Search Technologies

Video Summarization

Niall Bermingham
Vincent Achukwu
Alan Hian Wu

Introduction

Background:

- The application will be a client server based that will provide the user the ability to search through and find relevant videos with the feature to also search through the video content.

Our Approach:

- We will first categorize the video that will be uploaded to the server, for this we use keyframes of the video and use a ML model to categorise it.
- The information we retrieved from categorisation will go into a database to be used as the bases of our video search engine.

User Analysis

The User:

- The user should be seeking for videos or uploading a video
- Not required to have prior specialized knowledge
- To have minimum technical knowledge such as uploading a video or typing in search queries
- Target audience is any private individual

Using The System:

- The user provides Search queries. The more specific the query the more relevant videos will be given to the user to browse and choose from, for less specific and general queries our system will use a mix of popularity, viewer attention and basic search term match to try and retrieve relevant videos for the user.

Scientific Functional Description

Overall architecture

- Web app (front and back end)
- HTML5, CSS, Node.js for frontend, Python backend, MySQL for database

Detection Algorithms

- Keyframes (image recognition per frame, OCR), done by chopping up videos by factor of 10, minimum requirement of videos on our system is 24 frames, and a maxim of 120, and using ML for image recognition and OCR.
- Speech recognition (speech-to-text, captions), powered by a open source speech to text engine called DeepSpeech
- Storing and Retrieval of Video, raw videos are stored in servers and video indexes and metadata is stored in MySQL, our video retrieval algorithm is inspired by the paper *"Content based video retrieval systems"* by **B V Patel, B B Meshram**
- Summarization of Video, Video Summarization will be done by a Long Short-Term Memory ML model based on a paper in the Book *"ECCV 2016"* written by **Ke Zhang, Wei-Lun Chao, Fei Sha,** and **Kristen Grauman**
- Video chapters (timestamps of relevant query), using keyframes and speech recognition, we'll split the video into chapters for the user to browse

Scientific Functional Description

Pros:

- Accuracy of ML
- Search video based on content via keyframes
- Video can be quickly searched due to how we preprocessed it

Cons:

- Storing videos
- Efficiency
- Video processing speed

Scientific Functional Description

Limitations and Assumptions in our design

- Resolution of the video will cause inconsistencies
- Audio and video synchronization will cause problems (eg. Timestamps)
- Video type such as structured vs unstructured
- Video effects will cause issues with identifying keyframes
- Language inconsistencies

Evaluation Strategy

- **Evaluation objective**
  - To find problems with system and determine the relevancy of the results
- **Data used**
  - public videos
- **Search queries used for functionality assessment**
  - Comma separated words, short sentences
- **How are we going to collect data? Testing**
  - Users
  - Us
  - online video database
- **What metrics are we going to use to evaluate the performance?**
  - Response time
  - Video rank relevancy
  - Accuracy of keyframe search
  - Audience retention and click rate for success measurement

Conclusion

- System will enable user to search for a video or in a video
  - Keyframe
  - Speech recognition
  - Video Summarization
  - Storing and Retrieval of Video
- User require minimum technical knowledge to use the system
- The system uses a client-server based architecture
- It will be evaluated with users queries that should match related sections of video returned by the system