Mining Media Data I Winter Semester 25/26

Assignment-01

Prof. Dr. Rafet Sifa Armin Berger Dr. Lorenz Sparrenberg

04.11.2025

Introduction

In this assignment, we will investigate methods to mine frequent itemsets, implement the Apriori and the rule extraction algorithms, and apply the last to analyze user movie preferences. Finally, we will continue our story part by conducting further analysis on the dataset we started analyzing in the previous assignment. All the programming assignments will be implemented using the programming language Python 3.10+. This assignment has 30 Pts. in total.

1 Implementation of the Apriori and the rule extractor algorithms (8 Pts.)

In this part, using python, we will implement the Apriori algorithm to extract frequent itemsets and the algorithm for extracting association rules from frequent itemsets (see lecture slides or the data mining book). We will initially test the algorithm on our dataset from Table 1 by extracting the frequent itemsets for $S_{\min} = 0.1$ and association rules for $C_{\min} = 0.3$. For each rule also calculate the Lift value.

Table 1: A transactional DB T for Part 1

Player ID	**Bought Items**
Player 1	Elixir, Shield
Player 2	Gem, Shield
Player 3	Elixir, Sword
Player 4	Elixir, Wand, Shield
Player 5	Elixir, Sword
Player 6	Wand, Shield
Player 7	Elixir, Wand, Shield, Sword
Player 8	Elixir, Giant Wand

Visualize your results using matplotlib's scatter plots with x- and y-axes being the confidence and support values. How can you also incorporate the Lift values to such a plot?

2 Mining frequent itemsets and association rules in a movie rating dataset (22 Pts.)

In this part we will be applying the modules we have implemented in the previous part to analyze movie preferences of users. Imagine you are an analyst in a media-services provider that allows the customer to stream movies and rate them. You are asked to find what movies are liked and dislike at the same time to help design the application layout and the recommender systems. Our stakeholders are also interested in demographic analysis since they would include that into their recommender systems.

2.1 Building the transactional databases (8 pts.)

Let's simulate this scenario by starting with a descriptive and diagnostic data mining application and analyzing the Movielens dataset from GitHub (check the README file for more information about the content). Our observations are in the file `ratings.dat` and our first task is now to turn our observation into transactional database of preferences. To that end, every transaction in our database will correspond to a unique user (note the user ids defined in the README file). We will create two transactional databases T^+ and T^- that respectively contain the movies that users liked and disliked. Since we are dealing with a 5-star rating we will assume a user i likes movie j if the corresponding rating is either 5 or 4. Similarly, we will assume a user disliked a movie if the rating value is 2 or 1. In summary each transaction in $T_i \in T^+$ will contain the list of movie ids that the ith user liked and each transaction in $T_l \in T^-$ will contain the list of movie ids that the lth user disliked.

2.2 Getting first results (6 pts.)

Once we have created T^+ and T^- we will apply our previously implemented algorithms to extract the frequent itemsets as well as strong association rules in T^+ and T^-. Show a (concatenated) list of the extracted results and make sure you use the movie names. You can use your scatter plots from the previous section as well. Note that, we will determine the values of $S_{\min}$ and $C_{\min}$ based on our computational resources and the presentation content[1].

2.3 A deeper look (8 Pts.)

Repeat the same exercise by considering the demographical information in `users.dat`. Are there noticeable differences between female and male users? What about the preferences of the students compared to the rest of the population? Our results should be interpretable and tell a story because the stakeholders in this scenario are not analysts but designers and marketers.

3 Presenting the results

The results will be presented as either a Jupyter Notebook or a PDF with an accompanying code base. They will be discussed in an exercise session.

Submission

All the submissions will be made electronically by sending a single `.zip` file (including your Python code and PDF or Jupyter Notebook) to sparrenberg@bit.uni-bonn.de by the submission deadline with the title `MMD WS2025 Assignment 01 [GroupID]`, where `[GroupID]` refers to your group id (name). Submissions sent after the deadline and not following the title convention will not be evaluated. The submission deadline for this assignment is on 18/11/2025 at 23:59 (CET).

A Note on Plagiarism

Work containing plagiarism will not be graded. After a second warning, a disciplinary process will be started, and the corresponding students will not be allowed to attend the final examination.

[1] If you are not sure about selecting the values of $S_{\min}$ and $C_{\min}$, make sure we have at least one frequent 3-itemset for each of the transactional database in your results.