Recommender System(Review based)

EE 660 Project Type: Individual

Author: Ruchin Patel,

Email: [ruchinpa@usc.edu](mailto:ruchinpa@usc.edu)

USC ID: 9520665364

Phone number: 4322662949

Date: 3rd Dec 2018

**Please organize your report along the lines of this template**; you may use any word processing software you like, as long as you submit your report as the required pdf file described below.

**Your report must be typewritten and submitted as a pdf document**, in machine readable form (no scans or screen shots).

**Please submit your code as a second pdf file**, all code in the one file, also required to be machine readable (no scans or screenshots).

# Abstract

A brief, informative description of your project. Include the problem, approach (naming the machine learning methods you used in your project), and key results.

The abstract should be considered a “stand alone” section – it should be understandable on its own, and includes only information that is described (and supported) elsewhere in the report.

Tip: many people find it works better to write the abstract last, even though it will be read first.

# Introduction

# Problem Type, Statement and Goals

**Problem Type:** Classification + Sentiment Analysis + Recommendation

**Typical Systems:** Typical recommender systems perform collaborative filtering technique which uses “User Behavior” for recommending items. The “User Behavior” here is nothing but the purchasing pattern of every user. It is the most commonly used technique in the industry as it not dependent on any other additional information. However every or most of the recommender systems tend to use the final ratings to find the correlation or similarity between different users and thereby finally recommending items to different users.

**My Approach:**

A brief description of the problem that you are trying to solve, and/or the goals you want to achieve. Clearly state the type of your problem (classification or regression) and which classes or variable(s) you are trying to predict, and any other goals your project might have. Explain why the problem is important or interesting, and why it’s not trivial.

Example sources of difficulty (nontriviality) include:

1. A physical model that’s inherently complicated and hard to abstract.
2. High dimensionality of feature space.
3. Sparsity.
4. Nonlinear behaviors.
5. Limited number of training samples.
6. Significant amounts of preprocessing required.

# Literature Review (Optional)

Briefly describe existing approaches to your problem. The literature review doesn’t need to be exhaustive, but it should cover well known publications, so you are aware of their approach, tools and results.

# Prior and Related Work (Mandatory)

If this project is an extension of some work you previously did or are currently doing outside of EE 660, briefly summarize this other work, and clearly distinguish it from your EE 660 project work.

If you have no prior or related work, state so (*e.g.*, this can be done as a single heading: “Prior and Related Work - None”).

# Overview of Approach

Give an overview of the models and algorithms you used, how they were compared, performance metrics used, and any other key aspects of your project you would like to highlight. (Note that detailed descriptions will be given in Seciton 3, the next section, below.)

# Implementation

Report your implementation details and results in the following subsections. You should mention which libraries and functions you used but avoid including code in your report. Your description of what your system does should be readable and understandable to a reader that isn’t familiar with the functions and libraries you used, but is familiar with the algorithms and techniques that were covered in EE 660. (For example, stating “we standardized all real-valued features, and recast all categorical features using one-hot encoding” and also stating the functions used in your code for this, is fine; stating only the functions used in your code is not fine.)

## Data Set

Describe the dataset you used. Explain the meaning (if known) and data type (integer, real, string or binary, categorical, etc.) of each input variable and output variable. Give the number of data points, number of input variables, and number of output variables.

Provide a table with each feature's name, its type (categorical, real...), the cardinality (for categorical) or range and brief description if the name does not make it obvious (for example, a feature named “month" probably requires no description). If there are too many features, try to group them or give some general idea of what they describe. If this is not possible (example: features are given with names such as *feat1, feat2,* etc.) state so.

## Preprocessing, Feature Extraction, Dimensionality Adjustment

Describe in detail the pre-processing and feature extraction techniques you used. If you used any dimensionality reduction or sparse coding methods, explain in here as well. If the dataset has missing data, explain how you dealt with it (removing samples, removing features or data filling) and justify.

If you used different pre-processing for different machine learning methods, or if you tested the same machine learning method with different pre-processed inputs, state so. A table can be useful in these cases.

## Dataset Methodology

Describe the procedure you followed in the use of your dataset.

You should clearly state how many data points were used for pre-training (not mandatory), training, validation set(s) (or specify k if k-fold validation is used), and testing.

Describe clearly how validation sets were used, and where in the process they were separated from training sets. For cross validation, describe where in the process the cross validation loops were implemented; and if multiple cross validation loops were used, state whether they were nested or sequential, and their ordering. Describe where in the process the validation results were used to make decisions. You may find it useful to use flow charts or diagrams to illustrate your dataset methodology.

Also describe where in the process the test set was used, any decisions made based on the test set results, and how many times the test set was used.

## Training Process

Describe how you trained your model, the classifiers or regression processes you used, and the parameters you chose.

For each machine learning method or model you use:

* Describe your model and your algorithm in detail (with formulas and flowcharts). It generally isn’t necessary to repeat equations given in EE 660 just to describe the model and algorithm; however, you must give enough information to clearly define which model and algorithm (and which version of the model and algorithm) you are using. Also, if you want to refer to any equations (e.g., for your interpretation or analysis), you must include those equations in your report. If needed, also explain in detail what you did to adapt the method to your case, and explain the assumptions or “tricks” you used.
* Give some justification for why you chose this method. This can be a simple statement such as: simple method for baseline; non-linear method because problem seems complex; we believe this method would yield the best performance; etc.
* State the parameters of the model and how they were chosen. If a parameter is chosen by heuristics, state so. If a parameter is chosen by some model selection or validation process, state so and describe the details in Sec. 3.3 or 3.5.
* Analyze the complexity of your hypothesis set to the extent possible. Compare with the number of data points you have and the dimension of the pre-processed feature space. Explain what you did to avoid overfitting and underfitting.
* If you have sets of results to show for this machine learning method, include them here. (For a comparison of results from *different* machine learning methods, use the next subsection.)

## Model Selection and Comparison of Results

If you had multiple potential models in the beginning, explain how you performed model selection. No need to repeat what was covered in the Dataset Methodology subsection above.

Present performance comparison of your different models and methods here. Be sure to clearly show what dataset each result is from (training, validation, test if any, averages over multiple cross-validation runs, etc.). Use table(s) and/or plots. **Do not paste print screen images.**

Is the difference between these results as expected? In the case of classification, you can choose two salient features and plot your decision boundaries w.r.t. them. In the case of regression, you can plot the resulting regression function in 3D (as a function of two salient features) or in a few 2D plots (each as a function of one salient feature).

# Final Results and Interpretation

Document your final results. Describe your final system and its parameter values. Give the final performance of your system(s) and an estimate of it’s out of sample performance. Compare with your baseline results and with any results you found in the literature. Include figures, plots, and/or tables if appropriate.

**If you are working on an online competition, report the performance of your best submission and compare it to others on the leader board.** If you want to compare your results with other work, do so here.

Interpretation: Why do you think the results came out the way they did? What has been learned from them? Anything particularly noteworthy or unexpected? Showing your understanding of your work and results is an important part of this project. Note: you can interpret results throughout the report, but this section should contain a final interpretation (e.g., why this system worked the best, and how much better it is or is not, and what else might improve it further)

# Contributions of each team member

If a team project, state here what the contribution of each team member was (i.e., who did what).

# Summary and conclusions

Briefly summarize key findings, and optionally state what would be interesting or useful to do next.

# References

Cite the sources of your information that came from elsewhere. This includes other related works you compare with (if any), and sources of descriptions of systems or methods that you included in your report.

# Your code

**Please submit your code in a separate file.**  All your code should be in one pdf file, machine readable (no screen shots or scans).