Movie Recommendation System

TEAM 2: Zongyao Li, 002191556 Xuliang Mei, 001305818 Jiaqi Wang 001023711

PROJECT GOAL

This system is designed to:

- Let users get the content they need faster and better
- Let content be pushed faster and better to users who like it
- Let the website (platform) more effectively retain user resources

Use Case

Recommend movies for users based on:

- All-time popular movies
- Recent popular movies
- Popular movies in different genres

Build personalized movie recommendation list based on user activities:

- Analyze user real time ratings of movies and build recommendation
- Analyze activity history and build recommendation

Show similar movies based on movie properties:

- Name
- Genres

Scala Version

Used 2.11.* to be compatible with other packages

Problem with the latest 2.13.*/3.* version:

- Package consistency issue
- Tool incompatibility
- Dependency confliction
- System environment stability

Project Organization

Used Maven to organized the project for:

- Module management
- Package dependencies
- Program structure
- Easier workload division

Modules

Movie Recommendation

|_ Recommender

|_ Data Loader # Load data to database for future use

|_ Content Recommender # Recommend movies based on genre

|_ Statistics Recommender # Recommend movies based on popularity

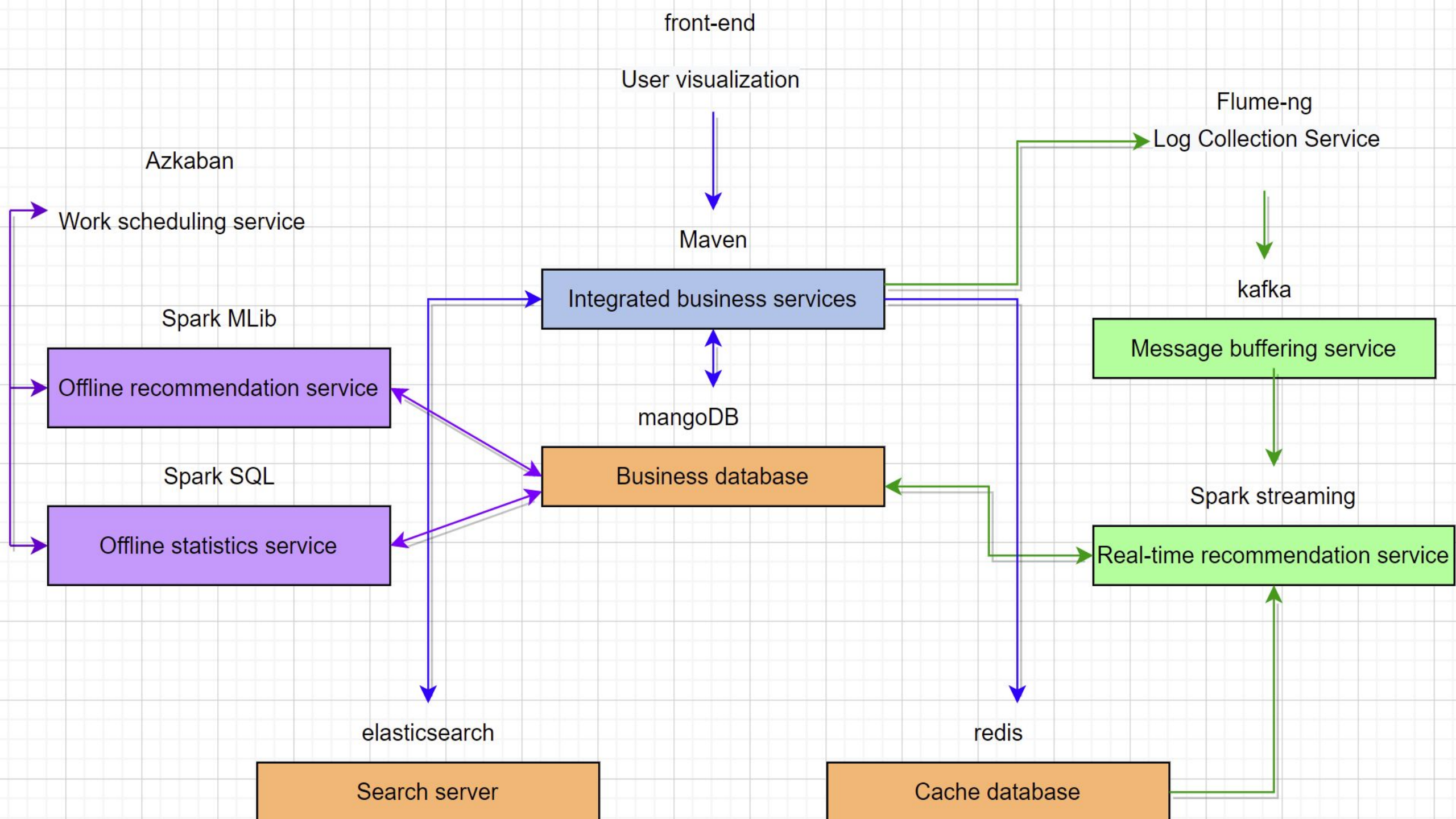
|_ Offline Recommender # Recommend movies based on historical activities

|_ Streaming Recommender # Recommend movies based on real time rating activities

|_ Kafka Log Manager # Logger and processor

|_ User Interface

|_ Web Page # Not yet implemented properly


front-end
User visualization
Flume-ng
Log Collection Service
Azkaban
Work scheduling service
Maven
kafka
Integrated business services
Spark MLib
Message buffering service
Offline recommendation service
mangoDB
Business database
Spark SQL
Spark streaming
Offline statistics service
Real-time recommendation service
elasticsearch
redis
Search server
Cache database

Data sources

The data source is [Kaggle's MovieLens 20M Dataset]

The magnitude

genome_scores.csv: 11.7m	genome_tags.csv: 1128	link.csv: 27.3k	movie.csv: 27.3k	rating.csv:20.0m	tag.csv: 466k

Data processing (Example: Movie.csv)

```
/**
* Movie data file format
*
* 260                                                  Movie_id
* Star Wars: Episode IV - A New Hope (1977)            Movie name
* Princess Leia is captured and held hostage           Detailed description
* 121 minutes                                          length of the movie
* September 21, 2004                                   time of issue
* 1977                                                 time of shoot
* English                                              type of language
* Action|Adventure|Sci-Fi                              genres
* Mark Hamill|Harrison Ford|Carrie Fisher              list of actors
* George Lucas                                         name of directors
*
*/
```

Dataframe or Dataset?

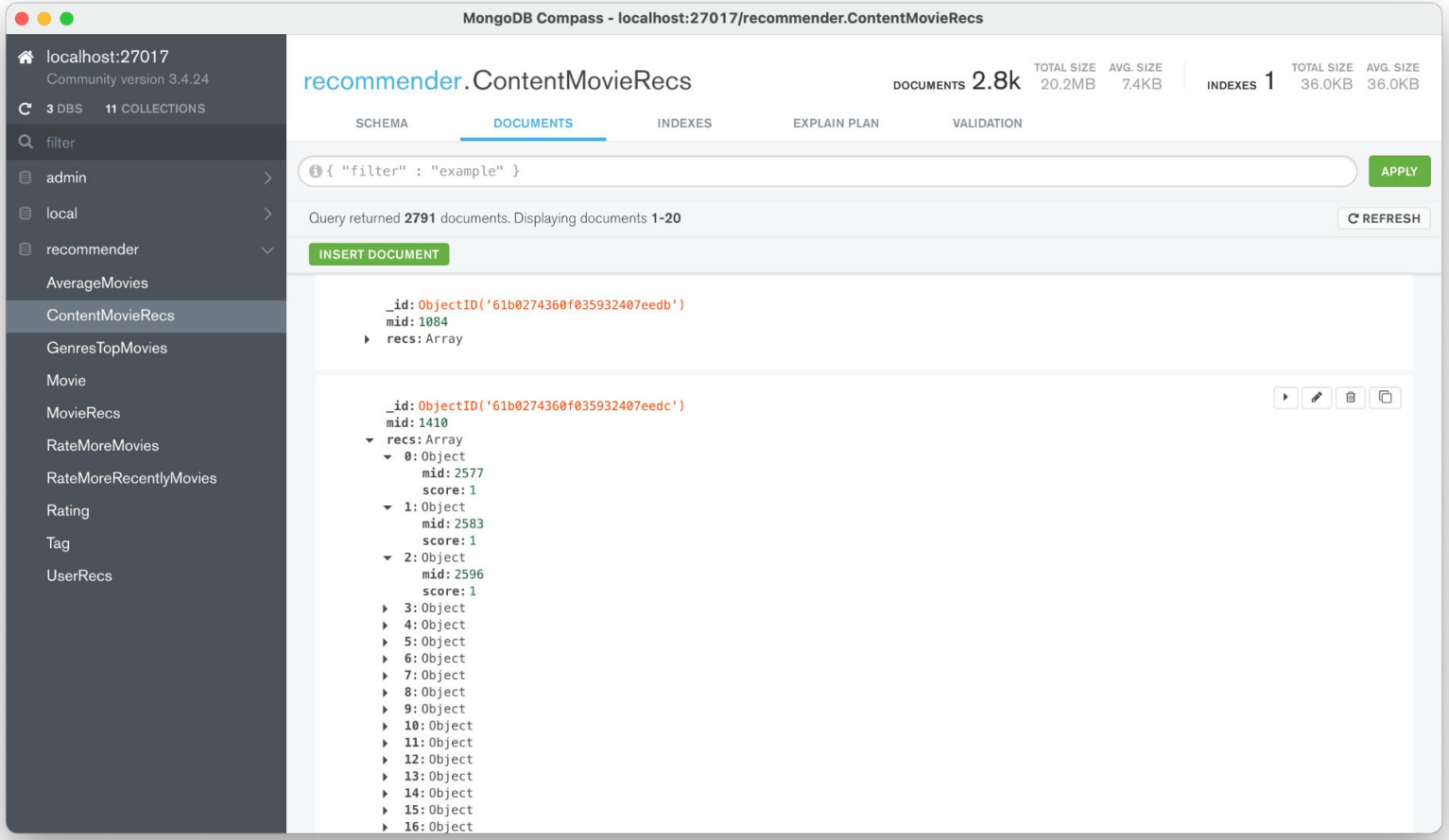
Used dataframe for:

- The main container of data during operation
- Reading data from files
- Reading data from database
- Writing new datas into database.

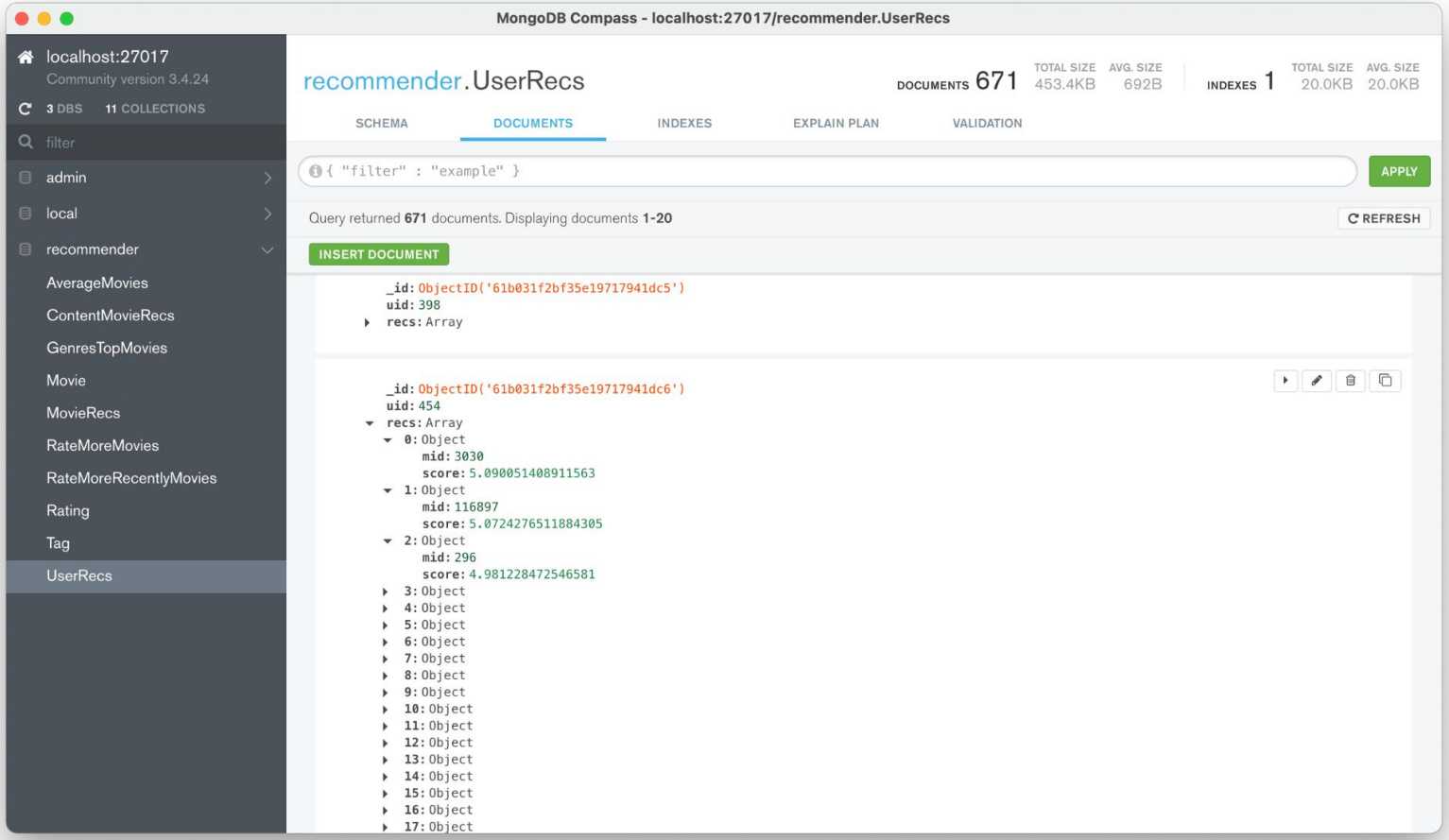
Used dataset for:

- Column selection (filter out unwanted columns)

Demo (content based recommendation)



Demo (offline recommendation)



Workload Division

Xuliang Mei	Zongyao Li	Jiaqi Wang
System Structure Design	System Structure Design	System Structure Design
Data Collector	Module Build Up	Data Loading
Content Module	Statistics Module	Algorithm Implementation
Database Management	Kafka Module	ALS model training
Technical Selection	Code Review	Offline Module
	Framework Deployment	Real Time Module
	Organizer	

Summary & Reflection

Achievements

1. Achieved the overall goal–attain various level of recommendation
2. Got familiar with common big data technologies and tools
3. Strengthened the programming ability of scala

Known Shortage

1. Lack of proper deployment of the front-end

Thanks!
Any questions?