Big Data Application - Project progress report 1

Group information

ID	Name
21127170	Nguyễn Thế Thiện
21127326	Nguyễn Trần Trung Kiên
21127329	Châu Tấn Kiệt

Problem summary

Problem

Title: Real-time anime recommendation system based on user ratings.

Description:

Analyze real-time anime ratings data using Big Data and Machine Learning tools, in order to recommend animes a user has yet to watch, based on the user's rating history.

Input: User's rating history on watched animes.

Output: Ranking list for recommended animes fitting user's tastes.

Dataset

Anime Dataset 2023 [1] by username Sajid from Kaggle, which is a collection of user and anime ratings on one of the largest anime databases and communities - MyAnimeList (myanimelist.net).

The files contributing to the dataset:

users-score-2023.csv (1.16GB): **The main data** consisting of user ratings on anime titles, provided by 270K users on 16K anime titles, with a total of 24.3M samples, **99.48% sparsity rate** for only the observed users and anime titles.

These files below have been decided to not be used after further research and considerations:

anime-dataset-2023.csv (15.92MB): Details of around 25K anime titles on MyAnimeList.

user-details-2023.csv (73.93MB): Details of around 730K users registered on MyAnimeList.

Main tasks

1. **Data ingestion**: Set up Redis database with imported data from the dataset files, and set up data streaming connection.
2. **Data streaming & preprocessing**: Apache Spark Streaming simulates real-time data from Redis davvtabase, then clean and prepare the raw data before feeding into the recommendation system.

3. **Real-time RS model training**: Pre-built model from MLlib is trained by feeding real-time data, from Spark Streaming.
4. **RS in use**: Input user's rating history to predict a ranking list for recommended animes which user has not watched.
5. **Real-time dashboard**: Visualize analyzed data and predictions with NetworkX-assisted Matplotlib.

Plan progress

1. **Preparation**: 100% done.

- Set up Redis database and data streaming to Spark.
- Figure data preprocessing strategies, perform data preprocessing on dataset using Spark.
- Set up RS model from MLlib, learn its required input and output forms for training and testing.
- Figure out how to save RS model into a file for further training.

2. **Tool testing and systematic setups**: 80% done.

- Perform real-time data processing using Redis and Spark, with data visualization using NetworkX and Matplotlib.
- Set up a basic user interface to apply the use of RS.
- Test RS model training on small scale with multiple batches.

3. **Main events**: 0% done.

- Perform real-time RS model training on dataset.
- Research and experiment documentation.
- Application of RS model into the problem.

4. **Project conclusion**: 0% done.

- Research and experiment documentation and presentation with Canva [7].
- Graphical demonstration.

Assignments

Note: The assignments here are summarized from the submitted work by each member of the group. For evidence, check out their corresponding directories.

Progress report no.	Who	Task(s)	Done	Not done	Upcoming obstacles	Directory
1	Kiên	Set up real-time data streaming from database	Redis installation and configuration. Upload datasets to Redis in the correct formats.	Set up data streaming to Spark to perform data preprocessing.	Connect with Spark Streaming for batch data preprocessing.	/Redis DB

Progress report no.	Who	Task(s)	Done	Not done	Upcoming obstacles	Directory
1	Thiện	Figure & apply data preprocessing strategies.	Load dataset in the correct format. Perform data transformation on categorical columns (anime dataset). Explanation on the strategies used.	Data cleaning: missing values, noisy data. Data transforming: standardization of numerical columns. Data reduction: dimensionality, support (one hot encoding).	Data transforming: extract features from text columns (anime names, synopsis).	/Preprocessing
1	Kiệt	Choose and set up RS model, explanation on why choosing	Setting up the recommendation system models. Read datasets and train/test split. Explore the reason to use the ALS model.	None.	Integrating Spark with Redis database. Solve the problem with real-time data on Redis while using ALS with micro-batch processing.	/RS model
2	Thiện	Apply data preprocessing strategies (cont.). Set up basic UI for prediction program. Deploy RS into prediction program.	Preprocess on datasets. Deploy RS into prediction program: embed user inputs.	Set up basic UI for prediction program. Deploy RS into prediction program: deploy RS.	Reconstruct dataset preprocessing to match with the chosen RS model. Deployed RS that can be updated in real time.	/Thien/predict /Thien/pre-process

Progress report no.	Who	Task(s)	Done	Not done	Upcoming obstacles	Directory
2	Kiên	Redis + Spark: set up data streaming to data preprocessing.	Set up data streaming from Redis to Spark.	Join data streaming to preprocessing.	Reconsider dataset preprocessing to match with the chosen RS model.	/Kien/streaming
3	Kiệt	Spark + ML: test model training with preprocessed datasets.	Setting up the UBCF/IBCF Recommendation System Linked the preprocessed dataset with Spark	Integrate with Redis Streaming Running and validating the model	Working on Redis with the Dataset	/Kiet/test.ipynb

Self-assessment

Struggles

1. **Priorities for other projects**: Delays were made to catch up with other projects for some members. For the others having to wait, would be given tasks on this subject's seminar.
2. **Problem re-definition**: During pre-processing, problem definition has to be assessed over and over again to ensure the most logical pre-processing strategies, which took a lot of time.
3. **Poor planning**: Pre-processed dataset turned out to be not matching with the chosen algorithm.

Problems:

1. **Dataset preprocessing**: Need to re-structure the dataset preprocessing to link the stages.
2. **RS model comparision**: As this project is set to be on our curriculum vitae, we shall need to elevate the problem difficulty and to make it as realistic as possible while still within our budget and time.

References

[1] Sajid Uddin (2023). Anime Dataset 2023. *Kaggle: Your Machine Learning and Data Science Community*. https://www.kaggle.com/datasets/dbdmobile/myanimelist-dataset?resource=download

[2] Salvatore Sanfilippo (2009). Redis 7.4.2 (2025). https://redis.io

[3] Matei Zaharia (2014). Apache Spark 3.5.4 (2024). https://spark.apache.org/

[5] John D. Hunter (2003). Matplotlib 3.10.0 (2024). https://matplotlib.org/

[6] Aric Hagberg, Pieter Swart, Dan Schult (2005). NetworkX 3.4.2 (2024). https://networkx.org/

[7] Melanie Perkins, Cliff Obrecht, Cameron Adams (2013). Canva. https://canva.com