CS 410: Final Project Documentation

Intelligent Course Explorer Extension for Enhanced Academic Understanding

Team Goat (Fall 2023)
Hao Huang, haoh11
Kaiyuan Wang, kw22

1. Overview

Our project creates a Chrome Extension specifically used for the Course Explorer website of UIUC. When the extension is being used, it provides user with a few keywords extracted from the topic and the course descriptions, along with relevant wikipedia links users can use to learn more about sophisticated terminologies. It is helpful for students who wants to explore new classes yet having trouble to understand new terminologies included in the course description.

2. NLP Implementation

- The overall structure of the extension uses javascript with a manifest.json file.
  - Content.js file reads only the topic and description of the course explorer webpage, then sift out irrelevant information such as prerequisite and credit hours.
  - The trimmed topic and description is then being sent to our local flask server by making AJAX calls. This API endpoint will run our python code that heavily uses NLP packages.
  - The popup.html and popup.js is used to design the UI when the extension is being used. It shows the keywords extracted from the content and relevant wikipedia links.
- NLP word analysis functions in ‘app.py’ file.
  - ‘extract_keywords’ function: this function inputs the course title, description, and number of keywords that need to be generated. It uses nlp model to analyze the title and description. The function extracts noun chunks from the description. Noun chunks are phrases that contain a noun and words describing the noun. The extraction excludes any noun chunks that are already present in the title, ensuring uniqueness and relevance to the description. For each noun chunk in the description, the function calculates a similarity score with the title. This score indicates how closely related the chunk is to the overall context of the title. The similarity is determined using the similarity method, which likely compares semantic similarity between the title and each noun chunk. The extracted noun chunks are sorted based on their similarity scores in descending order. This means the most relevant chunks to the title are considered first. The function

returns the top num_keywords noun chunks as the final set of keywords.

    - ‘extract_keywords_with_spacy’ function does a similar thing, but use another approach. The function uses spacy.load("en_core_web_md") to load a medium-sized English language model for Spacy. This model is well-suited for many general-purpose NLP tasks.
        - phrases: Extracts noun chunks from the description, which are phrases that contain a noun and words describing the noun.
        - entities: Extracts named entities recognized by Spacy, such as names of people, places, organizations, etc.
        - combined_terms: A list combining both phrases and entities for a comprehensive set of potential keywords.
    - ‘is_valid_wikipedia_page’ function validates whether the Wikipedia page exists.
    - ‘query_wikipedia’ function returns the Wikipedia links for the extracted keywords

3. Usage Instructions

1. Git clone the whole repository into a local file from: https://github.com/jackhaohuang/CS410-Final---GOAT/tree/main
2. Install all necessary packages in app.py
3. cd into CS410-Final–GOAT folder, then run command python app.py or python3 app.py to start the flask application
4. Open a chrome tab and go to chrome://extensions/
5. Open the developer mode and click “Load Unpacked”
6. Choose the whole CS410-Final–GOAT folder
7. Navigate to a Course Explorer course webpage https://courses.illinois.edu/schedule/2024/spring/ and click on the extension

4. Contributions

- Hao Huang (haoh11)
    - Implemented the Chrome Extension and the flask server
        - created the content.js and popup.js
        - constructed app.py flask server to run python code and communicate with the extension
- Kaiyuan Wang (kw22)
    - Implemented the Keyword Extraction Python code, including:
        - Two keywords extraction function to analyze the course title and description
        - Link extracted to Wikipedia API