Team 17 Project Charter
Annix

Team Members: Ethan Ling, Leo Gu, Matthew Sigit, Nick Song

Project Title: Annix, an easy way to recognize any character you find online

1. Problem Statement

Facial recognition is a popular and effective tool employed by applications and search engines to fill many needs. However, current models for facial recognition have significantly lower efficacy when applied to drawn or animated formats. As a result, existing services, such as google lens or other competitors, often fail to answer the frequently asked question of "who is this character?". And while there exists proofs of concept that attempt to solve this problem, no such technology exists on the current market that is widely available to users. Our solution to this problem is Annix, a multi platform service, which aims to use Machine Learning and Convolutional Neural Networks trained specifically on animated characters to fulfill this need.

2. Project Objectives

Machine Learning Model:

1. **Develop a system to efficiently obtain, label, and deliver triplet image data to our machine learning model.** This will involve web-scraping for images and developing a system for developers/volunteers to label said images.

2. **Produce a high efficacy image recognition model trained from the above data set that is able to discern between a large set of characters**. We aim to use the Triplet-Loss function, in which two similar and one different image is fed into a convolutional neural network (CNN) in order to train it to discern similarity.

3. **Design a structure for labeled data to be sent from a DBMS to our model which is not time or space intensive.**

Multi-platform Application:

4. Design and Implement user-friendly architecture that stores user-provided data and allows users to navigate between all provided services listed below.

5. Create functionality for users to send image requests to the model and receive a response in a timely manner.

6. Integrate a data analytics engine to allow for users to see commonly searched characters and also help with determining any flaws with the system.

7. Allow users to create and delete accounts associated with data such as but not limited to profile details, previous searches and (potentially) recommendations.

8. Efficiently use APIs to implement features that answer associated questions, such as “Where should I watch this show?” or “Who is the original artist?”.

3. Stakeholders

Users: People looking to identify characters.
Developers: Ethan Ling, Leo Gu, Matthew Sigit, Nick Song
Project Manager: Matthew Sigit
Project Owners: Ethan Ling, Leo Gu, Matthew Sigit, Nick Song

4. Deliverables

1. A Tensorflow and Keras image recognition model hosted on an AWS EC2
2. A Flask and MongoDB or any alternative DBMS backend to run our multi platform web application with a CDN through AWS Route 53
3. A ReactJS frontend that hosts features such as a profile system and data analytics pulled our backend that also routes to API related features
4. A S3 Bucket which stores thousands of images that were either user inserted or webscraped that can easily be referenced
5. A secure user permissions system that allows some users to label triplets, administrate, and other access permissions

5. Frameworks and Tools

- Tensorflow and Keras to implement an image recognition model.

- ReactJS (Front-end), Flask and MongoDB (Back-end) to implement the web application.

- Amazon Web Services. More specifically:
  – AWS EC2 for hosting machine learning models.
  – AWS Route 53 to support continuous delivery for our web application.
  – AWS S3 Bucket to store thousands of images in an easily referenceable area.