Project Phase lll

Rishab Katta
Rochester Institute of Technology
Rochester,NY, USA
rk4056@rit.edu

Milind Kamath
Rochester Institute of Technology
Rochester,NY, USA
mk6715@rit.edu

Bikash Roy
Rochester Institute of Technology
Rochester,NY, USA
br8376@rit.edu

Ankit Jain
Rochester Institute of Technology
Rochester,NY, USA
aj9761@rit.edu

ABSTRACT

This is a LaTeXdocument for the Phase lll of the Project for CSCI 620 Introduction to Big Data Course. We have chosen Daily-Historical-Stock-Prices-1970-2018 dataset available on Kaggle.com for this Project. Over the the three phases of this project, we have built both relational and non-relational databases by using SQL/NoSQL scripts to load them into Postgresql and MongoDB. In the previous phases we have developed ER- Models, discovered functional functional dependecies and Normalized the Databases. In this particular phase, we have used Data cleaning, Data Integration and Itemset mining concepts to clean, create views and do Itemset Mining on those views to discover interesting assocation rules.

1 INTRODUCTION

In this phase we have analyzed the dataset and checked for anomalies in the Dataset. We are mainly working on the open price, close price, ticker and stock date columns on this dataset. So we particularly looked for any discrepancies in those columns and cleaned the dataset before we began Integration. In the Integration part, we created materialized local views on Company Table and a global view on those local views. The global view that we created in this phase is going to be useful in the Itemset mining part. In the Final Itemset Mining part, we have written code to get all the companies whose stock price went up by 20% in the same day for more than 5 days. We've got all the companies whose stock price went up together and grouped them into itemsets at different levels of our lattice. We have also gotten interesting association rules by checking for support of different rules and having a confidence of more than 50%.

2 RELATIONAL OR DOCUMENTED-ORIENTED MODEL

In this phase of the project, we have selected Relational model as our data base choice over Documented oriented model.
Relational model is well suited for pre-defined schemas and structured data. The data for stocks that we have is well structured and in a tabular format, basically our data requirements are clear. Data in each attribute has the same property or type in that particular attribute and is not bound to change or evolve with time. Though stock values fluctuate, we are analyzing them for years for which the they have already been documented. The analysis is not real time. Additionally, complex queries involving joins are handled well in a relational model. They are a better option for statistical calculations and offer better data integrity using integrity constraints unlike document oriented models. We have used the primary and foreign key constraints to enforce integrity on stock data. Data integrity helps avoid redundant or invalid data. Since our data conforms to the above points, the choice was relational model.

3 DATA CLEANING

The Dataset we'have chosen is daily-historical-stock-prices-1970-2018. This Dataset is a well maintained dataset on Kaggle.com. However we did check for some anomalies. The Data cleaning mainly focuses on columns used for Itemset Mining. The Data Cleaning is performed as described below.

- Foreign key constraint on ticker column of Historical_stock_price to ticker column of Company Table. Ticker is the primary key of Company Table. This acheives that no weird values are present in ticker of Historical_stock_price.
- Checking for any Null or N/A values in the ticker column of Historical_stock_price and deleting those rows if present. Just for safe-keeping.
- Checking for null values in the open_price and close_price columns of the Historical_stock_price and deleting those entries from the table. We are going to be performing mathematical calculations on those rows so it's essential that no null values are present in those rows.
- Checking if all the dates in the stock_date column are in the right format by comparing counts.

Code is provided in the StocksProject3.py file

4 DATA INTEGRATION

We have chosen to create materialized views on Company table because the global view that we're creating will be used multiple times when we're performing Itemset Mining and the Datasets are not being updated by adding new entries at all. So it makes sense to use materialized views because we don't have "refresh" the view at all. So for the Local as view, we're creating finance companies as a subset from the company table where the sector is finance and tech companies as a subset from the company table where the sector is technology. We're a creating a GAV fin_tech_companies over the union of the above local views which we'll actually be using for our mining.

Code is provided in the StocksProject3.py file

5 ITEMSET MINING AND ASSOCIATION RULES

For the Mining Part, we want to get all the companies/group of companies whose stock price went up by 20% on the same day (close_price > open_price*1.2) for more than 5 days. We grouped all the companies whose stock price went up together in different itemsets at different levels of the lattice. We performed this mining on Finance and Technology Sector companies only. When we ran the code it generated 3 levels of the lattice with non-empty rows. L1 with 366 rows. L2 with 99 rows and L3 with 3 rows. We generated Association Rules by generating different combinations of the three rows of L3 and checking if the confidence exceeded our predefined confidence level which is 50. So we can say by 50 percent confidence that if the stock price of EGAIN CORPORATION and GENERAL FINANCE CORPORATION went up by 20% in the same day for more than 5 consecutive days that means that the stock price of PAYMENT DATA SYSTEMS, INC. will also go up. The rows in the final level of our lattice are as follows.

('ATTU', 'ATTUNITY LTD.', 'EGAN', 'EGAIN CORPORATION', 'PYDS', 'PAYMENT DATA SYSTEMS, INC.')

('EGAN', 'EGAIN CORPORATION', 'GFN', 'GENERAL FINANCE CORPORATION', 'PYDS', 'PAYMENT DATA SYSTEMS, INC.')

('EGAN', 'EGAIN CORPORATION', 'PYDS', 'PAYMENT DATA SYSTEMS, INC.', 'SCKT', 'SOCKET MOBILE, INC.')

The Association rules with more than 50% confidence are as follows:

GFN -> PYDS

EGAN,GFN->PYDS

Code is provided in the StocksProject3.py file

6 TIMINGS

Timings for different sections of the code are as follows:

— 21.584691524505615 seconds for cleaning data —

— 0.3907506465911865 seconds for integrating data —

— 2.793757438659668 seconds for creating Popular fintech companies table —

— 21.5847909450531 seconds for generating lattice —

— 0.04951739311218262 seconds for association rule discovery on final level of the lattice —

7 CONCLUSION

In conclusion we'd like to say that we had a lot of fun working on this project and we feel like we learned quite a lot about Normalization, Data Cleaning, Integration and Itemset Mining.