**Aligning Education with Job Market Needs**

**Abstract**

In response to the growing challenge of preparing students for successful integration into the workforce, this project presents a data-driven approach by creating a cloud-based platform. The primary objective is to bridge the skills an knowledge that academic programs provide to students and the job market demands. By processing data extracted from different sources, including company websites and job posting platforms, the approach aims to provide valuable insights into the most in-demand skills and knowledge in the industry.

The hypotheses of this initiative propose that if educational institutions design their programs in harmony with the evolving demands of the job market, the employment rate of their graduates will increase. Additionally, regular updates from real-time job postings will make it easier for schools to design efficient curricula.

The methods used so far in the project include Natural Language Processing (NLP) for information extraction from job postings and machine learning to predict job titles based on job descriptions.

The implementation environment is currently the cloud-based platform Google Colab and the data set consists of data from job postings in the field of Data Analytics. Future work would attempt to expand both the data sets and the implementation environment.

**Idea and Approach**

From personal experience and other people’s stories, it was noticed that the academic programs of many schools have difficulties to adequately preparing students for the job market. This gives students a hard time finding a job after graduation as well as companies finding good candidates for employment. Motivated by the need for a better alignment between education and industry demands, this project aims to create a platform that fills in the gap between the two. The platform would pull real-time data from many job-posting websites and tentatively companies. The data would then be processed to extract information on which skills and knowledge are more in demand in the industry. The result is then shown to universities and schools in a clear and well-designed dashboard. This way the institutions can design their curricula in a way that will provide their students with preparation in those specific skills and knowledge. While this project is still new, not everything described in this introduction is achieved yet. However, the main idea of the project is implemented in a sample data set and more work can be done in the future to build a high-quality platform.

**Hypothesis:**

The project is driven by these hypotheses:

If universities align their academic programs with the skills and knowledge which are needed in the job market, then the number of job placements for their alumni will increase.

If educational institutions receive regular updates on the most in-demand skills from job postings, then they can design more effective academic programs which are aligned with the job market requirements.

**Data Set and Sources:**

For this project, the data set used so far was utilized from Kaggle, a known platform for datasets. The dataset is a CSV file which contains job postings in the field of Data Analytics. There are 36050 rows in the data each consisting of a job posting entry. There are 27 columns in the data which provide information about each job postings. Below is some metadata taken from The main columns the project was focused on are title, job description, and company.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The reason for narrowing the focus onto these columns is the current aim to find out which skills and job positions are more in demand from which companies. However, other columns like salary range and posted date would be insightful for future work as the project could expand into predicting which jobs would be more in demand in the future and which skills are the most paid ones.

**Methods , Algorithms, Results**

So far, the project has incorporated Natural Language Processing (NLP) techniques to extract valuable information from the job posting in the dataset. The NLP process involved tokenizing and analyzing the textual content of job descriptions. After breaking down the text into individual words and removing the stop words, we were left with a bunch of individual words that could or could not be name of skills/knowledge concepts. A list of known skills was then created. These skills were manually gathered from job postings and career platforms like LinkedIn and Indeed. The bunch of individual words for each job posting was then scanned to check which words it contained from the list of known skills. Below is the code used.

# Tokenize the job descriptions

df['description\_tokens'] = df['description'].apply(lambda x: word\_tokenize(str(x).lower()))

# Remove stop words and non-alphabetic tokens

stop\_words = set(stopwords.words('english'))

df['description\_tokens'] = df['description\_tokens'].apply(lambda x: [word for word in x if word.isalpha() and word not in stop\_words])

# Display the first few rows to check the result

df[['description', 'description\_tokens']].head()

The result of tokenization step is a data frame which contains each job description along with the tokenized version of the description itself but without the stop words. It looks as follows:

A screenshot of a computer

Description automatically generated

The code that was used to create the known skills list is the following:

# List of known skills

known\_skills = ["python", "java", "data analysis", "sql", "excel", "machine learning", "statistics", "r", "snowflake", "quantitative analysis", "data pipelines", "data infrastructure", "hadoop", "SQL", "data mining", "data lakes", "pandas", "numpy", "matplotlib", "seaborn", "rstudio", "scikit-learn", "tensorflow", "pytorch", "XGBoost", "LightGBM","Hypothesis Testing", "Apache", "spark", "apache kafka", "noSQL", "Tableau", "Power BI","NLTK", "SpaCy", "Text Mining","AWS", "Azure", "Google Cloud Platform", "Data Governance", "Ethical Considerations", "Domain Knowledge", "Communication Skills", "GitHub", "Git", "Docker", "MySQL", "PostgreSQL", "MongoDB", "Feature Engineering", "A/B Testing", "Problem-Solving", "SAP", "ERP systems", "Demand Planning", "Forecasting ", "Supply Chain", "Logistics", "ML/AI prediction", "Text Analytics", "D3.js", "scikit-learn", "Data Visualisation", "Matplotlib", "Seaborn","RDBMS", "operational data store", "ODS", "data marts", "data lakes", "MongoDB", "data ETL", "Data extraction", "C#", "XML", "OMS / EMS data models", "Agile", "Google Kubernetes Engine", "GKE", "Cloud Storage", "HTML", "CSS", "Cascading Style Sheets", "JavaScript", "Slack", "Figma", "Google Meeet", "Excel", "AI/ML projects", "AI tools", "Deep Learning", "standard operating procedures", "SPOs", "Google workbooks", "Google App Script"]

The list can be extended and if the project is launched, it can be developed to add to the list as soon as new skills and techniques arise in the industry.

After extension, the dataframe contains a column which stores the extracted skills of each job description. Scanning for skills and adding the extracted skills in each row was done in one step using a “for” loop:

# Loop through each row and tokenize the words to check for known skills

for index, row in df.iterrows():

    description\_tokens = row['description\_tokens']

    if description\_tokens:

        extracted\_skills = [skill for skill in known\_skills if skill in description\_tokens]

        df.at[index, 'extracted\_skills'] = extracted\_skills

Using pyplot from matplolib, the top 15 most common skills in the dataset were plotted for a clear visualization. The result was the following with SQL, Excel, and Python leading the top skills required:

A graph with colorful bars

Description automatically generated

It is interesting to note that Excel is one of the top tools required by employers as so many complex new tools have been created for Data Analytics. This is very useful information for curriculum designers in universities. Even though they see that new tools are coming up, it is good to make sure that their students know the basic tool which is being used in the field (no matter how old it might be).

Another Data Science implementation used so far in the project is machine learning to make predictions. A model was built to predict job titles from mentioned skills. 20% of the data was used for testing (No adjustments have been made so far, but there is room to do so in order to improve the model in the future). Skills (words describing skills) were converted into numerical features for the model training step using CountVectorizer:

#Use CountVectorizer to convert 'skills' into numerical features

from sklearn.feature\_extraction.text import CountVectorizer

A RandomForestClassifier machine learning model was built and then used to make predictions on job titles if a job description was given. The accuracy of the current model is not high, but adjustments can be made to improve accuracy. Here are some of the predictions versus the actual data.

A screenshot of a computer

Description automatically generated

Calculated accuracy is as follows:

A screenshot of a computer

Description automatically generated

Moreover, other models can be used in the future for different predictions. A useful prediction would be to use large data sets to find out the skills that are going to be in demand in the future two or three years. This is because graduate students usually need two or three years to complete their program. Academic institutions might design better curricula if they consider the fact that the job market is very dynamic, and it might change from the time when students start a program until they finish.

**Review of Approach:**

In the diagram that follows, the main idea and process that would be applied is given. The diagram shows an ideal case in addition to what was already achieved at this stage of the project. More information on what could be done in the future is given in the next section as well.

**Future Work**

The project has a lot of work to be done in the future. A method to automatically extract information from job posting and company websites should be found. The method should include a way to feed the data directly into the cleaning and preprocessing steps. It is also important to consider what rules and policies would be applied to get permission from all the different resources to collect data from them. Some methods to achieve this might be using Wb Scraping to extract information or checking with job posting platforms like Linked and Indeed if they provide an API for data retrieval.

It is important to have a method to classify job skills in the future as we would not deal with the Data Analytics field only, but with many types of industries. To achieve this, we can train a machine learning model to classify job postings into different industries. Algorithms like Support Vector Machines might be useful on this step.

Another area of the project that requires a lot of attention is building a dashboard. The Dashboard needs to be built in a cloud-based environment which is accessible by many educational institutions. If successful, the dashboard could be made available to students as well who might want to view the dashboard on their phone. Cloud environment platforms to choose from might be Amazon Web Services, Google Cloud Platform or Microsoft Azure. Tableau, Power BI, or Google Data Studio can be used to build the Dashboard. AWS CloudWatch Events can be used to schedule tasks to regularly update the dashboard as new job postings appear.

Another aspect that needs to be considered in the future is monitoring and management of the database and the platform. Ways to collect feedback from users can also be implemented. This way the platform can continuously be improved based on the users’ needs.

**References**

1. ChatGPT AI , 12/19/2023 Version 3.5
2. Google Bard AI, 12/19/2023
3. Data Science Handbook, A Practical Approach, by Kolla Bhanu Prakash, ISBN 978-1-119-85733-4
4. Python for Data Science Handbook by Jake VanderPlas, 2nd Edition. ISBN 9781098121211
5. Cognitive Class.(2023). Data Visualization with Python. Cognitive Class <https://apps.cognitiveclass.ai/learning/course/course-v1:CognitiveClass+DV0101EN+v2/home>