Abstract

There are many different recommendation systems out there for tv shows, movies, music,books, and other forms of entertainment. Most book recommendations appear to be based on ‘what other buyers bought’ (e.g., Amazon), or which book a particular bookstore employee liked in a given month. This project seeks to create a book recommendation system that recognizes some overlap with a title, or author, or even a categorical or descriptive input and then return a book recommendation that is based on either the provided criteria, or even a user-selected criteria from a list of accepted features. A Hugging Face model was used on a Kaggle book dataset to create a basic book recommendation system where a reader can find a new book based on a book’s title, subtitle, author, category, or description.

Data Wrangling and Exploration

A dataset of books, including title, ratings, average ratings, descriptions, and covers, was downloaded from Kaggle, and one key libary was used for the LLM itself: Hugging Face, specifically the MiniLM L6 v2 SentenceTransformer model.

After removing any unusable data (i.e., mainly rows with missing data), a table with 2182 books remained from an original list of 6810 titles. As this is a proof-of-concept recommendation system (and one that could not handle nan’s), it was more straightforward to remove entries with missing data.

Model and Results

The MiniLM L6 v2 SentenceTransformer model was built upon the pretrained nreimers/MiniLM-L6-H384-uncased model. It was then trained on a one billion sentence pairs dataset, the top source being Reddit comments, with almost 0.75 billion training tuples! This is a good model for the book recommendation system because it is meant for only short sentences - any inputs longer than 256 words is truncated.

The features listed in Table 1 were encoded using the model, and then tested by using sentences or phrases to manually test the book recommendations the model pulled from the book data frame. SKLearn’s cosine similarity functionality was used to determine the ‘closeness’ of the user’s queries to the model’s encodings. By default the model will return the top 5 closest book matches (again, based on the user’s desired feature), but this number is easily changed in the recommendation query.

Conclusions

Hugging Face provides a number of pre-trained models, and the MiniLM L6 v2 SentenceTransformer was extremely easy and intuitive to use from the documentation. For future iterations, it may be interesting to experiment with other forms of distance other than cosine similarity between the queries. Next steps will be to build out a basic web interface on Streamlit for the user to easily get a book recommendation based on the features available. Perhaps integrating image processing to return slightly more novel recommendations based on book covers would also be a fun feature. Recommendation systems need a facelift to give the user a more flexible and imaginative recommendation based on their interests. This recommendation system begins the steps

Appendix: Model Metrics and Data

Table 1: Features

Feature	Data Type
title	object
subtitle	object
categories	object
description	object
authors	object

Table 2: Book Data

Book Attribute	Data Type
isbn13	int64
isbn10	object
title	object
subtitle	object
authors	object
categories	object
thumbnail	object
description	object
published_year	float64
average_rating	float64
num_pages	float64
ratings_count	float64