Thomas Peray - Grégoire Cosne - Rémi Delord

Job recommendation report

1. Documentation

a. Algorithm

We use the NearestNeighbors algorithm used by the SkLearn library.

b. Architecture

Our project is splitted in different components.

The first one, `user_interface.py` handles the displaying of the user interface. It creates many frames and widgets provided by the CustomTKinter library. Clicking on the "Search" button launches the "search" method. It will retrieve the fields filled in by the user and use them to generate a result.

The second one is `data_proccessing.py`. This Python script is primarily used for data processing. It imports the csv module, which provides functionality to both read from and write to CSV files.

exp_to_range(exp: str) -> (int, int): This function takes a string exp representing experience, trims the last 6 characters, and splits the remaining string on " to ". It then converts the two resulting strings to integers, which represent the minimum and maximum experience range, and returns them as a tuple.

salary_to_range(salary: str) -> (int, int): This function works similarly to exp_to_range, but it processes salary data. It trims the first and last characters of the input string, splits the remaining string on "K-$", and converts the resulting strings to integers. These integers represent the minimum and maximum salary range, and are returned as a tuple.

genre_to_int(genre: str) -> int: This function takes a string genre and returns 0 if the genre is "Female" and 1 otherwise. This is a simple way to convert categorical data to numerical data, which can be useful for certain types of data analysis or machine learning.

load_data(filepath: str) -> (list, list): This function takes a file path to a CSV file, reads the file, and processes its contents. It initializes two lists, evidence and labels, to store processed data. For each row in the CSV file, it processes several fields using the previously defined functions and appends the processed data to the evidence list. It also appends the "Job Title" field to the labels list. The function finally returns the evidence and labels lists.

The load_data function is the main function that uses all the other helper functions to process the data. It reads the CSV file row by row and for each row, it processes the "Experience", "Qualifications", "Salary Range", "location", "Country", "Company Size", and "Preference" fields, and stores the processed data in a list. This list is then appended to the evidence list. The "Job Title" field is directly appended to the labels list. The function returns the evidence and labels lists, which can then be used for further data analysis or machine learning tasks.

The next one is `data_cleaner.py.` This Python script uses the pandas, re, and nltk libraries to clean and process a dataset of job profiles.

The script starts by importing necessary libraries. pandas is used for data manipulation and analysis, re is used for regular expression operations, and nltk (Natural Language Toolkit) is used for working with human language data.

The script downloads the English stopwords from nltk, which are common words that are usually filtered out in natural language processing tasks.

The script reads a CSV file named 'job_profiles.csv' into a pandas DataFrame.

Two conversion functions are defined: convert_salary and convert_size. These functions are used to clean and standardize the 'Salary Range' and 'Company Size' columns, respectively. They handle various formats of data and convert them into a consistent format.

The conversion functions are applied to the 'Salary Range' and 'Company Size' columns of the DataFrame using the apply method.

The 'Job Description' column is processed to remove stopwords and words of length 2 or less. The processed descriptions are stored in a new column 'Processed_JD'.

A list of selected columns is defined, and a new DataFrame is created with only these columns.

Any rows with null values in the 'Salary Range' column are dropped from the new DataFrame.

Finally, the cleaned and processed DataFrame is saved to a new CSV file named 'structured_data.csv'.

For the Machine learning file, we name it `job_recommender.py.` This Python script uses Natural Language Processing (NLP) techniques to recommend jobs based on user skill.

The script starts by importing necessary libraries. re is used for regular expression operations, ftfy for fixing text encoding issues, TfidfVectorizer from sklearn.feature_extraction.text for converting text data into a matrix of TF-IDF features, NearestNeighbors from sklearn.neighbors for unsupervised nearest neighbors learning, pandas for data manipulation and analysis, and stopwords from nltk.corpus for filtering out common words in natural language processing tasks.

The script reads a CSV file named 'structured_data.csv' into a pandas DataFrame.

The script defines stop words using the nltk library.

The script reads a resume text file and extracts information such as experience, qualifications, salary range, and skills using regular expressions. This information is stored in a dictionary and then converted into a DataFrame.

The script defines a function ngrams that processes a string by fixing text encoding issues, removing non-ascii characters, converting to lowercase, removing certain characters, replacing certain characters with others, normalizing case, removing multiple spaces, and generating n-grams. N-grams are contiguous sequences of n items from a given sample of text or speech.

The script uses the TfidfVectorizer to convert the resume text into a matrix of TF-IDF features. TF-IDF stands for Term Frequency-Inverse Document Frequency, a numerical statistic used to reflect how important a word is to a document in a collection or corpus.

The script fits a NearestNeighbors model to the TF-IDF matrix. This model will be used to find the nearest neighbors of a point in the dataset, which in this case is the TF-IDF matrix of the resume text.

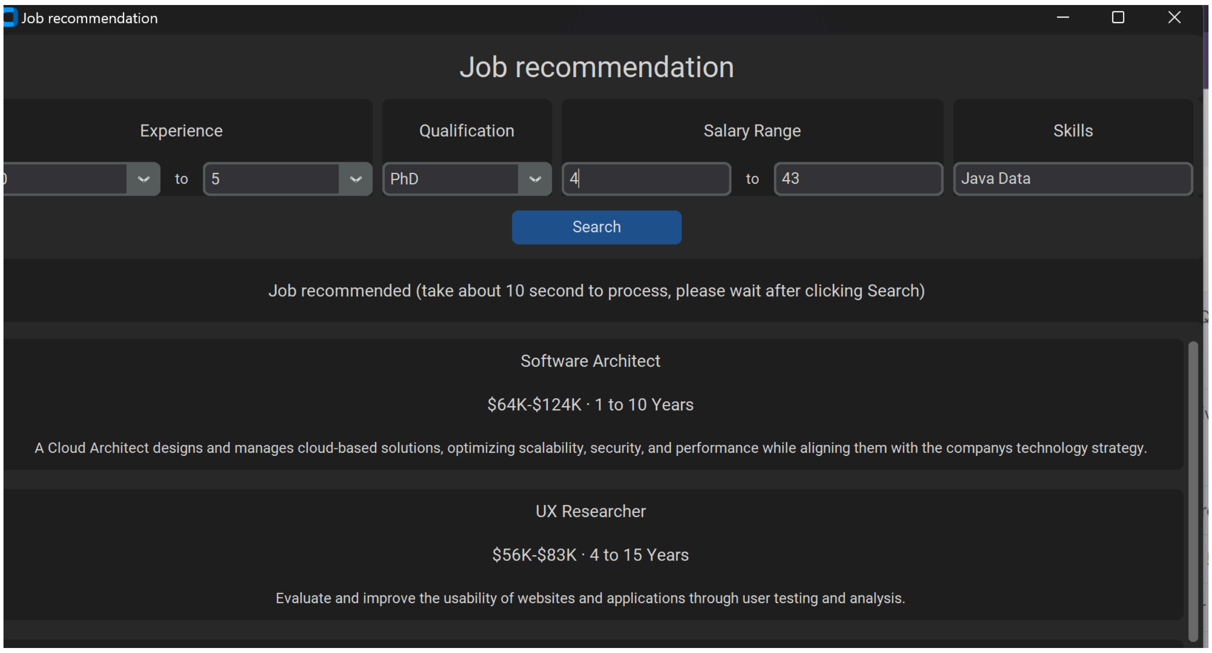
The script defines a function getNearestN that takes a query, transforms it into a TF-IDF matrix using the vectorizer, and returns the distances and indices of its nearest neighbors in the fitted model.

The script uses the getNearestN function to find the nearest neighbors of each job description in the dataset. It stores the distances and indices in two separate lists.

The script creates a DataFrame from the distances and adds it as a new column 'Match confidence' to the job descriptions DataFrame.

The script sorts the DataFrame by 'Match confidence' and prints the top 5 recommended jobs.

c. Results



Job recommendation

Job recommendation

Experience: 0 to 5

Qualification: PhD

Salary Range: 4 to 43

Skills: Java Data

Search

Job recommended (take about 10 second to process, please wait after clicking Search)

Software Architect

$64K-$124K · 1 to 10 Years

A Cloud Architect designs and manages cloud-based solutions, optimizing scalability, security, and performance while aligning them with the companys technology strategy.

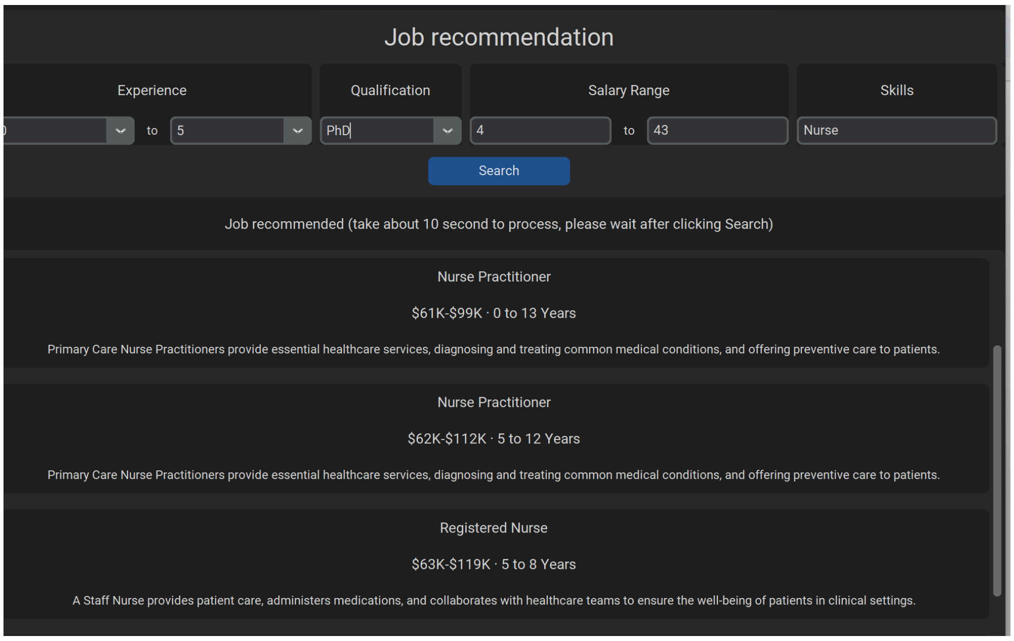
UX Researcher

$56K-$83K · 4 to 15 Years

Evaluate and improve the usability of websites and applications through user testing and analysis.

The results we got were based on different inputs, the most important was the "Skills" inputs, we have decided to take in account the experience and the qualification and also the salary range wanted in thousand euros.

For the outputs we have decided to just show the title of the job, his salary and a brief description of it.



Job recommendation

Experience: 0 to 5

Qualification: PhD

Salary Range: 4 to 43

Skills: Nurse

Search

Job recommended (take about 10 second to process, please wait after clicking Search)

Nurse Practitioner

$61K-$99K · 0 to 13 Years

Primary Care Nurse Practitioners provide essential healthcare services, diagnosing and treating common medical conditions, and offering preventive care to patients.

Nurse Practitioner

$62K-$112K · 5 to 12 Years

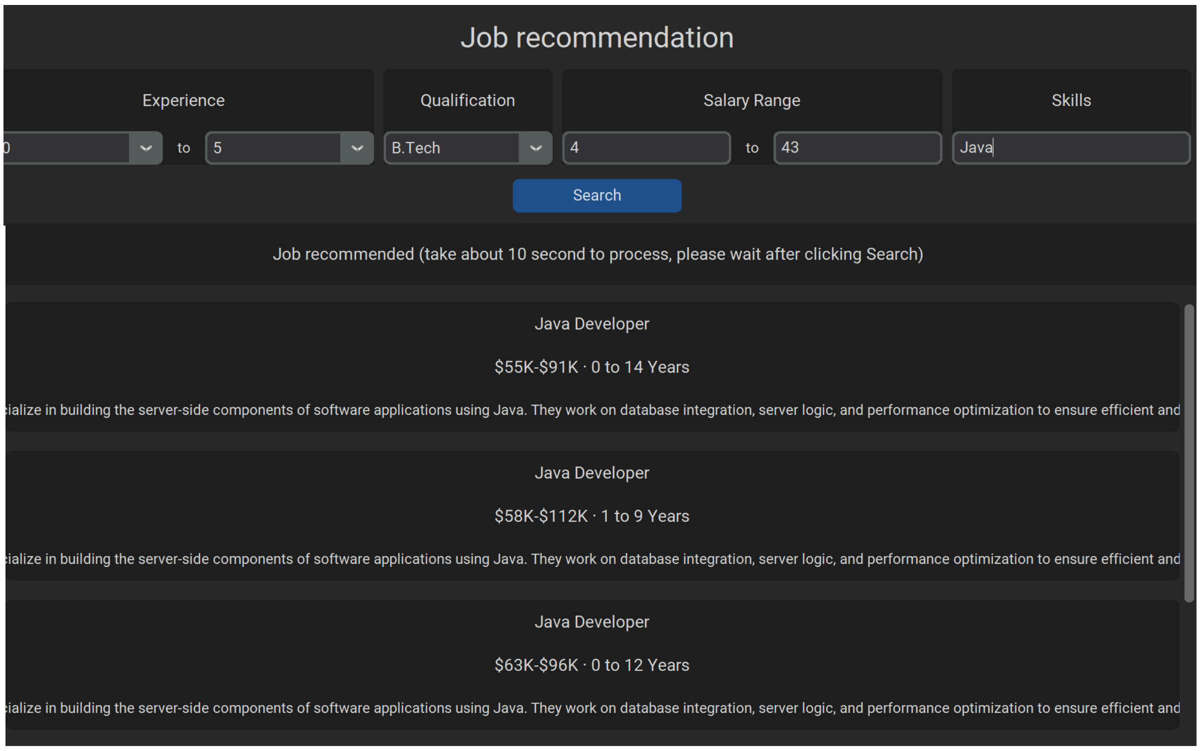
Primary Care Nurse Practitioners provide essential healthcare services, diagnosing and treating common medical conditions, and offering preventive care to patients.

Registered Nurse

$63K-$119K · 5 to 8 Years

A Staff Nurse provides patient care, administers medications, and collaborates with healthcare teams to ensure the well-being of patients in clinical settings.

The results show different offers for some same job but with different salary or experience.


Job recommendation

Experience: 0 to 5 · Qualification: B.Tech · Salary Range: 4 to 43 · Skills: Java

Search

Job recommended (take about 10 second to process, please wait after clicking Search)

Java Developer

$55K-$91K · 0 to 14 Years

ialize in building the server-side components of software applications using Java. They work on database integration, server logic, and performance optimization to ensure efficient and

Java Developer

$58K-$112K · 1 to 9 Years

ialize in building the server-side components of software applications using Java. They work on database integration, server logic, and performance optimization to ensure efficient and

Java Developer

$63K-$96K · 0 to 12 Years

ialize in building the server-side components of software applications using Java. They work on database integration, server logic, and performance optimization to ensure efficient and

2. Challenges and Improvements

Some challenges that we have faced were : check if the data we use were correct for building this job app. We used a Kaggle dataset and we were not sure about if it was containing enough data or columns that satisfy this task.

They were also checking the accuracy of the result, making the result keep sens about the skills and other information put in input and to check if the output was also correct and could match with the skills of the job candidate.

The csv chosen was really really big with more than 1 millions of rows so it takes time to download it and we must use a csv splitter on a website to get the final csv and use it.

Another challenge was to deal with some scikit-learn function like “TfidfVectorizer” because it was the first time that we used it and in the beginning we didn’t fully understand what she was doing for the ML part.

For the improvements, the system takes a bit of time to show the result of the query so it’ll be better to try to get the results faster. Sometimes for some niche job the results shown contain one or two jobs that weren’t linked to the skills due to the Kneirest-neigbhor algorithm and for the future it’ll be great to fix that and maybe limit the number of offers shown.

There is also that the skills are very important for the query but some inputs like the experience or the qualification are not taken into account much in the results and the latter really have a lot to do with the skills and the salary range.