Machine Learning Engineer Nanodegree
Capstone Proposal
Tyler Shumaker
June 3, 2017

Proposal

Domain Background

I am currently working on a project to develop a proof of concept implementation of intelligent virtual agents that will provide virtual personal assistant, recommendations, and product socializing capabilities to an user. The users will interact with the intelligent agents via chat, using an open source webchat platform as the interface to the chatbot.
 One of the jobs being performed in the background by the intelligent agents is to provide recommendations to the users. We intend to recommend chat rooms to users they should consider joining.
 * Details of the project are intentionally vague in this proposal because the work is on-going.

Problem Statement

A common problem for our clients is that user's spend a lot of time finding a subject-matter expert (SME) or relevant information to assist them complete their tasks. Based upon the user's interaction with the chatbot and other users we want to be able to recommend chat rooms where other users are working on similar topics. So that the users can trust in the intelligent agent's recommendations a degree of explainability needs to be provided. (Example: Chat Room #34 is recommended because users are also discussing X)

Datasets and Inputs

The Cornell Movie-Dialog Corpus [1] will be used because it is a collection of conversations extracted movie scripts. The dataset consist of 220,579 conversational exchanges between 10,292 pairs of movie characters involves 9,035 characters from 617 movies. Each character can be treated as a user in a chat platform and each movie can be related to chat rooms or channels.
The movie scripts were gathered from publicly available scripts that are referenced in the raw_scripts_urls file. The metadata for the movies were gathered from IMDB.

The corpus consist of a movie_titles_metadata file that provides; movieID, movie title, movie year, IMDB rating, no. IMDB votes and genres. In our client's application the chat rooms can be associated with topics of interests which can be resented by the movie's genres.
The movie_characters_metatadata, movie_lines and movie_conversations files can be can be used to construct the conversations between movie characters.

Solution Statement

A content-based recommender will be created to provide recommend movies (chat rooms) to character (users). Using TF-IDF (Term Frequency - Inverse Document Frequency) to parse to all of the conversations for each character.
Cosine similarity will then be used to identify which characters discussing similar topics. [2] The movies that the most similar characters are from will be recommended to the character predicted.

Benchmark Model

It maybe difficult to create a benchmark model because of processing conversational data from different movies may not result in a clear model. Also, dialogs between movie characters are not truthful representations of real-life conversations. They often are “too carefully polished, too rhythmically balanced, too self-consciously artful” (Kozloff, 2000), due to practical and artistic constraints and scriptwriting practice (McKee,1999). [3] A

character from a movie should return positive results for the movie which the character is from. To create a benchmark a sample of characters from particular movies will be used.

Evaluation Metrics

The evaluation metrics for this model will be use Root Mean Squared Error (RMSE) [4] to compare random test characters to a set of character from movies used for benchmarking.

Project Design

The first step for this project will be creating the conversations for each character from the Cornell Movie-Dialog Corpus. All of the conversations each character has will then be parsed using TF-IDF that will create a matrix of unigrams, bigrams, and trigrams for each conversation. The similarity between all conversations will be computed using cosine similarity. A number of similar conversations and their scores will then be stored for each conversation. The similar conversations (characters from conversation and movie the conversation took place in) and scores will then be retrieved for each character.

References

¹Danescu-Niculescu-Mizil, C. (2011). Cornell Movie--Dialogs Corpus. Retrieved July 6, 2017, from http://www.cs.cornell.edu/~cristian//Cornell_Movie-Dialogs_Corpus.html

² Clark, C. (2016, June 09). A Simple Content-Based Recommendation Engine in Python. Retrieved July 6, 2017, from http://blog.untrod.com/2016/06/simple-similar-products-recommendation-engine-in-python.html

³Danescu-Niculescu-Mizil, C., & Lee, L. (2011). Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL.

⁴A. (2017, May 31). Root Mean Squared Error (RMSE). Retrieved July 6, 2017, from https://www.kaggle.com/wiki/RootMeanSquaredError