**Deep Learning using Big Data**

Research Question:

An exploration of the predictive power of social news sentiment on stock market prices.

**Overview**

This paper takes a primary focus on the predictability of language, particularly through social media blog posts, in future stock market price movements and whether the development of sophisticated neural networks, comprised of market sentiment, can provide a return for investors. The motivation behind this research focus was to test the efficient market hypothesis and factor in market sentiment as a potential determinant of price movement. The three types of Neural Networks explored in this research paper are Artificial Neural Networks (ANNs), Recurrent Neural Networks (RNNs), and Convolutional Neural Networks (CNNs).

The purpose of this literature review is to summarize and comment on existing literature in the areas of; applications of pruning methods related to different neural network architectures, deep learning for sentiment analysis, and methods and tools for big data storage. Following a critical evaluation of sources, the aim is to outline a primary motivation for research, assess deep learning model architectures (pruning methods) related to a specific application field, analyze the data storage requirements of this project and to identify any potential research gaps in these fields.

Sentiment is associated with different attributes and, as such, the literature points towards no universal definition. Schleifer and Summers (1990) define sentiment as an investors overall attitude toward financial markets, DeLong et al. (1990) defines it as investors’ formation of beliefs about future cash flows and investment risks that are not justified by existing facts, and Brown and Cliff (2004) are of the opinion it represents the ‘expectations of market participants relative to a norm: a bullish (bearish) investor expects returns to be above (below) average, whatever average may be’. A myriad of research papers criticizes the efficient market hypothesis (EMH), the rational behavior of investors, questioning the ‘random walk of future asset prices’ and whether or not other factors can act as future indicators for stock price movements. Behavioral finance argues that sentiment proxies may be candidates for explaining subtle pricing anomalies. Lee et al. (1991) argues that sentiment may be a candidate explanation for different asset valuations across investors. Trading strategies, such as the one suggested by Zhang and Skiena (2010) have been developed utilizing natural language processors (NLPs) on blog/news data to generate consistent returns for investors. Katayama and Tsuda (2020) investigated whether a trading strategy based on a sentiment analyzer using deep learning can be profitable using Nikkei Telecom from the Nikkei Shimbun morning edition. Consequently, it is worthwhile to investigate whether deep learning models based on market sentiment can predict future stock price movements.

Deep learning (DL) is the subfield of artificial intelligence that focuses on creating large neural network models that are capable of making accurate data-driven decisions, particularly suited to contexts where the data available is complex and large (Kelleher, DL). There have been three waves of developments in DL: cybernetics in the 1940s-1960s, connectionism in the 1980s-1990s and the current resurgence beginning in 2006 (Goodfellow, DL). DL has attracted much attention from the academic community due to its state-of-the-art performance in many research domains such as speech recognition (G. E. Dahl, D. Yu, L. Deng and A. Acero, "Context-dependent pretrained deep neural networks for large-vocabulary speech recognition", IEEE Trans. Audio Speech Lang. Process., vol. 20, no. 1, pp. 30-41, Jan. 2012), (G. Hinton et al., "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups", IEEE Signal Process. Mag., vol. 29, no. 6, pp. 82-97, Nov. 2012.), collaborative fultering (R. Salakhutdinov, A. Mnih and G. Hinton, "Restricted Boltzmann machines for collaborative filtering", Proc. 24th Int. Conf. Mach. Learn., pp. 791-798, 2007.), and computer vision (D. Cireşan, U. Meler, L. Cambardella and J. Schmidhuber, "Deep big simple neural nets for handwritten digit recognition", Neural Comput., vol. 22, no. 12, pp. 3207-3220, 2010.),(M. Zeiler, G. Taylor and R. Fergus, "Adaptive deconvolutional networks for mid and high level feature learning", Proc. IEEE Int. Conf. Comput. Vis., pp. 2018-2025, Nov. 2011.). Artificial/Feed-forward Neural Networks (ANNs) consist of an input layer, an output layer and a specific number of hidden layers. Neurons are contained in the input and output layers related to the number of input and output parameters. It is then an objective to determine the number of hidden layers and the number of neurons in each hidden layer. Activation functions are applied by each neuron in the hidden and output layers to the weighted sum of inputs. A Recurrent Neural Network (RNN) is an extension to the traditional ANN except that it is designed for processing sequences of data. The output from the neurons in the current layer are used as feedback to the neurons of the previous layer. RNNs are used to solve problems related to; time series data, text data and audio data. Convolutional Neural Networks (CNNs) are mainly used for image and video processing and are a subtype of Deep Neural Networks (DNNs) inspired by the visual cortex of animals. There have been applications of CNNs for the ordering of text for particular categorizations. Johnson and Zhang (2014) apply CNN to high-dimensional text data for use in classification.

A plethora of research has been carried out applying sentiment and financial market trading activities. Most studies have utilized traditional methods such as conventional classification algorithms and Pearson correlation coefficient. Loughran and McDonald (2011) developed a word list using a sample of the United States’ Security and Exchange Commission (SEC) 10-K filings from 1994 to 2008 categorizing words into six groups; positive, negative, litigious, uncertainty, model strong and model weak. They found significant relations between the categories and file date returns, trading volume, subsequent return volatility, standardized unexpected earnings, and two separate samples of fraud and material weakness. Wang et al. (2014) investigate the quality and impact of content on social investment platforms against historical performance of relevant stocks. Datasets are constructed from 9 years of SeekingAlpha articles and 4 years of StockTwits messages. Pearson’s correlation is computed between the article sentiment and a stock’s future price change. Wang et al. ignores the strength of sentiments and the magnitude of price movements, which will be explored in this paper. Da Silva et al. (2014) uses an ensemble of classifiers to label tweets as positive or negative concerning a query term. Multinomial Naïve Bayes, a Support-Vector Machine (SVM), Random Forest and Logistic Regression form the ensembles. The paper demonstrates that the use of ensembles of multiple base classifiers, combined with scores obtained from lexicons can improve the accuracy of tweet sentiment classification.

The emergence of Big Data has allowed for more powerful tools such as Deep Learning approaches to be utilized.

**Methodology**

*Data Collection & Preparaton*

The data used for the purpose of this research paper takes the form of titles of threads in the popular social news website, Reddit. The specific thread titles are collected from the ‘r/Economics’ community. The community is the largest for economics and serves as a central forum for users to read, discuss, and learn about topics related to the economic discipline. It allows for users to post opinions about current affairs such as immigration, trade, protectionism etc. The data was obtained from a specific thread in the ‘r/pushshift’ community as a torrent. Torrents allow for the download of larger files faster than traditional downloading methods. Once downloaded, the file is in a format called zstandard compressed ndjson and is extracted using the 7-Zip software to decompress the file. The file is decompressed into a 525MB text file with separate ‘JSON’ objects on each line of the text file.

Each line of the ‘Economics-submissions.txt’ file contains information such as the author’s username, the title of the thread, the UTC timestamp it was posted, the number of upvotes etc. To achieve the purpose of this research paper, the title of the thread and the UTC timestamp were the components of interest to extract from the text file. Due to the size (525MB) of the text file containing the relevant information for the subreddit threads relating to economics, the HDFS was utilized and interacted with ‘PySpark’, the python library for Apache Spark. The file was stored in HDFS on a virtual machine and was loaded as a ‘PySpark DataFrame’ with a single column, ‘value’. Each observation in ‘value’ represents one of the ‘JSON’ objects of the ‘Economics-Submissions.txt.’ file. After loading in as a dataframe, the titles and UTC timestamps were extracted by user-defined python functions used to transform the ‘value’ column into json objects, which creates key/value pairs. Two separate pyspark dataframes were created for the UTC timestamps and titles and were exported to HDFS. These dataframes were then allowed for sharing between the virtual machine and local machine on a shared drive established between the two, allowing for the utilization of the powerful ‘tensorflow’ library for building neural networks.

The stock price index used for the purpose of this research was the closing price of the S&P 500. The S&P 500 is widely regarded as the best single gauge of large-cap U.S equities. The index includes 500 leading companies and covers approximately 80% of available market capitalization (**SPGlobal**). The data was downloaded via a Bloomberg terminal using the ticker ‘SPX’ for the relevant time.

The dataframes shared from the virtual machine were then loaded into python for data preparation. The two independent dataframes obtained from the reddit text file and the closing prices of the S&P 500 index were merged, resulting in a reddit thread title, and closing price pairing for each date. The resulting dataframe is dated from January 2008 to December 2022. This data forms the dependent variable, the output of various neural networks. The target is the one-day stock price movement direction which will either be ‘Up’ or ‘Down’. Thus, this research paper explores a binary classification problem.

*Sentiment Analysis*

The Natural Language Toolkit (NLTK) package is imported into python which provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum (**NLTK.org**). The Valence Aware Dictionary and sentiment Reasoner (VADER) is used as a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media (**Shelar & Huang, 2018**). VADER utilizes a mix of lexical highlights marked by their semantic direction as either positive or negative. It produces four sentiment measurements from each word grading, a positive, neutral, negative, and compound score. The compound score is a normalized score describing the total amount of the lexicon grades (**Analytics Vidhra**). These compound scores are aggregated for each date by taking the average score and then a sentiment is assigned to a particular date from the group of three based upon the below logic.

|  |  |
| --- | --- |
| **Compound Score** | **Sentiment** |
| >= 0.05 | Positive |
| <= -0.05 | Negative |
| -0.05 < Score < 0.05 | Neutral |

The sentiment assigned to the scores becomes an independent variable that will feed into various neural networks to assess predictability with stock market price movements. Two forms of the sentiment variable are considered as an input, one where the sentiment the day before a stock price movement and another the day of a stock price movement. The one-day lagged stock price market movement direction is also included as an input to decipher whether previous price movements can be attributed to future price movements. Therefore, we have three independent variables to consider as inputs when developing neural networks.

*Neural Network*

As previously mentioned, three different neural network types will be explored and incorporated in this research paper, namely ANN, RNN and CNN. Each of these networks have differing sets of requirements which allow for their efficient use given a particular problem set. These requirements have been optimized for each neural network through hyperparameter tuning and the performance of the networks will be compared through the accuracy metric to decide which neural network architecture is the most effective in classifying binary movements in future stock prices.

**Artificial/Feed-Forward Neural Network**

ANNs have been successfully applied to many diverse fields such as pattern classification, image processing and stock market predictions (**Sazli**) and thus are very attractive to use for learning on any nonlinear problem. These networks, due to their usability are often referred to as Universal Function Approximators and are particularly useful when dealing with tabular data. Shulka, 2019, provides informative detail on the set of requirements to consider when developing neural networks.

An ANN consists of an input layer, a specified number of hidden layers and an output layer. The input layer contains a specific number of neurons related to the number of features used by the network to make predictions. As this paper considers three independent variables, the specified dimension used in the design of the input layer for the ANN is three. The number of hidden layers and neurons within these layers is dependent on the problem, however, generally one to five hidden layers will suffice. For this paper, one hidden layer will be used and the number of neurons in this layer is tuned using the ‘GridSearchCV’ hyperparameter tuning method. Following tuning, the optimal number of neurons was set to 24. An activation function is defined in the hidden layer and is the function used to transform the input layer to the output (Sharma & Athaiya). For ANN this is optimally tuned to the sigmoid function. The sigmoid function is the most widely used activation function as it is a non-linear function. It is continuously differentiable and a smooth S-shaped function. For binary classification one output neuron is used per positive class and the sigmoid activation function is used to limit the output between zero and one. Binary cross-entropy is the default loss function for binary classification problems as it is the preferred function under the inference framework of maximum likelihood. The Stochastic Gradient Descent optimizer is tuned to decide the weights and learning rates of the neural network. The accuracy of the ANN built on the above requirements using three input variables is 54.69% on the training dataset and 53.66% on the testing dataset, slightly better than a random guess.