Gideon Blinick

Capstone 2 Proposal: Movie Clusterer/Recommender

1. **What is the problem you want to solve?**

The movie recommendation problem. We want to create a recommendation system for movies on IMDb where we are given a movie and are able to recommend the closest movie to that movie based on movie features (length, genre, runtime, etc.) and plot (text).

2. **Is it a real problem that someone cares about? Who is your client and why do they care about this problem? In other words, what will your client do or decide based on your analysis that they wouldn't have done otherwise?**

Yes, people care about this problem.

The client is anyone that enjoys watching movies and is looking for another movie to see. They would be able to enter a movie that they have seen that they enjoy and get back a recommendation for a movie similar to it.

With this analysis, my client will be able to watch a movie that they previously would not have known to watch, or at least had to spend more time and effort discovering.

3. **What data are you using? Is there real-data available? How will you acquire the data? Is the data easy to acquire and clean?**

I will be using real movie data from IMDb. Data will be sourced from IMDb public datasets (https://www.imdb.com/interfaces/). Any data missing from the datasets, such as plot synopses info, will be sourced from IMDb using the IMDbPY API (https://imdbpy.readthedocs.io/en/latest/index.html).

The data shouldn't be too hard to obtain. On the other hand, it's definitely harder to get than a Kaggle or UCI dataset.

4. **Briefly outline how you'll solve the problem.**

Modelling: Build the recommendation engine and use NLP techniques to take advantage of the plot text that we have available; use different metrics for similarity such as pairwise cosine similarity. Among the NLP tools used will be the "Semantic Similarity with TF-Hub Universal Encoder" and word2vec.

(If Have time): Deploy as a web service.

Problem in Brief:

a. **Is this a supervised or unsupervised problem?** Neither. It's a recommendation problem. This quora answer defines it as an information retravel problem: https://www.quora.com/Where-do-recommender-systems-fall-in-machine-learning-approaches
b. **What variable are you trying to predict?** We're not trying to predict, but recommend a movie given a movie.
c. **What variables will you use as predictors?** Movie features such as ratings, genres, year-of-release and plot synopsis
d. **What will be your training data?** A subset of the 10,000 movies we pull from IMDb

5. **What are your deliverables?** Code, paper, and a slide deck