Name: Alfred Gonsalves
ASU ID: 1211150145

CSE 515: Multimedia and Web Databases Project Phase 1 Report

Group Members: (Group 15)

Xiangyu Guo

Siddhant Tanpure

Chenchu Gowtham Yerrapothu

Alfred Gonsalves

Greggory Scherer

Abstract:

This report describes an information retrieval technique by implementing vector models and graph models on data. The data considered here is the MoviesLens and IMDB data consisting of actors, movies, and users. The information retrieved in the model is mapped to a weighted tag vector which is calculated based on the tags associated with the movies. The two models considered for mapping are Term Frequency (TF) model and Inverse Document Frequency (IDF) model. These models present information about the tags which have a high discriminant power in the document. Three difference models are also created based on comparing two genres that print the TF-IDF-DIFF which is a distance metrics. P-DIFF1 and P-DIFF2 models are also used for differentiating two genres based on a weight factor involving a probabilistic feedback mechanism.

Keywords:

Term frequency (TF), Inverse document frequency (IDF), Term Frequency – Inverse Document Frequency (TF-IDF), Vector model, Graph model, TF-IDF-DIFF, P-DIFF1, P-DIFF2

Name: Alfred Gonsalves
ASU ID: 1211150145

I. Introduction

a. Terminology

The report consists of various terminologies describing topics in information retrieval.

i. Term Frequency
The number of times a term occurs in a document is called its term frequency.

ii. Inverse Document Frequency
The inverse document frequency diminishes the weight of terms that occur very frequently in the document set and increases the weight of terms that occur rarely.

iii. Term Frequency – Inverse Docsument Frequency
It is a numerical statistic, or a weighting factor, that is intended to reflect how important a word is to a document in a collection or corpus

iv. Vector space
It is a representation of all objects in a given N dimensional space.

v. Probabilistic feedback mechanism
This mechanism provides a weighting factor by considering the relevant and irrelevant aspects of a feature in the document.

b. Goal Description

The goal of this project is to learn how to model a given feature into a vector space by retrieving relevant information about the objects present in the document. The project aims at creating three vector models for actor, user and genre for a given movie dataset. These models retrieve relevant details about the feature and generate the weighted tag vector. This tag vector helps to provide information about the discriminant power present with a tag and how well it describes a given document. The tool would also provide a differentiating tag vector model with TF-IDF-DIFF, which will calculate the difference among the data points.

Name: Alfred Gonsalves
ASU ID: 1211150145

c. Assumptions

There are various assumptions considered while implementing the project.

i. For calculating the actor vector model, we consider the actors from the imdb-actor-info table rather than movie-actor table. We assumed that even though the actor is not associated with any movie, his/her contribution will be important when considering the entire document.

ii. The total genres are calculated by mapping movies from mltags to mlmovies and calculating the genres as they come.

iii. The users list for generating user vector model are considered from mltags and mlratings for considering the movies "watched" by a user and not just movies that are simply tagged.

iv. For calculating the differentiating tag vector for TF-IDF-DIFF, we consider the movies associated with genre1 and genre2, and calculate IDF based on the set of movies and not on genres.

II. Proposed Solution/ Implementation

This solution was implemented in Java by using MySQL database to store the data from CSV files. The user interface was developed to take the vector model to be displayed as input along with the objects and the model to be used.

The project phase was split in four tasks. Every task needed a separate model to be generated. The program would take this as input and generate the required model. The output of every model was a sorted tag vector in the most significant to least significant order. For TF-IDF-DIFF the output would be a weight distance between the data point of the tag.

Name: Alfred Gonsalves
ASU ID: 1211150145

III. Interface Specification

The program is user interactive. It asks the user to enter commands as follows:

- print_actor_vector actorID model
- print_genre_vector genreID model
- print_user_vector userID model
- differentiate_genre genre1 genre2 model

The model can be TF or TF-IDF for the first 3 models. For the differentiator model we have TF-IDF-DIFF, P-DIFF1, P-DIFF2.

The model will display the tag vector with weights in a sorted manner.

IV. System Requirements

Since this is a Java application, it would run on any machine/OS with Java installed.

The database is a MySQL database and hence will need a JDBC connector which is included in the project zip. It also uses an external jar for DateUtil.

The user has to directly execute the ModelExecuter.java main file in the main package.

V. Conclusions

We conclude that by using proper weight function, we can generate easily generate a tag vector that clearly shows which tag has a higher discriminating power than other tags. This tag can be used to learn more about the document. The differentiating vector helps the user understand the difference in any two genres and how closely they are related to each other.

Name: Alfred Gonsalves
ASU ID: 1211150145

VI. Appendix

a. Specific roles of group members

Xiangyu Guo,Group discussion, Independent implementation

Siddhant Tanpure, Group discussion, Independent implementation

Chenchu Gowtham Yerrapothu,Group discussion, Independent implementation

Alfred Gonsalyes,Group discussion, Independent implementation

Greggory Scherer, Group discussion, Independent implementation