MIE 1624 Introduction to Data Science and Analytics – Fall 2024

Assignment 3

Due Date: 11:59pm, December 5th, 2024
Submit via Quercus

Introduction:

For this assignment, you are responsible for answering the questions below based on the dataset provided. You will then need to submit a 3-page report in which you present the results of your analysis. In your report, you should use visual forms to present your results. How you decide to present your results (i.e., with tables/plots/etc.) is up to you but your choice should make the results of your analysis clear and obvious. In your report, you will need to explain what you have used to arrive at the answer to the research question and why it was appropriate for the data/question. You must interpret your final results in the context of the dataset for your problem.

Background:

Data science, analytics, AI, big data are becoming widely used in many fields, that leads to the ever-increasing demand of data analysts, data scientists, ML engineers, managers of analytics and other data professionals. Due to that, data science education is now a hot topic for educators and entrepreneurs.

In this assignment, you will need to design a course curriculum for a new "Master of Business and Management in Data Science and Artificial Intelligence" program at University of Toronto with focus not only on technical but also on business and soft skills. Your curriculum would need to contain optimal courses (and topics covered in each course) for students to obtain necessary technical and business skills to pursue a successful career as data scientist, analytics and data manager, data analyst, business analyst, AI system designer, etc. You are required to extract skills that are in demand in the job market from job vacancies posted on Indeed.com web-portal and apply clustering algorithms to group/segment skills into courses.

You are provided with a sample Python code to web-scrape job postings from Indeed.com web-portal, that you would need to modify for your assignment. You can decide on the **geographical locations** of the **job postings** (e.g., Canada, USA, Canada and USA) and job roles (e.g., "data scientist", "data analyst", "manager of analytics", "director of analytics") of the posting that you will be web-scraping, but your dataset should contain at least 1000 unique job postings. You can use a combination of locations and job titles as search parameters in developing your dataset.

Experiment with different Natural Language Processing (NLP) algorithms to extract skills (features) from the web-scarped job postings. You may manually define your own list of keywords/key-phrases (N-grams) that represent skills, e.g., "Python", "R", "deep learning", "problem solving", "communications", "teamwork", or use pre-trained NLP algorithms that automatically extract skills/features from your dataset.

You will need to use OpenAI's ChatGPT API in order to have a list of skills automatically generated as well as interpretation of your final results.

Finally, you will need to use skills extracted from job postings as features and run two clustering algorithms to create clusters of skills that can be interpreted as courses. First clustering algorithm that you are required to use is **hierarchical clustering algorithm with one feature**, where that feature represents a distance between each pair of skills, known as a distance matrix. The idea is that if a pair of skills is found together in many job postings, those two skills would be required together on the job, and it makes sense to teach those skills (topics) together within the same course (cluster). Using this idea, you will need to define your own distance measure, create a dendrogram (see slides 43-44 of "Lecture 8 – Advanced Machine Learning" for an example), and interpret each cluster as a course. For the second clustering algorithm you can **choose between k-means and DBSCAN**. You will be required to use **at least 10 features as inputs for your second clustering algorithms.**

Based on your first and second clustering analysis separately, create a sequence of **8-12 courses. For each course include 3 or more topics (based on skills) that should be taught in each course.** You may find that you need to manually adjust your clusters to form 8-12 clusters of 3 or more skills/topics, this is fine, but you must explain your reasoning for the adjustments you choose to make. Please list your courses in a logical order, i.e., a course that requires another course as a pre-requisite should be listed after the pre-requisite course. You can use your own judgement for deciding about a logical sequence of courses or try to interpret your clustering results for that. For visualizing your course curriculum, feel free to use Python or any other software like Tableau and Power BI. As a bonus, you are asked to develop another course curriculum by creating text embeddings of your dataset using OpenAI GPT Embeddings and clustering the embeddings using a model of your choice to form another course curriculum.

Learning objectives:

1. Understand how to clean and prepare data for machine learning, including transforming unstructured web-scaped data into structured data for analysis. Convert categorical features into numerical features and perform data standardization/normalization, if necessary, prior to modeling.
2. Understand how to apply unsupervised machine learning algorithms (clustering) to the task of grouping similar items.
3. Interpret your modeling results and visualize those.
4. Improve on skill and competencies required to collate and present domain specific, evidence-based insights.

Questions:

The following sections should be included but the order does not need to be followed. The discussion for each section is included in that section's marks.

1. **[1 pt] Data collection and cleaning:**
    a) Adapt provided web-scraping code:
        i. Decide on geographical location and job role/title.
        ii. Scrape Indeed job postings for data.
        Note: you have been provided a dataset of 1000 unique postings, if you choose to use the given dataset you will forfeit the 1 mark for this step.

2. **[3 pts] Data processing, feature engineering, and visualization:**
    a) Engineer features for clustering analysis and visualization:
        i. Define skills by combining a list of skills you come up with from your own knowledge/research with skills generated by ChatGPT API. You will need to acquire your own API key from Open AI.
        ii. Extract skills using N-grams or pre-trained NLP algorithms.
    b) Visualize key information:
        i. Generate at least four visual depictions of the information you've collected and/or features you've engineered to describe your data.

3. **[3 pts] Hierarchical clustering implementation:**
    a) Implement **hierarchical clustering** algorithm:
        i. Generate a distance matrix to describe the relationship between skills.
        ii. Perform hierarchical clustering of skills.
        iii. Generate a **dendrogram** and interpret it to develop a course curriculum (8-12 courses with at least 3 skills/topics covered in each). You may need to make manual adjustments to your clusters that differ from what you get from your dendrogram; reasoning for any adjustments must be explained. Repeating skills across separate courses should be avoided.

4. **[4 pts] K-means or DBSCAN clustering implementation:**
    a) Implement **k-means clustering** algorithm or **DBSCAN clustering** algorithm:
        i. Engineer 10 or more unique features to describe each skill for clustering (e.g., skill frequency, average salary for skill, etc.). You will be penalized for overly simplistic or redundant features.
    b) Include visualization of **elbow method** used to determine the optimal *k* number of clusters for **k-means clustering** or *eps* value if using **DBSCAN clustering**.
    c) Develop a course curriculum based on clustering results (8-12 courses with 3 or more skills/topics covered in each). You may need to make manual adjustments to your clusters that differ from the optimal *k* value; reasoning for any adjustments must be explained. Repeating skills across separate courses should be avoided.

d) Generate a labeled scatterplot from **k-means clustering** algorithm (with optimal *k* number of clusters determined by elbow method) or **DBSCAN clustering** algorithm:
    i. To generate a scatterplot you will first need to perform dimensionality reduction via the method of your choice (e.g., t-SNE, PCA).

5. [2 pts] Interpretation of results using ChatGPT API:

a) Decide on a final course curriculum and feed it into ChatGPT API to expand on your clustering results:
    i. Write a prompt to have ChatGPT API write a short description of your course curriculum that should entice potential students to enrol in this program.
    ii. Write a second prompt of your choice that adds to your overall analysis (e.g., have ChatGPT describe the similarities within clusters).

6. [2 pts] Discussion and final course curriculum:

Discuss your results. Present and justify your final course curriculum. You may select curriculum from Section 3 (hierarchical clustering algorithm) or from Section 4 (second clustering algorithm) as your final course curriculum. Insufficient discussion will lead to the deduction of marks.

7. [+1 bonus pt] OpenAI to describe clustering results:

First create text embeddings of the job descriptions using OpenAI GPT Embeddings, then use the embeddings to perform clustering using the algorithm of your choice. Use these clustering results to develop an additional set of course curriculum.

Or

Develop an ensemble method to combine your clustering results from the hierarchical and *k*-means or DBSCAN algorithms.

Note: your max assignment mark cannot exceed 15 pts including bonus.

Submission:

1) Produce an IPython Notebook (.ipynb file) detailing the analysis you performed to answer the questions based on your data set. Your .csv file containing the results of your webscraping must also be submitted. Your webscraping code does not need to be submitted unless you've made changes to the code beyond your "location" and "job title" parameters.

2) Produce a 3-page report explaining your response to each question for your data set and detailing the analysis you performed. When writing the report, make sure to explain for each step, what you are doing, why it is important, and the pros and cons of that approach. **You may choose include an appendix for your figures, ChatGPT outputs, and bonus question material. To receive full marks, all figures (including elbow curve) must be contained in the report, be labeled, have figure captions, and should**

large enough they can be easily read and interpreted. Your report should include a detailed description of your engineered features and how they were calculated, clustering algorithm implementation and results, any manual adjustments you made to your clustering results, interpretation of elbow curve, etc.

Tools:

- **Software:**
  - **Python Version 3.X** is required for this assignment. Make sure that your Jupyter notebook runs on Google Colab (https://colab.research.google.com) portal. All libraries are allowed but here is a list of the major libraries you might consider: Numpy, Scipy, Sklearn, Matplotlib, Pandas, NLTK, OpenAI.
  - No other tool or software besides Python and its component libraries can be used to collect your data and touch the data files. For instance, using Microsoft Excel to clean the data is not allowed. Please dump your web-scraping results into a file, submit it with the assignment, and comment your web-scraping code in the notebook.
  - Upload the required data file to your notebook on Google Colab – for example,

```
from google.colab import files
uploaded = files.upload()
```

  - You are allowed to use any software for visualizing your course curricula (Tableau, Power BI), but you should use Python for everything else.

- **Required data files to be submitted:**
  - **webscraping_results_assignmnet3.csv**: file to be submitted with the assignment.
  - The notebook will be run using the local version of this data file. Do not save anything to file within the notebook and read it back.

- Auxiliary files:
  - **Indeed_webscraping.ipynb**: Jupyter notebook used to web-scrape job postings from Indeed web-portal. Please modify this code for your own needs.
  - **Assignment_3_Template.ipynb**: this code will help you get started with NLP feature extraction, OpenAI API implementation, and clustering.
  - In order to run the Indeed webscraping script you need to download a local version of ChromeDriver so that Selenium WebDriver can use Google Chrome to launch executables through your Jupyter notebook. You can download the correct version of ChromeDriver for your sister from ChromeDriver Github Page. If you choose to use a Gecko based browser, such as Firefox, you may need to acquire a local version of this API through Geckodriver's Github Page but the current Firefox version of the webscraping code is set up so that downloading Geckodriver should not be necessary.

What to submit:

1. Submit via Quercus a Jupyter (IPython) notebook containing your implementation and motivation for all the steps of the analysis with the following naming convention:

   lastname_studentnumber_assignment3.ipynb

   Make sure that you **comment** your code appropriately and describe **each step** in sufficient detail. Respect the above convention when naming your file, making sure that all letters are lowercase and underscores are used as shown. **A program that cannot be evaluated because it varies from specifications will receive zero marks.**
2. Before submitting your Jupyter notebook please remove your personal API key.
3. Submit via Quercus a csv file with your web-scraping results with the following naming convention:

   webscraping_results_assignment3.csv
4. Submit a report in PDF (up to 3 pages) including the findings from your analysis. Use the following naming conventions **lastname_studentnumber_assignment3.pdf**.
5. If you make any significant changes to the webscraping code please upload it as a separate file or include it as a function in your main notebook but comment out the function call.

Late submissions will receive a standard penalty:

- up to one hour late - no penalty
- one day late - 15% penalty
- two days late - 30% penalty
- three days late - 45% penalty
- more than three days late - 0 mark

Other requirements and tips:

1. A large portion of marks are allocated to analysis and justification. Full marks will not be given for the code alone.
2. Output must be shown and readable in the notebook. The only file that can be read into the notebook is the file **webscraping_results_assignmnet3.csv** with your web-scraping results.
3. Ensure the code runs in full before submitting. Open the code in Google Colab and navigate to Runtime -> Restart runtime and Run **all** Cells. Ensure that there are no errors.
4. You have a lot of freedom with how you want to approach each step and with whatever library or function you want to use. As open-ended as the problem seems, the emphasis of the assignment is for you to be able to ***explain the reasoning behind every step***.
5. Webscraping can sometimes have spontaneous issue due to internet firewalls and other uncontrollable factors. To avoid complications, it is suggested that you try to complete this step early on.