UNIVERSIDAD AUTÓNOMA DE MADRID
ESCUELA POLITÉCNICA SUPERIOR

Bachelor as Ingeniería Informática Bilingüe

BACHELOR THESIS

Develop a web application to interact with predictive algorithms about the stock market.

Autor: Juan Llamazares Ruiz
Tutor: Franciso SAi

January 2023

Juan Llamazares Ruiz
Develop a web application to interact with predictive algorithms about the stock market.

Juan Llamazares Ruiz

C\ Francisco Tomás y Valiente No 11

Resumen

Palabras clave

Aplicación web, algoritmos predictivos, mercado de valores, predicción bursátil, aprendizaje automático, redes neuronales, LSTM, análisis de datos, tendencias del mercado, datos históricos.

Abstract

The stock market is a complex and dynamic, and it can be difficult for investors to keep up with the latest trends and make accurate predictions about future market movements. By developing a web application that can interact with predictive algorithms, investors can access real-time data and historical trends to make informed investment decisions. This can help individuals and investors to optimize their time and effort to decide the best

Keywords

Web application, predictive algorithms, stock market, stock prediction, machine learning, neuronal networks, LSTM, data analysis, market trends, historical data,

Table Of Contents

Lists

List of equations

List of figures

Introduction

In recent years, machine learning has improved to the point that is a very powerful tool to analyse and improve the performance and predictions about future events. It has been used by many institutions and individuals as it has plenty of use cases. For financial purposes has heavily increased in the past years. I end up searching about the use cases of machine learning in the stock market.

The stock market is huge financial market where publicly traded companies' stocks (shares) are bought and sold. It is one of the most important sources of capital for companies, and it allows many individual investors and institutions to buy and sell ownership stakes in publicly traded companies.

One of the main advantages of using machine learning for stock market predictions is its ability to process large amounts of data and identify patterns that may not be visible for the human eye.

Many people in the world invest in this type of asset. Most of them do it based on the financial company results or sometimes based on technical analysis.

1.1. Motivation

Stock market is laborious asset that requires time a deep understanding of the financial markets. More and more people are entering this sector and because of the vast amount of data available its learning curve is increasing constantly. A web-based interactive tool could help investors and non-experience individuals to learn about the effectiveness of prediction models and make more informed decisions. Additionally, the interactive web-based tool will provide an intuitive and user-friendly platform for users to interact with the predictions making it accessible for a large range of users.

1.2. Objectives

The goal of this work is to develop a web-based application tool to provide and interactive platform that allows users to learn about stock market predictions and make informed decisions using different machine learning prediction algorithms.

The main objectives are now described:

Error! Use the Home tab to apply Título 3 to the text that you want to appear here.

- Obtain the minimum possible error compared to the actual closing price; it will be visualized through the web-based tool. Historical data will be used for these predictions.
- Provide different techniques to predict the stock market to improve the user decision-making with visual representations.
- Facilitate data analysis. The tool will allow users to filter, search and analyse different prediction models and compare their effectiveness based on various parameters. I will help users to understand the performance of different models and identify patterns and trends in the data that may not be immediately apparent to the human eye.
- The tool will be up to date as it uses data from financial APIs, it will allow to have an updated, providing accurate up-to-date predictions.

, search and analyse different prediction algorithm models and compare them by its effectiveness based on different parameters. The data used will be extracted from different financial API's. The project will be developed in a python environment using Django as the main framework.

happen

2

State of the art

Machine learning techniques, like Long Short-term Memory (LSTM) networks, have been widely used in recent years for stock market predictions. LSTM networks are a type of recurrent neural network that is used mainly for times series data, like stock predictions. It can learn and remember past patterns in the data, making them very useful for predicting future trends.

Apart from LSTM networks, other studies have also used other machine learning techniques such as Random Forest and Gated Recurrent Unit (GRU)

In the following section, we will see the explanation if each of the technologies involved in the development of the project.

2.1. Machine Learning

The technology used for predictions is called "Machine Learning" it is a division of artificial intelligence and computer science that uses data and algorithms to generate an imitation to an analysis improving its accuracy.

..1. Long Short-Term Memory neuronal network

Long Short-term Memory (LSTM) networks is a type of neural network (RNN) used for a variety of tasks, including stock market predictions, machine translation, speech recognition, and more. It is particularly effective for tasks that involve long-term dependencies, where the past inputs have a significant impact on future outputs,

The LSTM architecture consists of memory cells and gates that regulate the flow of the information through the network. The memory cells are used to store information over time, allowing the network to selectively remember and forget information as needed. The gates are made up of sigmoid and tanh activation functions, they control the flow of the information into and out of the memory cells.
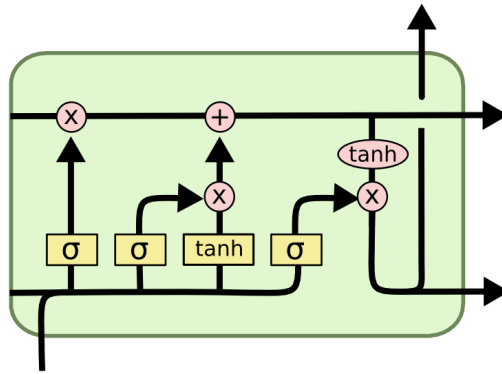
Figure 1: LSTM architecture

..2. Gated Recurrent Units (GRU)

A Gated Recurrent Unit (GRU) is a type of recurrent neural network architecture for sequential data processing. The have been designed to handle sequential data, such a speech signals, time series data and text.

The architecture of the GRU neuronal network consists of an input layer, a hidden layer and an output layer. The hidden layer contains a set of GRU units, each of which maintains a hidden state vector that is updated based on the input data and the previous hidden state.

The GRU network is computationally efficient and can be trained using standard backpropagation techniques.

Figure 2: GRU architecture
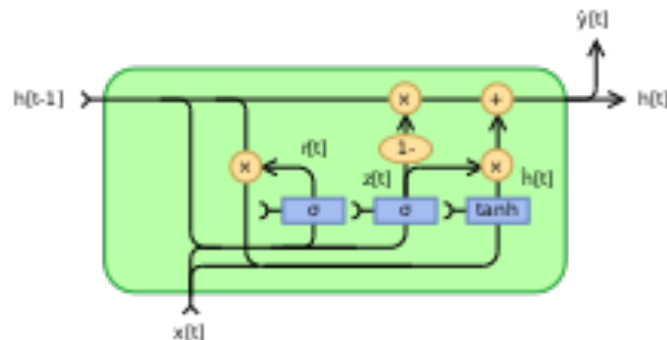
..3. Long Short-Term Memory (LSTM) vs Gated Recurrent Units (GRU)

The LSTM (Long Short-Term Memory) and GRU (Gated Recurrent Unit) networks are both types of recurrent neural networks that have been widely used for sequential data processing. While both networks are designed to handle long-term dependencies in sequential data, there are some key differences between the two.

One of the main differences between these two neuronal networks is the number of control gates to control the flow of the information, While GRU uses two gates (an update gate and a reset gate), the LSTM networks uses three gates (an input gate, an output gate and a forget gate), this last gate allows to selectively discard information that is no longer relevant, which is particularly useful for tasks such as language modelling and performance,

2.2. Python 3

Python is a popular, high-level programming language that is widely used for web development

..4. Framework Django

All code has been written on Django [1], it is one the most used frameworks in python 3.

2.3. Alphavantage API

To get the data used as input of the neuronal network we will use [2].

In order to get the data of the times series, it is necessary to generate an API key.

2.4. Javascript

2.5. ChartJS

ChartJs is an open-source JavaScript library for creating charts on web pages. It allows to customize and modify various types of charts, including bar charts, pie charts and scatter plots.

Analysis and Design

This chapter will describe the design realised and the procedures for achieving the project goals. it will be divided in two main layers, the backend, responsible for processing and managing data, logic, and communication between the server and client-side of the web application, And the frontend which is responsible for displaying and interacting with the user interface of the web application, including layout, design, and user experience.

Backend development

..5. 3.2 Datasets

The data provided for the development of the project as well as the data used to train the models to predict the future values of the company stock comes from an API. In this case we will use the free tier version of the Alphavantage API.

3.2.1 Alphavantage API

will be managed by Alphavantage API [2] it provides financial market data from traditional asset classes such as stocks, ETFs, or mutual funds. It has a free version that will be used in this project

The endpoint used to get historic closing prices of each stock is called TIME_SERIES_DAILY, it covers at least 20 years of data

3.1 Deep Learning

There are different ways to try to predict the value of an asset. In this case, deep algorithms are used for the estimation. Inside of deep learning we find a variety of recurrent neural networks that are algorithms used for learning long term dependencies.

3.3.1 Develop LSTM model

The steps to create a LSTM model are:

Python libraries

- **Tensorflow:** is an open-source library developed by Google for machine learning and artificial intelligence. It is used to train and deploy machine learning models.
- **Keras:** is an open-source software library that provides a Python interface of artificial neural networks.
- **Scikit-learn:** is an open-source Python [4] library for machine learning. It provides a range of tools for tasks.
- **Numpy:** is a Python library that supports a large collection of high-level mathematical functions to operate multi-dimensional arrays and matrices.
- **Matplot:** Matplot is a plotting library for Python. It provides many customizable plotting options. Allowing to visualize data in 2D and 3D. It is well used with other Python libraries such as Numpy and Pandas.
- **CSV File reading and writing:** Python library used to read and write the financial data used in the model prediction.

[4]
3.4 Frontend libraries

For the front e

4

Implementation

Through this section, the development of the project will be described. It will be divided in different steps.

The project has been developed following a lean process. Starting from a first web version that worked with all the components of the project, from the API to the frontend development.

The distributed system will be divided into two sections, the Back-end and the Front-end.

Back-end development

The backend of the project is divided into two main parts. First one is the neuronal network which is the most important part since is used to make the stock predictions. The second main part is the development of the views in the Django framework.

Starting with the neuronal network, the development has followed different steps:

- **Import data:** Begin by importing the required libraries, such as pandas and numpy. Use the csv library to read the data from a CSV file. If the data is not stored locally, it will make a call to a financial API to retrieve the full database of stock. The only metadata we need for the time series prediction is the date and the closing price of the asset.

- **Data normalization:** Once you have the data required it is crucial to normalize the data. This step ensures that all input features have similar scales, which helps in achieving better performance and convergence. There are a few different methods to normalize the data. In this case we have used Min-Max Scaling technique from library.

- **Dataset split, train, and test arrays:** The division of the dataset into training and testing arrays is a fundamental element of machine learning and predictive modelling. This division has the purpose of ensuring an appropriate evaluation of the model's performance over unseen data, which in turn provides a thorough understanding of its generalization capacity. The split ratio used is 80:20 is adopted, with 80% of the dataset assigned to the training array, and the remaining 20% to the testing array.

Built and train models: Next, build and train the LSTM model using deep learning libraries like TensorFlow and Keras.

- Define the number of LSTM layers, hidden units, and activation functions according to your specific requirements. Train the model using the normalized data, split into training and validation sets. Adjust hyperparameters like learning rate, batch size, and number of epochs to optimize model performance. Repeat this step for the GRU (Gated Recurrent Unit) model, which is another type of recurrent neural network.

- **Predictions and test:** After completing training, predictions are made on the test dataset and are compared against the actual values. To measure the accuracy and precision of the models, various metrics are used, such as Mean Squared Error (MSE) and Root Mean Square Error (RMSE) to assess the accuracy and precision of the models. We will go into details of the results in section 5.

- **Calculating Metrics:** To accurately evaluate the performance of the models, additional evaluation metrics should be calculated. This includes mean absolute error (MAE), mean absolute percentage error (MAPE) and R-squared (R2) score. These metrics will provide an exhaustive insight into the model's predictive power and accuracy.

- **Plot results:** In the last part of the development of the neuronal network, we have plotted the results of the predictions versus the true values is essential to understand the performance of the model. Line plots or candlestick charts can be used to visualize the results. Such visualizations can help to identify patterns and trends in the stock market data that the model was able to capture. By comparing the predictions and ground truth data, any discrepancies or patterns can be observed.

- **Improve Neuronal Network:** To optimize the performance of the LSTM and GRU models, analyse these models and identify areas of improvement. Try different network architectures, activation functions, regularization techniques, or hyperparameter tuning to increase model performance. Utilize the process of model building, training, and evaluation to see greater results. Iterate through these steps, and assess the results, to reach a higher level of excellence.

Front-end development

As we have mention in the *Analysis and design* section. In this section, we will talk about all the Front-end development of our web application. All the views, components and functions used will be described. The following figure show a diagram of the structure used in the project.

The main views are divided into different templates.

The first template it's the home view in Figure 3. It's the first view the user will access, it contains a short description about the project and some contact information.
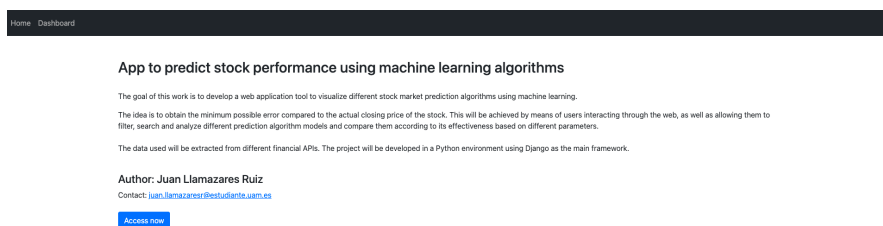

Figure 3: Home view

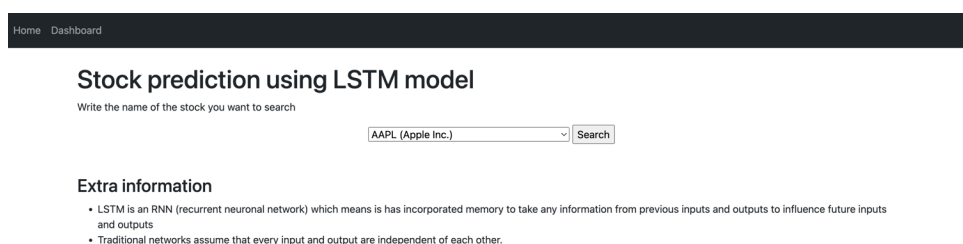The dashboard will be the main view of the project, it has a

Figure 4: Dashboard view

Once the user interacts with the search input, it will have access to the list of all the stock models generated.
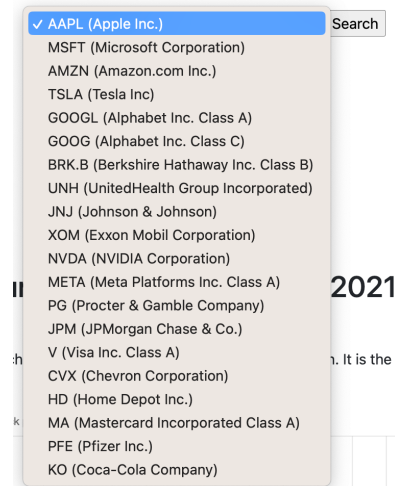
AAPL (Apple Inc.) ✓
MSFT (Microsoft Corporation)
AMZN (Amazon.com Inc.)
TSLA (Tesla Inc)
GOOGL (Alphabet Inc. Class A)
GOOG (Alphabet Inc. Class C)
BRK.B (Berkshire Hathaway Inc. Class B)
UNH (UnitedHealth Group Incorporated)
JNJ (Johnson & Johnson)
XOM (Exxon Mobil Corporation)
NVDA (NVIDIA Corporation)
META (Meta Platforms Inc. Class A)
PG (Procter & Gamble Company)
JPM (JPMorgan Chase & Co.)
V (Visa Inc. Class A)
CVX (Chevron Corporation)
HD (Home Depot Inc.)
MA (Mastercard Incorporated Class A)
PFE (Pfizer Inc.)
KO (Coca-Cola Company)

Search

Figure 5: List of stocks available

LSTM model (Sequential model)

Description:
Test: 20%
Train: 80%

Root Mean Square Error (RMSE): 10.2%

Mean Square Error (MAPE): 2.69%

Predicting one day into the future (TRAIN vs TEST) [08/2021 - 01/2023]

Definition:

Close stock price: The closing price is the last price at which a stock trades during a regular trading session. It is the price at which a trader can close out an existing long or short position.

Figure 6: Chart view

5

<u>Testing and Results</u>

Introduction

In this section, we will go into detail on all the tests carried out including the development of results obtained.

As we have mention, testing and results has been a very important role in the evolution of the project. As the project consist in developing a web application, the testing will consist in the improvement of the technology used in the backend. Without testing different parameters and machine learning techniques, the results itself would not be as accurate as the obtained.

LSTM model tests

GRU model tests

Alternatives to improve results

Improve financial API (free tier API does not consider splits and other data)

Improve the size of the testing

Use other techniques like

6

Conclusions and future work

Conclusions

In conclusion, stock prediction

Future work

Bibliography

[1] "Alphavantage API," [Online]. Available: https://www.alphavantage.co/.

[2] "Alphavantage API," [Online]. Available: https://www.alphavantage.co/.

[3] "Python," [Online]. Available: https://www.python.org/. [Accessed 2023].

Terminology

15

Appendices

Example of Alphavantage API response

Apéndice A