Forecasting Stock Market Prices

Abstract:-

In the era of big data, deep learning for predicting stock market prices and trends has become even more popular than before. We collected 2 years of data from Chinese stock market and proposed a comprehensive customization of feature engineering and deep learning-based model for predicting price trend of stock markets. The proposed solution is comprehensive as it includes pre-processing of the stock market dataset, utilization of multiple feature engineering techniques, combined with a customized deep learning based system for stock market price trend prediction. We conducted comprehensive evaluations on frequently used machine learning models and conclude that our proposed solution outperforms due to the comprehensive feature engineering that we built.

Introduction:-

A time series is simply a series of data points ordered in time. In a time series, time is often the independent variable and the goal is usually to make a forecast for the future. Our Aim is to create a model that can forecast the future stock price based on the model training and provided dataset.

Stock market is one of the major fields that investors are dedicated to, thus stock market price trend prediction is always a hot topic for researchers from both financial and technical domains. In this research, our objective is to build a state-of-art prediction model for price trend prediction, which focuses on short-term price trend prediction.

Dataset:-

This section details the data that was extracted from the public data sources, and the final dataset that was prepared. Stock market-related data are diverse, so we first compared the related works from the survey of financial research works in stock market data analysis to specify the data collection directions. After collecting the data, we defined a data structure of the dataset.

We will be using a [Huge stock market dataset]from the Kaggle platform which has a very good collection of datasets.The file we will be using is present in following directory in the dataset zip file input\Data\Stocks\gs.us.txt

The data is presented in CSV format as follows : Date, Open, High, Low, Close, Volume, OpenInt.

Note that prices have been adjusted for dividends and splits.

Dataflow:-

* Applying recursive feature elimination
* Applying principal component analysis (PCA)
* Fitting long short-term memory (LSTM) model

Diagram

Description automatically generated

Feature extension and RFE:-

From the result of the previous subsection, we can see that when predicting the price trend for every other day or biweekly, the best result is achieved by selecting a large number of features. Within the selected features, some features processed from extension methods have better ranks than original features, which proves that the feature extension method is useful for optimizing the model.

Chart, waterfall chart

Description automatically generated

Relationship between feature number and training time:-

Chart, bar chart

Description automatically generated

Comparison with related works:-

From the previous works, we found the most commonly exploited models for short-term stock market price trend prediction are support vector machine (SVM), multilayer perceptron artificial neural network (MLP), Naive Bayes classifier (NB), random forest classifier (RAF) and logistic regression classifier (LR). The test case of comparison is also bi-weekly price trend prediction, to evaluate the best result of all models, we keep all 29 features selected by the RFE algorithm. For MLP evaluation, to test if the number of hidden layers would affect the metric scores, we noted layer number as *n* and tested *n*= {1, 3, 5}, 150 training epochs for all the tests, found slight differences in the model performance, which indicates that the variable of MLP layer number hardly affects the metric scores.

Graphical user interface, application, Teams

Description automatically generated

 it is also a unique and heuristic innovation in our proposed solution, we transform the problem of predicting an exact price straight forward to two sequential problems, i.e., predicting the price trend first, focus on building an accurate binary classification model, construct a solid foundation for predicting the exact price change in future works. Besides the different result structure, the datasets that previous works researched on are also different from our work.

Chart, histogram

Description automatically generated

Libraries Involved:-

1. NumPy

2. Pandas

3. matplotlib

4. scikit-learn

5. statsmodels

Steps:-

1. Importing Libraries

2. Exploring the Dataset

3. Exploratory Data Analysis

\* Univariate Analysis

4. Data Preprocessing

5. Model Building

\* AUTOREGRESSIVE MODEL

\* MOVING AVERAGE MODEL

6. Evaluation

\* MEAN SQUARE ERROR

\* MEAN ABSOLUTE ERROR

\* ROOT MEAN SQUARE ERROR

Models Used:

Autoregressive Model:-

In a multiple regression model, we forecast the variable of interest using a linear combination of predictors. In an autoregression model, we forecast the variable of interest using a linear combination of *past values of the variable*. The term *auto*regression indicates that it is a regression of the variable against itself.

Thus, an autoregressive model of order pp can be written asyt=c+ϕ1yt−1+ϕ2yt−2+⋯+ϕpyt−p+εt,yt=c+ϕ1yt−1+ϕ2yt−2+⋯+ϕpyt−p+εt,where εtεt is white noise. This is like a multiple regression but with *lagged values* of ytyt as predictors. We refer to this as an AR(pp) model, an autoregressive model of order pp.

A picture containing text, needle

Description automatically generated

For an AR(1) model:

when ϕ1=0ϕ1=0, ytyt is equivalent to white noise;

when ϕ1=1ϕ1=1 and c=0c=0, ytyt is equivalent to a random walk;

when ϕ1=1ϕ1=1 and c≠0c≠0, ytyt is equivalent to a random walk with drift;

when ϕ1<0ϕ1<0, ytyt tends to oscillate around the mean.

We normally restrict autoregressive models to stationary data, in which case some constraints on the values of the parameters are required.

For an AR(1) model: −1<ϕ1<1−1<ϕ1<1.

For an AR(2) model: −1<ϕ2<1−1<ϕ2<1, ϕ1+ϕ2<1ϕ1+ϕ2<1, ϕ2−ϕ1<1ϕ2−ϕ1<1.

When p≥3p≥3, the restrictions are much more complicated. R takes care of these restrictions when estimating a model.

Moving Average model:-

Rather than using past values of the forecast variable in a regression, a moving average model uses past forecast errors in a regression-like model.

yt=c+εt+θ1εt−1+θ2εt−2+⋯+θqεt−q,yt=c+εt+θ1εt−1+θ2εt−2+⋯+θqεt−q,

where εtεt is white noise. We refer to this as an MA(qq) model, a moving average model of order qq. Of course, we do not *observe* the values of εtεt, so it is not really a regression in the usual sense.

Notice that each value of ytyt can be thought of as a weighted moving average of the past few forecast errors. However, moving average *models* should not be confused with the moving average. A moving average model is used for forecasting future values, while moving average smoothing is used for estimating the trend-cycle of past values.

A picture containing text, needle, antenna

Description automatically generated

It is possible to write any stationary AR(pp) model as an MA(∞∞) model. For example, using repeated substitution, we can demonstrate this for an AR(1) model:

Yt =ϕ1yt−1+εt

=ϕ1(ϕ1yt−2+εt−1)+εt

=ϕ21yt−2+ϕ1εt−1+εt

=ϕ31yt−3+ϕ21εt−2+ϕ1εt−1+εtetc.

Yt =ϕ1yt−1+εt

=ϕ1(ϕ1yt−2+εt−1)+εt

=ϕ12yt−2+ϕ1εt−1+εt

=ϕ13yt−3+ϕ12εt−2+ϕ1εt−1+εtetc.

Provided −1<ϕ1<1−1<ϕ1<1, the value of ϕk1ϕ1k will get smaller as kk gets larger. So eventually we

obtainyt=εt+ϕ1εt−1+ϕ21εt−2+ϕ31εt−3+⋯

Conclusion:-

This work consists of three parts: data extraction and pre-processing of the Chinese stock market dataset, carrying out feature engineering, and stock price trend prediction model based on the long short-term memory (LSTM). We collected, cleaned-up, and structured 2 years of Chinese stock market data. We reviewed different techniques often used by real-world investors, developed a new algorithm component, and named it as feature extension, which is proved to be effective. We applied the feature expansion (FE) approaches with recursive feature elimination (RFE), followed by principal component analysis (PCA), to build a feature engineering procedure that is both effective and efficient.

References:-

* Atsalakis GS, Valavanis KP. Forecasting stock market short-term trends using a neuro-fuzzy based methodology. Expert Syst Appl. 2009;36(7):10696–707.
* Ayo CK. Stock price prediction using the ARIMA model. In: 2014 UKSim-AMSS 16th international conference on computer modelling and simulation. 2014. <https://doi.org/10.1109/UKSim.2014.67>.
* Brownlee J. Deep learning for time series forecasting: predict the future with MLPs, CNNs and LSTMs in Python. Machine Learning Mastery. 2018. <https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/>