International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Real-Time Stock Market Simulator

Shubham Gupta[1], Dr. Vishalakshi Prabhu[2]

[1]Student, Dept. of Computer Science and Engineering, R V College of Engineering, Bengaluru
Postal Address: 42, 2nd Cross, Ayyappa Layout, Marathahalli, Bengaluru-560037
[2]Assistant Professor, Dept. of Computer Science and Engineering, R V College of Engineering, Bengaluru
Postal Address: Dr. Vishalakshi Prabhu, Assistant Professor, Dept. of Computer Science and Engineering, R V College of Engineering, Bengaluru-560059

Abstract - *The stock market can be a network that provides the platform for almost all major financial transactions worldwide, known as the stock price, which is measured at market equilibrium. Tinkering stock prices provides huge information for profit opportunities that can be a huge inspiration for research in this area. Getting information on stock prices for a fraction of a second already results in higher profits. Similarly, accurate estimates of small start-up costs can be very profitable. This fascination with finding answers has prompted researchers in industry and academia to find some way to address issues such as volatility, seasonality and dependence on time, economy and the rest of the market.*

Key Words: stocks, stock market, stock value, bear market, bull market

1. INTRODUCTION

Today, all organizations operate in an environment of uncertainty. The decisions made by organizations today affect future results. Organizations and platforms continue to produce enormous data for analysis: complete, optimization and development of electrical circuits. Analyze market data, change it, and visualize according to wants. A method of having clean data in real time for faster and better analysis and management of information. Market Simulator is a major part of the market research. It displays long-term numbers, characteristics, and trends in your target market. A simple analysis is divided into the number of potential customers. Most investment banks do not have their own Forecaster tool that provides detailed analysis of the information. They rely on third party portals and services for detailed market research. In the market forecast, the example number indicates that 25,000 domestic offices are included in the market, an increase in the usual fraction estimated annually. There are many small businesses in the region, and this number is growing at a rate of five percent annually. These numbers are estimates. No one really knows, but we all make educated guesses. The developers of this scheme have done further research on the market because they can estimate the target consumer population in their area and the annual growth rate of each. You can use your market forecast numbers to draw a chart of projected market growth, as shown below. It provides a visual view of the market forecast.

The paper is organized as follows. Section II discusses the language and framework needed to understand the rest of the paper. Section III discusses the high-level and low-level design of simulation. Section IV presents the results of this paper. Section V discusses the end of the paper and its future possibilities.

2. FRAMEWORKS

2.1 JAVA

Java is the most used programming language, primarily designed for use in the Internet and the local machine distribution environment. It is also the most popular programming language for android smartphone applications and also the most popular for edge devices and IoT development.

Java is designed to look, feel and resemble C and C ++ programming languages, but is simple to use and allows for an object-oriented programming model. This platform is independent. It can also be used to create a small application module or applet for use as part of a webpage.[3]

2.2 SPRING FRAMEWORK

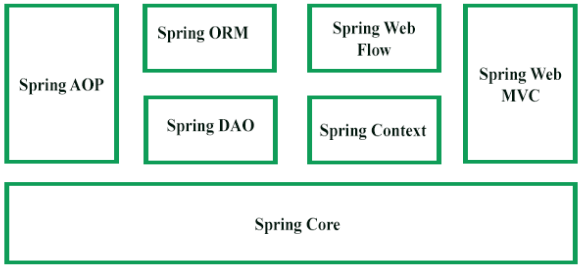

Figure 1: Spring framework architecture

Spring is one of the most popular frameworks (fig. 1) for the Java Enterprise Edition. Developers around the world use Spring to develop reliable and high-quality applications. Spring Framework was designed by Rod Johnson. Since then, Java has become an alternative technology to the EJB model

in the Java world.[4] You can create a variety of applications using the Spring Framework. The Spring Framework is divided into several different modules, which allow you to decide what to use in your application. The image below shows the most important modules of Spring Framework architecture. At its core, Spring Framework is really a DI container with some feature layers (for eg. Proxy, database access, aspect-based programming, Web Framework and RPC). This will help you build Java applications faster and easier.[5]

2.3 DEPENDENCY INJECTION

The most commonly identified technique with spring and other methods is the abstract inversion of dependency injection (DI) control. Inverse control (IoC) can be a simple and basic concept and is often expressed in many different ways. Dependency injection is just one concrete example of inverse control.[5] Application classes should be as independent as possible to other Java classes when writing fancy Java applications, to extend the similarity to reuse these classes, and to independently examine other classes when performing unit tests. What is Dependency Injection? Let us study these two terms separately. The dependency component here translates into a collaborative collection between two classes. For example, the tenth grade is based on Advanced Y. Now, study the second part of the injection. This indicates that the tenth class is injected into class A by the IoC. Dependency injection operates on parameters beyond the constructor or by using setter and getter methods after construction.[3]

2.4 ASPECT ORIENTED PROGRAMMING

One of the most important features of Spring is the Aspect Oriented Programming (AOP) framework. Functions that extend different points of function are called cross-cutting concerns, and the application's business and work logic vary according to these cross-cutting concerns. There are many good examples, including declarative transactions, logging, security, caching, and more.[2] The main characteristic of modularity in object-oriented is that the class is the unit of modularity in aspect-oriented programming. Dependency injection helps you reduce our application objects from other objects, while aspect-based programming helps us reduce cross-cutting concerns from affected objects. The Spring Framework's aspect-based programming module provides an AOP implementation that allows you to define method-blockers and points for clean separation of code that separates functionality and features.[6]

3. EXPERIMENT

As the problem of the prediction may be complex and lengthy, a series of actions and activities within the sort of several phases was considered to interrupt down the matter to beat the complexity. Figure 2 below depicts the stock value analysis for company A and Figure 3 below depicts the stock value analysis for company B. These are analysis between actual and predicted stock price and provides complete overview of what is to be market share prices in future.[8]

4. RESULT

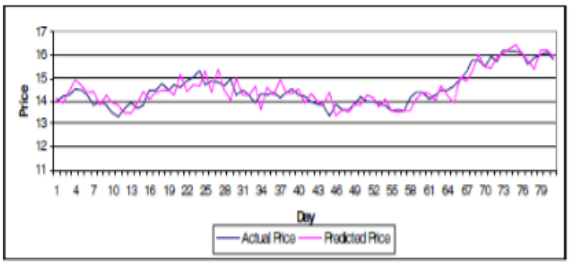

Figure 2. Actual Vs. Predicted stock price

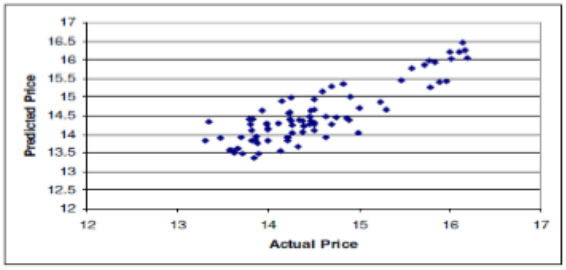

Figure 3. The correlation between predicted and actual closing stock price

5. CONCLUSION AND FUTURE WORKS

As it is expected, we have been successful getting fair idea of estimation of the long run data required for the project. Though proper algorithms are to be implemented so as to get precise estimation. Although HMM gives the precise fraction change in the futured value which are close to the future stock value predicted in real life.[2] Moreover the application can be used to buy and sell stocks given the correct information of the broker. The application directly gets linked to broker server and helps in smooth transaction. In future works, the application will try to generate visuals graphs for separate companies which involves collection of data for individual companies and store it into an array. This array is then fed the value to the axis and generate graph.[1]

Also, some data filtration techniques are to be implemented so as to ensure no null data is stored in the data. Partition of the data is to be implemented which keeps data backup. This is what is known as data warehousing.

REFERENCES

[1] Kuo R J, Lee L C and Lee C F (1996), Integration of Artificial NN and Fuzzy Delphi for Stock market forecasting, IEEE International Conference on Systems, Man, and Cybernetics, Vol. 2, pp. 1073-1078.

[2] Kimoto T, Asakawa K, Yoda M and Takeoka M (1990), Stock market prediction system with modular neural networks, Proc. International Joint Conference on Neural Networks, San Diego, Vol. 1, pp. 1-6.

[3] D. R. Prasanna, Dependency Injection: Design Patterns Using Spring and Guice, Manning Publications, 2009.

[4] R. Johnson, J. Hoeller, A. Arendsen, T. Risberg, and D. Kopylenko, Professional Java development with the Spring Framework, vol. 2005. 2005.

[5] F. Gutierrez, "Spring Boot, Simplifying Everything," in Introducing Spring Framework, 2014, pp. 263–176.

[6] Thammano A (1999), Neuro-fuzzy Model for Stock Market Prediction, Proceedings of the Artificial Neural Networks in Engineering Conference, ASME Press, New York, pp. 587-591.

[7] Abraham A, Nath B and Mahanti P K (2001), Hybrid Intelligent Systems for Stock Market Analysis,Proceedings of the International Conference on Computational Science. Springer, pp. 337-345.

[8] Raposo R De C T and Cruz A J De O (2004), Stock Market prediction based on fundamentalist analysis with Fuzzy-Neural Networks. http://www.labic.nce.ufrj.br/downloads/3wses_fsfs_2002.pdf