Sarah Martin

Professor Lance Levenson

Northwestern University, School of Professional Studies, Foundations of Data Engineering

February 16, 2019

Project 1: Business Processes and Requirements Definition

for a Movie Recommender Engine

**Section 1: Management Overview and Recommendations**

The task at hand is to define the business processes and high-level system requirement for a new movie recommender engine. (vs. Systems Design and Specification). This is: Functions, performance, qualities, and constraints – the “what” to guide the design. Not, Architecture and the technical “how” to meet the requirements and to guide the builders or coders

A data scientist has a new business/product opportunity. Inspired by neural network encoding techniques, the data scientist has devised a way to represent feature length movies by vectors of numbers. The method draws data from the entire Internet Movie Database (IMDb) from Amazon, as well as open-source movie ratings from MovieLens.org. With movie embeddings and preference rankings of twenty or more movies from an individual reviewer, the data scientist can generate personalized movie recommendations that are more accurate than those from other recommendation engines.

Provide a general description of the recommended solution in terms that management can understand. Functional requirements definition for a working prototype:

**Section 2: Data and Database Software**

The movie recommender engine currently requires input data from three sources:

1. The Internet Movie Database (IMDb) owned by Amazon
2. MovieLens.org, a research site hosted by GroupLens Research, a group of undergraduate students, graduate students, staff, and visitors, engaged in social computing research at the University of Minnesota, and headed by faculty in the Department of Computer Science and Engineering.
3. Users of the recommender engine themselves

Section 2.1: IMDb Data

While IMDb does not provide an API for public queries, it offers seven daily updated downloadable compressed tab-separated UTF-8 encoded plain text files that contain much of the data found on IMDb via this link: <https://datasets.imdbws.com/>. This data is listed as available for personal and non-commercial uses only with their Non-Commercial Licensing compliance specifics available in here. Research will be required immediately to determine if we can publish modeling results based on the data, and if we ultimately intend to monetize our recommender engine product, we should first determine the viability of Commercial licensing with IMDb. If Commercial licensing is an option, technical and legal resources will be required to communicate and coordinate with IMDb to develop and maintain a functioning relationship ensuring continued use of their data.

IMDb lists three main options for accessing the information contained in these files: command-line interface tools, a Java-based GUI that allows searches and displays of the information provided, and a Python package, “IMDbPY”, that can process the files into a number of different SQL-based databases.

Our system will require loading the IMDb data into the analytics tooling we choose for Neural Network encoding. We will also need to extensively investigate, explore, manipulate, and potentially alter the structure or encoding of the data. In addition, ideally, the system would allow us to refer back to any previous version of the data at any time. As such we should probably store the data, and all or most previous versions of it, in house. Thus, an initial assessment suggests the IMDbPY package may be appropriate for accessing the data and feeding it into a SQL-based database and/or data warehouse.

Our system should also allow for regularly scheduled pulls of new versions of the files on IMDb and batch processing to load them into the database and/or data warehousing solutions we choose. The appropriate cadence of these data pulls will be highly dependent on the performance of the recommender results, both in terms of model accuracy and in terms of user affinity. As such, the precise cadence of IMDb data acquisition may change over time and should be reassessed periodically leveraging input from the modelers as well as business decision makers.

A technical resource or a team of technical resources should be put in charge of developing, executing, and monitoring data acquisition, basic pre-processing, and batch loading into our database and/or data warehouse systems. This resource or team must have open lines of communication with those coordinating any contract relationship with IMDb *and* with those tasked with developing, monitoring, and updating the neural network modeling in case of deviations in the source data.

Section 2.2: MovieLens.org Data

GroupLens offers downloads of 7 UTF-8 encoded csv structured zipped dataset files sampling various MovieLens.org data. Some of these datasets are static and others seems to be periodically, though infrequently, updated. Each dataset comes with a README file outlining appropriate use cases of the file. Specifically, “the user may not use this information for any commercial or revenue-bearing purposes without first obtaining permission from a faculty member of the GroupLens Research Project at the University of Minnesota.”[cite] As such, if we ultimately intend to monetize our recommender engine product, research will be required immediately to assess the viability of continued use of the MovieLens.org data. If viable, resources will again be required to communicate and coordinate with GroupLens to develop and maintain a functioning relationship ensuring continued use of their data.

Similar to the requirements necessitated by incorporation of the IMDb data, our data-flow system must be able to unpack the MovieLens.org files, load them into our own databases and/or data warehouses, and load them into any and all analytics tooling we choose for data exploration, Neural Network encoding and prediction generation.

Again, we likely want our system to allow for periodic pulls of new versions of the some of the files provided by GroupLens and batch processing to load them into the database and/or data warehousing solutions we choose. However, in this case, we can likely perform these data acquisition tasks at more infrequent intervals as compared with the IMDb data acquisition schedule. A technical resource or a team of technical resources should be in charge of developing, executing, and monitoring MovieLens.org data acquisition, basic pre-processing, and batch loading into our database and/or data warehouse systems. This resource or team could easily be the same resource/team handling the IMDb data acquisition and also, should probably have direct lines of communication with technical resources at GroupLens. Given GroupLens status as a small research organization versus IMDb as a subsidiary to a company founded by someone understood to be the richest man in the world, it is likely the relationship with GroupLens will be far less demanding to cultivate and monitor than that with IMDb.

Section 2.3: User Input Data

With movie embeddings and preference rankings of twenty or more movies from an individual reviewer, the data scientist can generate personalized movie recommendations

What should the system do? How can it best be defined to serve movie consumers? How shall data be stored, searched, and processed?

Section 2.4: Storage Systems

Specify data sources and how data will be acquired and maintained.

Note database systems that will likely be used for implementing the application or system.

Justify data and database design decisions.

**Analytics and Modeling Software**

Specify analytics and modeling software that will be needed for development, testing, and implementation.

Justify analytics and modeling design decisions.

**Computing and Communications Systems**

Specify the computing infrastructure, resource requirements (in terms of processing power and memory?)

information systems and connections between systems.

Justify information infrastructure design decisions.

Works Cited

A works cited page beginning on a separate page at the end of the paper. Ensure that all resource materials as properly cited, with APA or Chicago style references.

*Designing Data-Intensive Applications:*

Chapter 1: Foundations of Data Systems (pages 3–25),

Chapter 2: Data Models and Query Languages (pages 27–67)

Chapter 3: Storage and Retrieval (pages 69–107).

<https://reqexperts.com/resources/requirements-articles/articles-what-is-the-difference/>

<https://grouplens.org/>

<https://help.imdb.com/article/imdb/general-information/can-i-use-imdb-data-in-my-software/G5JTRESSHJBBHTGX?pf_rd_m=A2FGELUUNOQJNL&pf_rd_p=3aefe545-f8d3-4562-976a-e5eb47d1bb18&pf_rd_r=A4RHXCC91B96D7392C05&pf_rd_s=center-1&pf_rd_t=60601&pf_rd_i=interfaces&ref_=fea_mn_lk1#>

<https://www.imdb.com/licensing/?ref_=helpms_ih_gi_license>

<https://grouplens.org/datasets/movielens/>

<http://files.grouplens.org/datasets/movielens/ml-latest-small-README.html>