Smart Video Search

Diana Arita, Kartik Patel, Matthew Kryczka, Selvaganapathy Thirugnanam

Problem

There is no easy way to search by video content. This makes it hard for users who are interested in finding precise locations of the videos where a concept or a term is mentioned.

Abstract

Build a search engine that can support video segment search for Coursera lecture videos with transcripts. This would allow a user to type in a query and see a ranked list of short video segments so that the user can precisely locate which segment to watch in a lecture in order to know more about a concept. We intend to build the search engine based on the scrapping/indexing/ranking concepts learnt from this course. The end solution will provide a User Interface for the user to type in the query and to view the results as links to the video segments.

Overview

1. Scrape the videos along with the transcripts (Coursera dl)
2. Extract documents out of scrapped videos and transcripts and build an association between them
3. Use Tokenizer(Stemming and other normalization techniques) to extract lexical units (words)
4. Use an Indexer (inverted Index) for faster response from the Search Engine.
5. Perform Ranking/Scoring based on Probabilistic retrieval functions (Eg: BM25)

Software

- Code base: https://github.com/Kartikp2/CourseProject.git
- Coursera dl: https://github.com/coursera-dl/coursera-dl
- Libraries
  - Metapy: https://github.com/meta-toolkit/metapy

How to run!

- Download code from github: https://github.com/Kartikp2/CourseProject.git
- Run backend
  - Be in the directory! (same as server.py)
  - To install the dependencies - npm install
  - Run server.py
- Run Frontend
  - Be in the directory (search-ui -> smart-video-search)
  - To install the dependencies - npm install
  - To run app - npm start
  - Go to browser and hit url https://localhost:3000

DEMO!

Implementation

Data

- Use coursera-dl package to scrape coursera video files and transcripts
    - Initial_analysis -> coursera-dl ->cs-410
- Extract the srt files
    - coursera_video_lessons.csv
    - coursera_video_segments.csv
    - Segment_extractor.py
- Build thumbnails for video using timestamps

Implementation

Search Engine

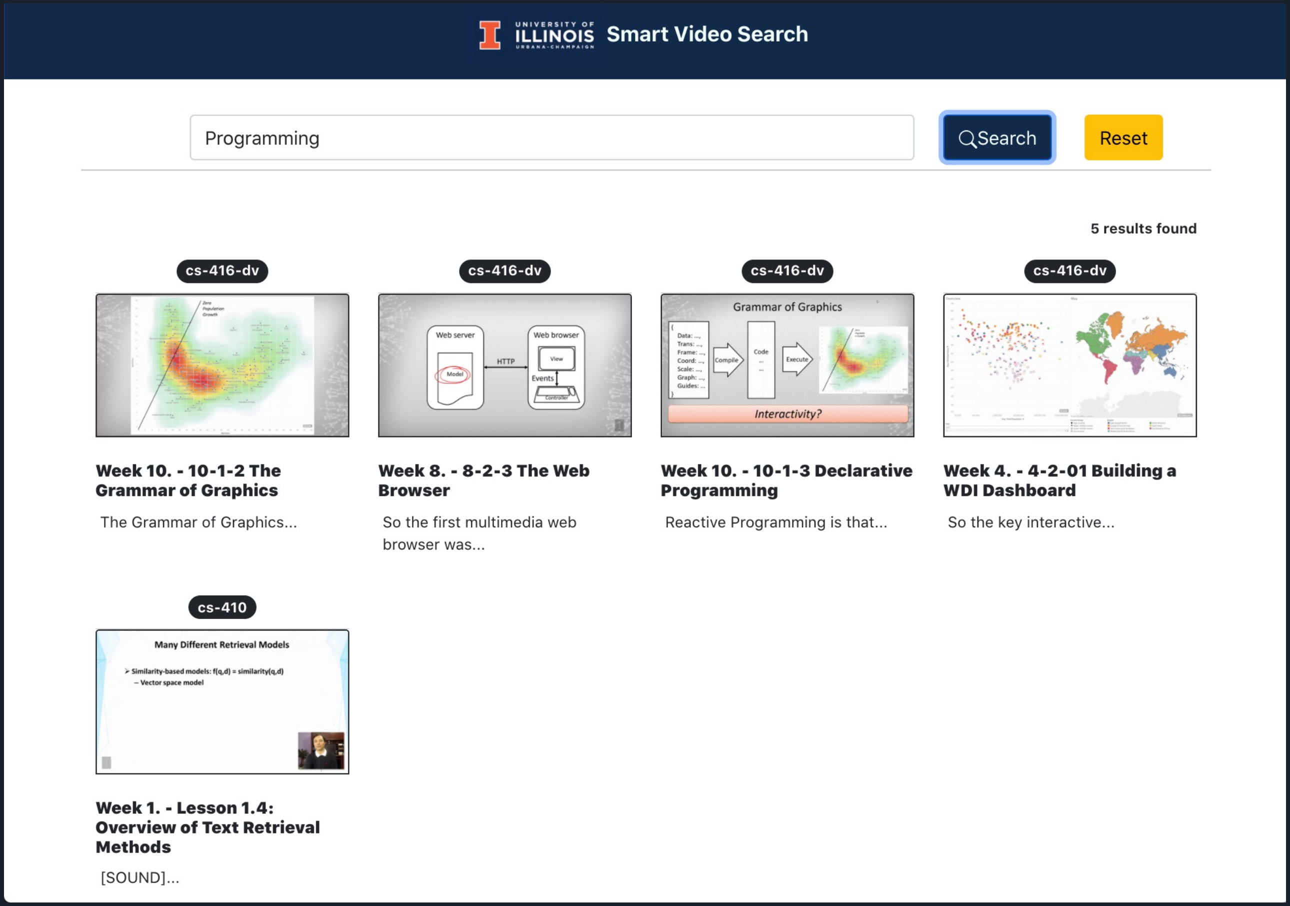
- Build corpus using coursera_video_lesson.csv
- Create config.toml
- Build Inverted index
  - metpay.index.make_inverted_index
- Build Ranker
  - Using OkapiBM25
- Score the query
- Return results (based on course, week, lesson)

Implementation

Server

- Fetch query from frontend
- Use Ranker to retrieve the most relevant video segments for the query
- Try to find the first video segment from the Ranker that contains the query text, if not found then use the first segment
- For each segment, get more details
    - { course ID, week number, video title, text preview, link, image path }
- Return results to frontend!

DEMO!



Smart Video Search

Programming

5 results found

cs-416-dv

Week 10. - 10-1-2 The Grammar of Graphics

The Grammar of Graphics...

cs-416-dv

Week 8. - 8-2-3 The Web Browser

So the first multimedia web browser was...

cs-416-dv

Week 10. - 10-1-3 Declarative Programming

Reactive Programming is that...

cs-416-dv

Week 4. - 4-2-01 Building a WDI Dashboard

So the key interactive...

cs-410

Week 1. - Lesson 1.4: Overview of Text Retrieval Methods

[SOUND]...

Further Improvements!

- Give user control of parameters!
    - Ranker
    - Number of Documents
- Add more courses!