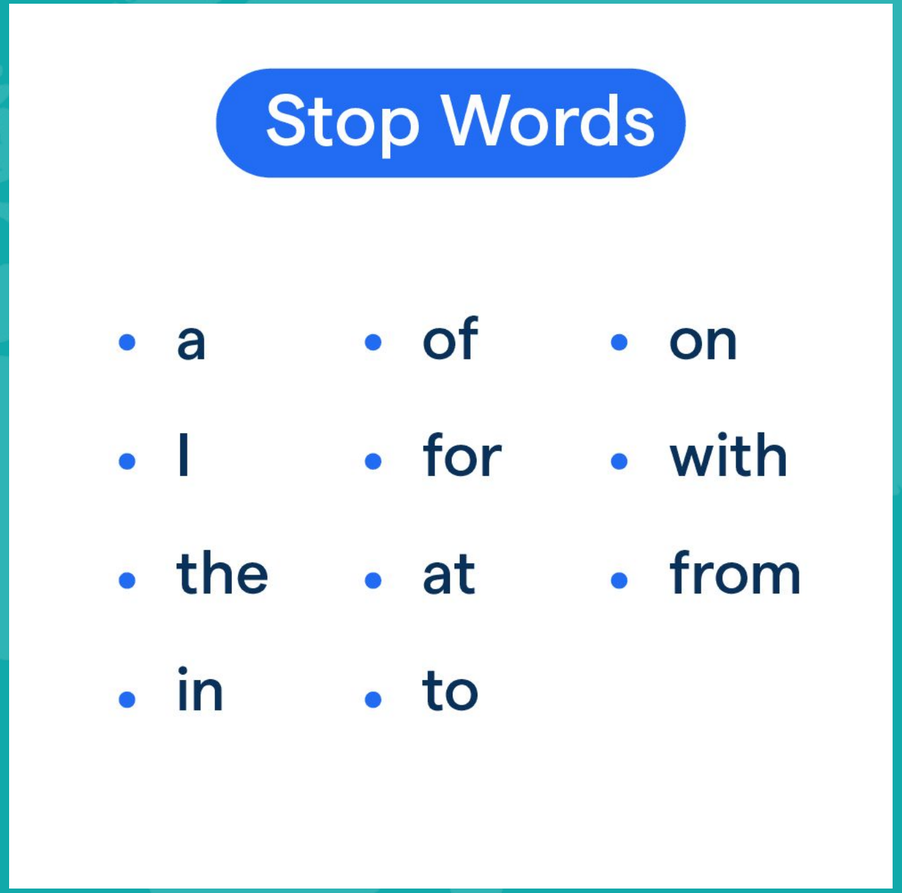
	title
0	DAML smart contract Developer
1	Dual Fuel Smart Meter Engineer
2	Dual Fuel Smart Meter Engineer
3	Dual Fuel Smart Meter Engineer
4	Dual Fuel Smart Meter Engineer
5	Controls Engineer
6	Sales Engineer/Pre - Sales
7	Technical Escalation Engineer
8	Customer Support Engineer - Onsite
9	Customer Support Engineer

Data Analysis Techniques

- **We used natural language processing (NLP) to calculate a similarity score between the user's profile and the job description:**
  - We converted the job descriptions and the user's profile in numerical vectors based on word frequency and importance
  - Then we compared the vector representing the user's profile and the job descriptions

Problems faced/Potential issues

- **Stop words used for NLP:**
  - This implies that we can only calculate similarity scores for languages of chosen stop words (here English)
- **Data aggregation:**
  - Careful handling of the dataset to ensure no duplicates or errors and making it appropriate for visualisation



Problems faced/Potential issues

→ **Difficulty to observe long time trends**

- The only tool given to us by the API is an history feature that returns the salary for given job category and location over a time period

Ex: it-jobs in New-York the response is :

"month": {
"2024-07": 65025.13,
"2024-09": 78193.44,
"2024-03": 71314.31,
"2024-04": 69694.3,
"2024-06": 62789.18,
"2024-10": 78704.29,
"2024-05": 57984.3,
"2024-02": 70137.13,
"2024-11": 75506.25,
"2024-01": 69238.5,
"2024-08": 74562.62,
"2023-12": 70545
},

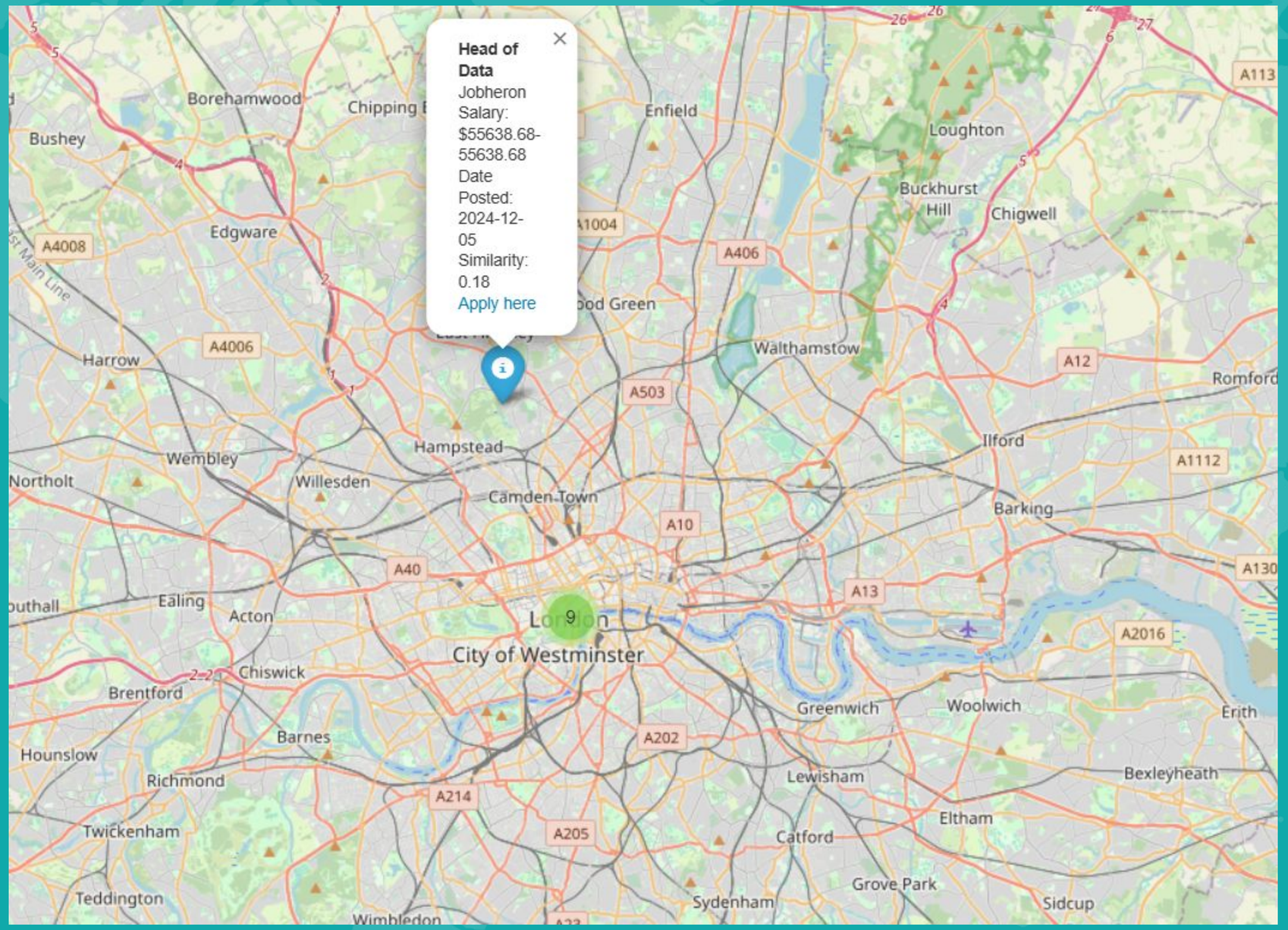
Data Visualization

→ We created a Folium map that shows the jobs having the best similarities for a given user

- Example for the following user:

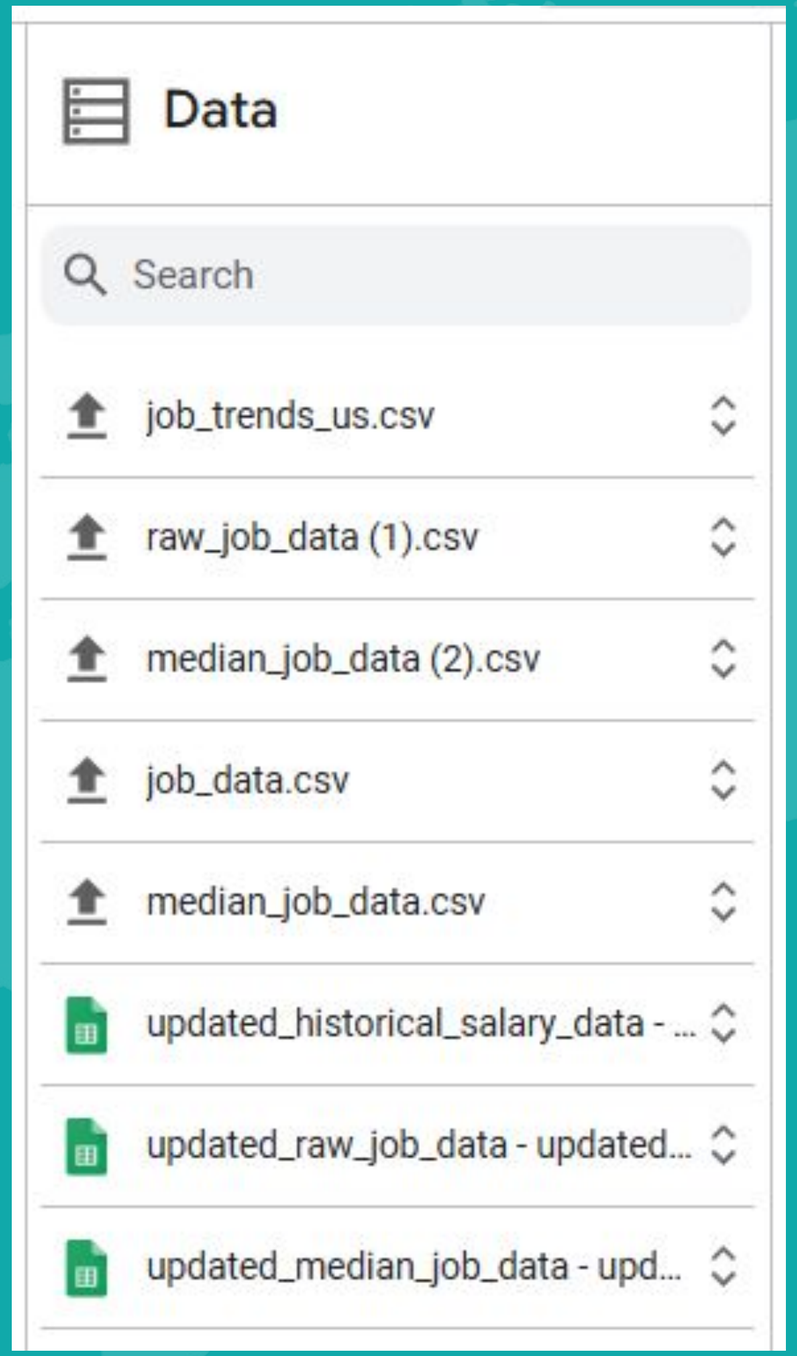
```
user_profile = {
        "field": "Data",
        "experience": "Senior",
        "location": "London",
        "min_salary": 30000,
        "max_salary": 150000,
        "category": "IT Jobs",
        "date_posted_within_days": 7
}
```

Data Visualization



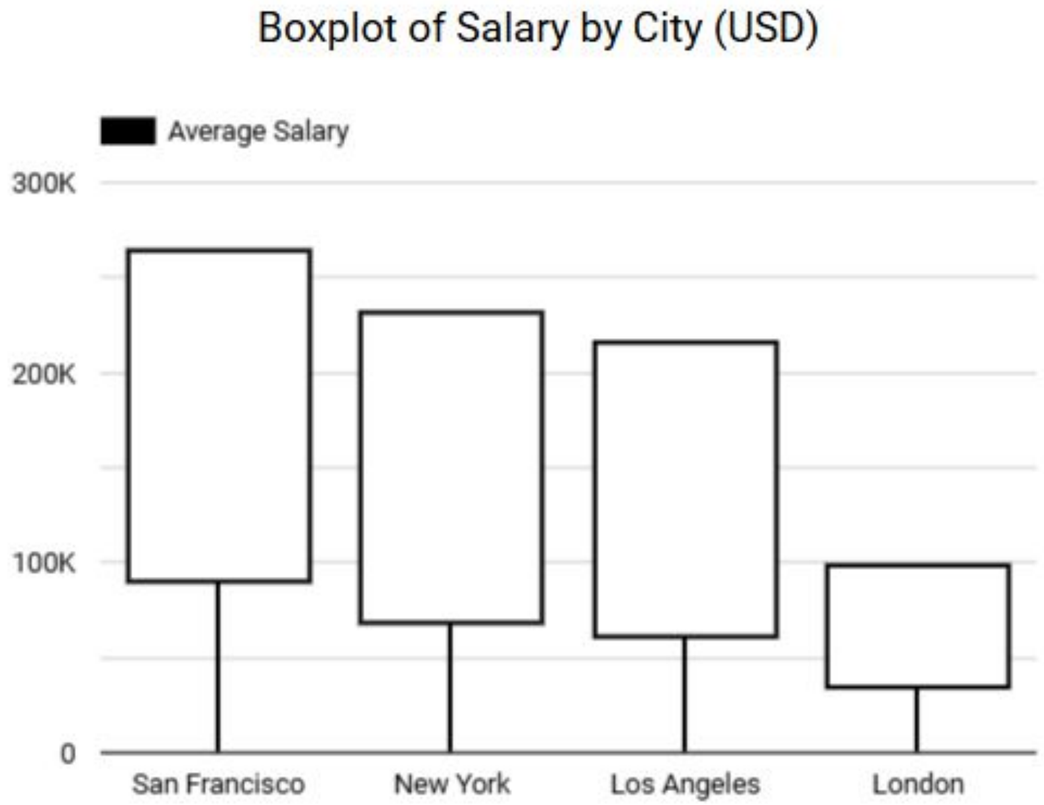
Data Visualization

- Creation of a Google dashboard:
  - A barplot to visualize the median salaries for certain job offers in different cities
  - A boxplot to visualize the salaries for certain job offers in different cities



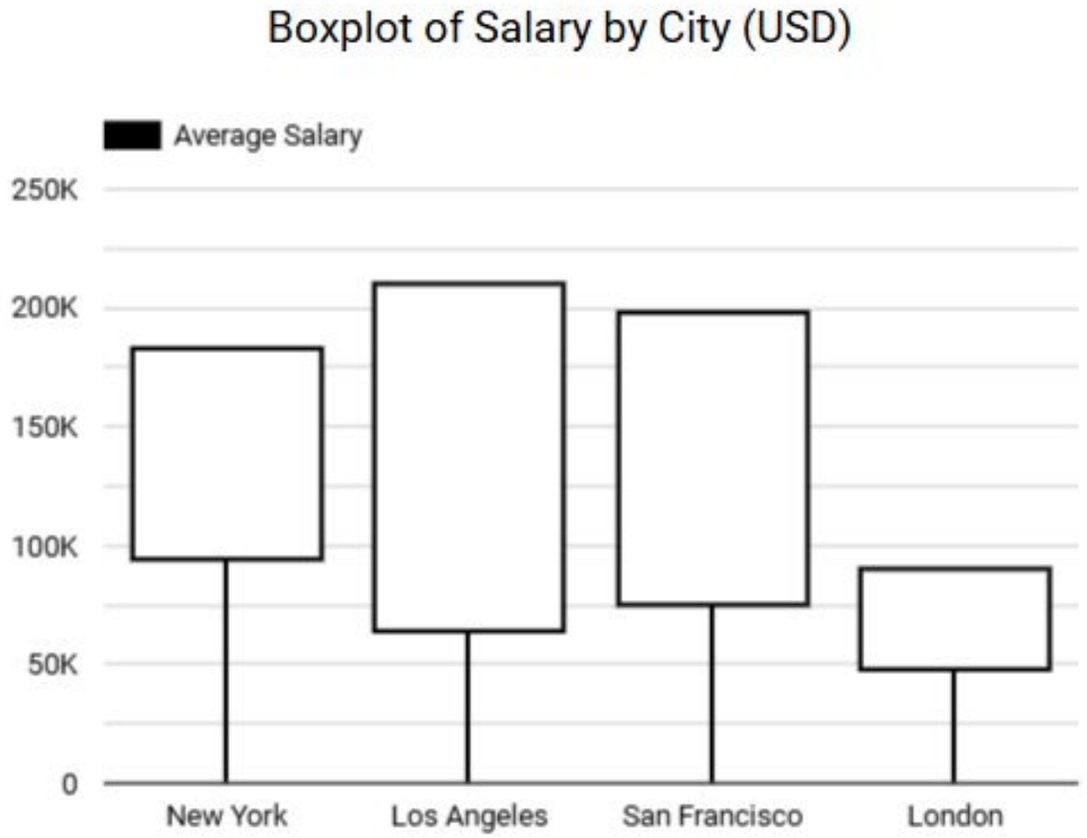
Data Visualization (Software Engineer)

Job Title: Software Engineer (1)



Data Visualization (Cybersecurity Specialist)

Job Title: Cybersecurity Specialist (1)

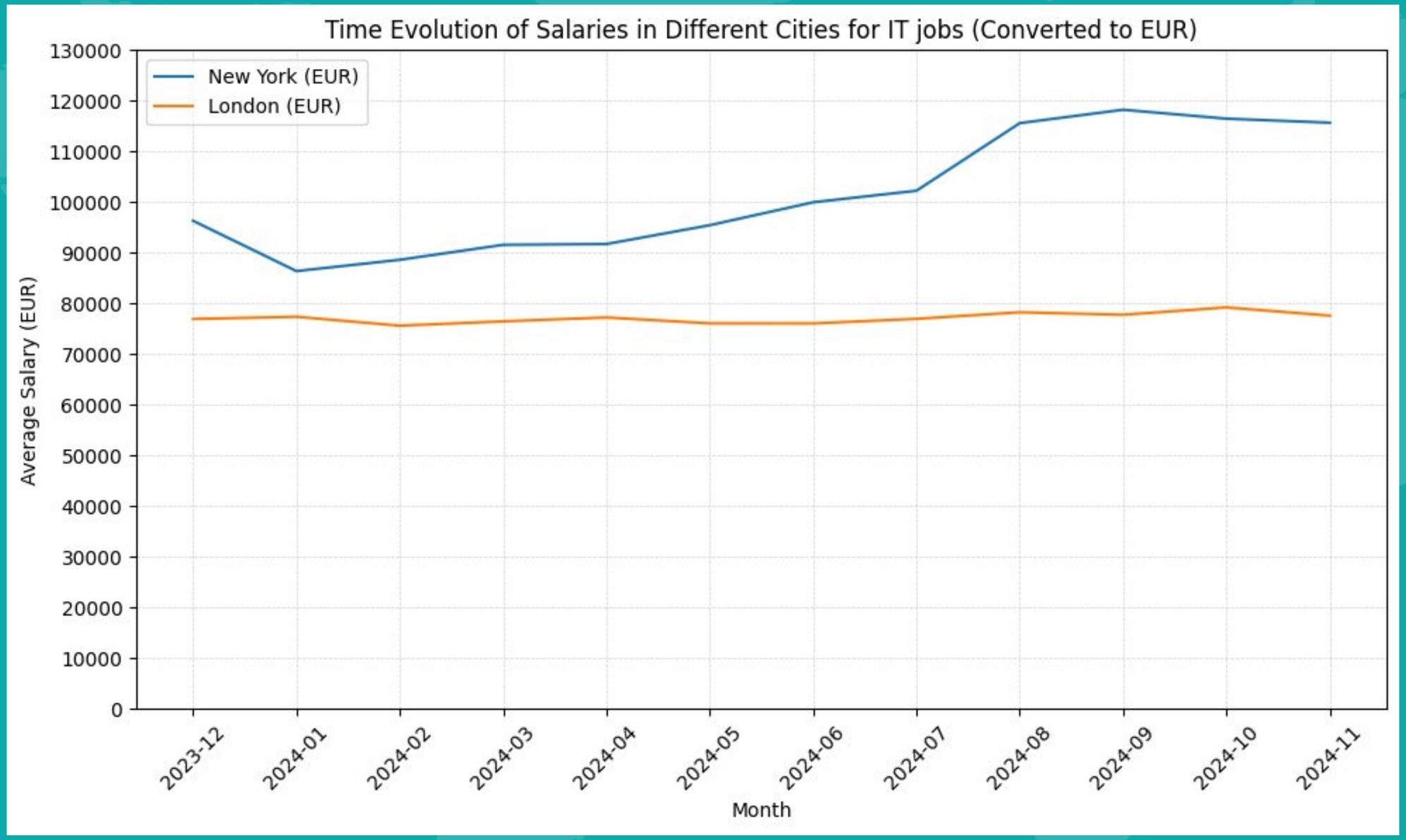


Data Visualization

- We also created a plot that represents the time evolution of salaries for a given job category and some given location

For example: it-jobs in London and New York

Data Visualization



Did we meet our goals?

→ Yes, most of them.

- A platform that allows user to find the best job

- Provides useful information on jobs

- Sadly we don’t have accurate trends over time for a given/specific job or detailed feedback for applicants

Main Takeaways

- ➔ **The salary distribution varies a lot across cities**
  - We can see specific data trends for overall jobs and salaries, which is more clear compared to other state-of-the-art solutions (e.g., LinkedIn)
    - We also provide exact locations/geographic info
- ➔ **Importance of NLP for job matching**
- ➔ **The data gathered provides useful insights for everyday users**
  - See specific locations and salaries

Future Work

- **Expand data collection**
  - Expand to other countries around the world
  - Other languages besides English
- **Incorporate more data sources:**
  - Maybe incorporate data from different APIs
- **Machine learning**
  - Techniques such as more complex NLP
    - Personalized job recommendations given a specific prompt

Any questions?