FORECASTING AND MULTI-CRITERIA DECISION MAKING FOR TRADING AND INVESTMENT IN STOCK MARKET USING SOFT COMPUTING TECHNIQUES

Thesis

Submitted to the

G. B. Pant University of Agriculture and Technology
Pantnagar-263145, U.S. Nagar, Uttarakhand, India

By

Kiran Bisht
Id. No. 53998

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy
(Mathematics)

July, 2022

ACKNOWLEDGEMENTS

First and foremost, I would like to thank the Almighty ***GOD*** *and my loving parents* ***Mr. Jaswant Singh Bisht*** *and* ***Mrs. Tara Bisht*** *for their never ending grace and shower of blessings throughout my research work to complete it successfully.*

I would like to extend my sincere gratitude to my research supervisor, ***Dr. Arun Kumar****, Professor of Mathematics, Statistics and Computer Science, G. B. P. U. A. T. Pantnagar, for the continuous support during my Ph.D. study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor than him for my Ph.D.*

Beside my advisor, I would like to express my gratitude and thanks to the rest of the members of my advisory committee, ***Dr. Manoj Kumar*** *(Professor and Head),* ***Dr. S.B. Singh*** *(Professor) and* ***Dr. Sanjay Kumar*** *(Associate Professor), Department of Mathematics, Statistics and Computer Science, College of Basic Sciences and Humanities for their insightful comments and encouragement.*

It is my privilege to express my heartiest regards and sincere thanks to the entire faculty members ***Dr. A. K. Shukla, Dr. Vinod Kumar, and Dr. R. S. Rajput****, for their cooperation and support throughout the degree programme. I also acknowledge thanks to support staff of computer center and department.*

I am extremely thankful to my brother ***Kamal Bisht*** *and my sister* ***Kaamini Bisht*** *for their love and understanding.*

A special thanks and debt of gratitude is owed towards my fiancé ***Manu Singh Beniwal*** *for bringing smile on my face during every ups and downs and for always being a big support.*

There are no words to pay my regards to many unforgettable persons during my stay in Pantnagar. Special thanks to my fellow batchmates and friends ***Dr. Himani Pant, Makrand Dhyani, Kirti Nagarkoti, Pragati Joshi and Manjula Tiwari*** *for the stimulating discussions about research and life, for the sleepless nights we were working together before deadlines, and for all the fun we have had in the last four years. You guys have made my stay here memorable.*

I sincerely thank to all my seniors specially ***Dr. Soni Bisht****,* ***Dr. Krishna Kumar Gupta, Dr. Ashish Mishra, Dr. Neelam Chantola, Dr. Akanksha Singh, Dr. Savan Kumar Rawat, Dr. Himanshu Upreti, Dr. Shweta Agarwal, Dr. Manish Pant*** *and most loving juniors* ***Ankita, Amisha, Shivani, Laxmi, Yaseen, Mintu, Garima, Priyanshu, Vaibhav, Ayush, Shiv Pratap, Tanya and Geeta*** *for their help and support during whole period of my work.*

Lastly, I shall always remain thankful to all those who could not have find separate names but had directly or indirectly helped me. This research work is dedicated to my Mummy and Papa.

(Kiran Bisht)
Authoress

Pantnagar
July, 2022

CERTIFICATE-I

This is to certify that the thesis entitled **"FORECASTING AND MULTI-CRITERIA DECISION MAKING FOR TRADING AND INVESTMENT IN STOCK MARKET USING SOFT COMPUTING TECHNIQUES"** submitted in partial fulfillment of the requirements for the degree of **Doctor of Philosophy** with major in **Mathematics** and minor in **Computer Science**, of the College of Post-Graduate Studies, G. B. Pant University of Agriculture and Technology, Pantnagar, is a record of *bona fide* research carried out by **Ms. Kiran Bisht, Id. No. 53998** under my supervision and no part of the thesis has been submitted for any other degree or diploma.

The assistance and help received during the course of this investigation have been acknowledged.

Pantnagar
July, 2022

(Arun Kumar)
Chairman
Advisory Committee

CERTIFICATE-II

We, the undersigned, members of the Advisory Committee of **Ms. Kiran Bisht, Id. No. 53998,** a candidate for the degree of **Doctor of Philosophy** with major in **Mathematics** and minor in **Computer Science**, agree that the thesis entitled **"FORECASTING AND MULTI-CRITERIA DECISION MAKING FOR TRADING AND INVESTMENT IN STOCK MARKET USING SOFT COMPUTING TECHNIQUES"** may be submitted in partial fulfillment of the requirements for the degree.

(Arun Kumar)
Chairman
Advisory committee

(Manoj Kumar)
Member

(S.B. Singh)
Member

(Sanjay Kumar)
Member

14.7.2022

(Ex-Officio member)
Head of the Department

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACO	Ant Colony Optimization
ADX	Average Directional Index
AHP	Analytic Hierarchy Process
AIC	Akaike Information Criterion
ANN	Artificial Neural Network
ANP	Analytic Network Process
BCM	Base-Criterion Method
BPA	Basic Probability Assignment
BSE	Bombay Stock Exchange of India
BWM	Best–Worst Method
CNN	Convolutional Neural Network
CoCoSo	Combined Compromise Solution
COPRAS	Complex Proportional Assessment
CRITIC	Criteria Importance Through Intercriteria Correlation
D/E	Debt Equity Ratio
DBN	Deep Belief Network
DEMETAL	Decision-Making Trial and Evaluation Laboratory
DL	Deep Learning
DNN	Deep Neural Network
DRL	Deep Reinforcement Learning
D-S Theory	Dempster-Shafer theory
EDAS	Evaluation based on Distance from Average Solution
ELECTRE	Elimination Et Choice Translating Reality
EMA	Exponential Moving Average
FCM	Fuzzy c-Means Clustering

FDM	Fuzzy Delphi Method
FLR	Fuzzy Logic Relation
FMCG	Fast Moving Consumer Goods
FOD	Frame of Discernment
FPE	Final Prediction Error
FTSF	Fuzzy Time Series Forecasting
GA	Genetic Algorithm
GMIR	Graded Mean Integration Representation
HFS	Hesitant Fuzzy Set
IFS	Intuitionistic Fuzzy Set
IT	Information Technology
LR test	Likelihood Ratio test
LSTM	Long Short-Term Memory
MABAC	The Multi-Attributive Border Approximation area Comparison
MADM	Multiple Attribute Decision Making
MAUT	Multi-Attribute Utility Theory
MCDA	Multi-Criteria Decision Aid
MCDM	Multi-Criteria Decision Making
MEREC	MEthod based on the Removal Effects of Criteria
MLP	Multilayer Perceptron
MODM	Multiple Objective Decision Making
MODRL	Multi-Objective Deep Reinforcement Learning
MOORA	Multi-Objective Optimization Ratio Analysis
MPT	Modern Portfolio Theory
NFS	Neutrosophic Fuzzy Set
NIS	Negative Ideal Solution
NOI	Number of Intervals

NSE	National Stock Exchange of India
P&L	Profit and Loss
P/B	Price to Book Value ratio
P/E	Price to Earnings ratio
PFS	Pythagorean Fuzzy Set
PIS	Positive Ideal Solution
PROMETHEE	Preference Ranking Organization METHod for Enrichment of Evaluations
PSO	Particle Swarm Optimization
RF	Random Forest
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROE	Return on Equity
RSI	Relative Strength Indicator
SAW	Simple Additive Weighting
SC	Schwarz Information Criterion
SMA	Simple Moving Average
SMAPE	Symmetric Mean Absolute Percentage Error
SR	Sharpe Ratio
SRS	Sector's Relative Strength
SSE	Sum Squared Error
SVM	Support Vector Machine
SVTN	Single Valued Triangular Neutrosophic Number
SVTNWAO	Single Valued Triangular Neutrosophic Weight Averaging Operator
SVTNWPAO	Single Valued Triangular Neutrosophic Weight Power Averaging Operator
SWARA	Stepwise Weight Assessment Ratio Analysis

TFN	Triangular fuzzy Number
TOPSIS	Technique for Preference by Similarity to the Ideal Solution
TP	Total Profit
TT	Total no. of Transactions
UOD	Universe of Discourse
VAR	Vector Auto Regression
VECM	Vector Error Correction Model
VIKOR	Vlse Kriterijumska Optimizacija I Kompromisno Resenje
WASPAS	Weighted Aggregates Some Product Assessment
WPM	Weighted Product Measure
WSM	Weighted Sum Measure

Introduction

Chapter 1 INTRODUCTION

With the emerging global economy, stock market investment has piqued investors' attention as it offers a flexible and transparent means to disperse risk while increasing the potential for profit. In the stock market, decision making entails building and overseeing a selection of investments that meets the financial goals and risk tolerance of an investor. It is a place where investors and traders can legally gamble on the values of stocks to gain some benefit or sometimes lose to the plummeting wave of the highly volatile market. The stock market goes up and down, and securities, commodities, currencies, and bond prices behave likewise. It gives investors the chance to make money if they know how to play smart in the game of stock market analysis. The performance of a stock is defined by a multitude of elements based on risk-return measures. The main target of the stock market investor is to make such decisions that end up gaining maximum profit while taking minimum risk during investment. To achieve an investor's goal following objectives should be considered to make reliable decisions:

i. How to tackle the uncertainty of the stock market?

ii. How to handle the different conflicting factors of stock market while making decisions?

iii. How to maximize profit restraining the level of the risk?

The complexity and dynamism of the stock market with noise, ambiguity, and chaos are the main challenge in modelling the financial decision making procedures. Soft computing techniques are progressively gaining presence in the financial world. These are fledging approaches providing imprecise but usable solutions to complex computational problems parallel the remarkable ability of the human mind to reason and learn in an environment of uncertainty and opacity. Soft computing is a set of algorithms, including neural networks, fuzzy logic, and evolutionary algorithms. These algorithms are tolerant of imprecision, uncertainty, partial truth, and approximation. Compared to traditional techniques for applying to the financial market, soft computing gains the advantage of accuracy and speed.

1.1 Deep learning

Deep learning (DL) is a specialized form of machine learning in artificial intelligence, motivated by the structure of human brain. DL algorithms attempt to draw conclusions similar to humans that analyze data with a continuous logical structure. To carry out this, DL algorithms use a multi-layer structure called neural network. A neural network is a machine designed to model how the brain performs a particular task. A neural network takes data samples instead of entire data sets to arrive at solutions, which saves time. It changes its structure based on external or internal information that passes through the network during the learning phase. The neural network consists of three layers, as shown in Figure 1.1.

- **Input layer:** It carries initial data for the neural network.
- **Hidden layer:** It is the intermediate layer between the input layer and the output layer. This is the place where all kinds of computations are performed.
- **Output layer:** In this layer, the result is produced for the given input.

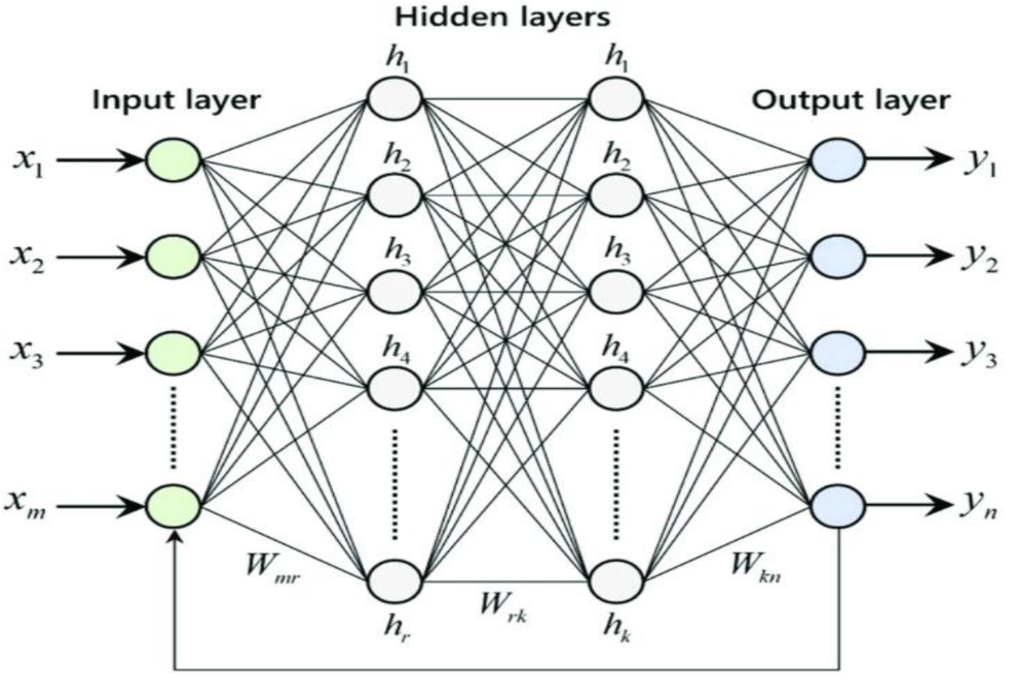
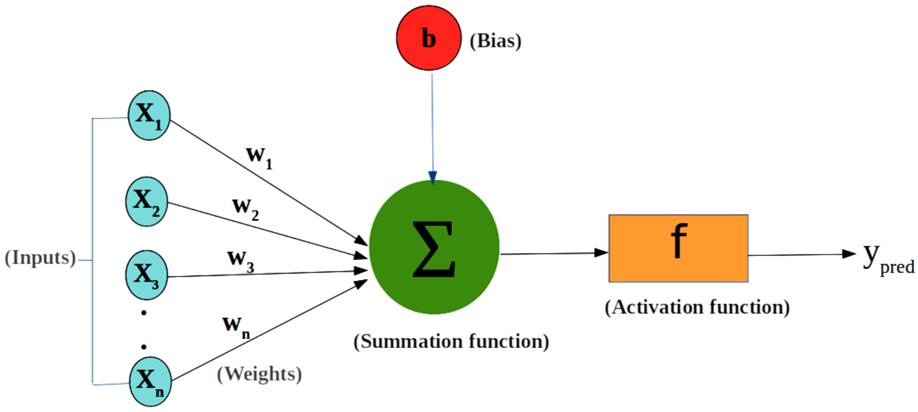


Figure 1.1: Neural network

A neural network is built up by a network of computing units known as neurons. These computing units are represented as nodes in the network. A neuron receives inputs, performs some calculations with the inputs, and generates output. Each neuron is connected to another neuron through a connection link. Each connection link is connected to a weight that contains information about the input. The processing of the network is as follows:

- Each input is multiplied by a weight.
- All weighted inputs are sum up together with a bias b (unique to each neuron layer).
- The sum is passed through an activation function.
- Finally, output signals are received that are generated after combining the input signal and the activation rule.



$$y = f(x_1 * w_1 + x_2 * w_2 + \cdots + b)$$

Figure 1.2: Working of neural network

An activation function is a mathematical formula combined into an artificial neural network (ANN) to help learn complex patterns in the network. Various activation functions (given in Table 1.1) are available according to the input values. A bias value permits the activation function to be shifted to the left or right, and it helps to be a better fit for the data. The working of a neural network is depicted in Figure 1.2. After a network is structured for a particular application, that network is ready to be trained. The initial weights are chosen randomly to initiate this process. Then, the training or learning process begins. This training or learning can be classified into three classes:

- **Supervised learning:** In supervised learning, both the input and output are provided to the network. The network then performs computation with the input and compares its resulting output against the required output.
- **Unsupervised learning:** In this type of learning, only input is provided to the network. The network has to decide for itself which feature it will use to group the input data. This is also called an adaptive learning.
- **Reinforcement learning:** It is a reward based learning. It is about taking suitable action by an agent to maximize reward in a particular situation. Reinforcement agent decides what to do to perform the given task. In the absence of training dataset, it is bound to learn from its experience.

Table 1.1: Activation functions

Unit step (Heaviside)	$f(z) = \begin{cases} 0, & z < 0 \\ 0.5, & z = 0 \\ 1, & z > 0 \end{cases}$
Signum	$f(z) = \begin{cases} -1, & z < 0 \\ 0, & z = 0 \\ 1, & z > 0 \end{cases}$
Linear	$f(z) = z$
Piece-wise linear	$f(z) = \begin{cases} 1, & z \geq \frac{1}{2} \\ z + \frac{1}{2}, & -\frac{1}{2} < z < \frac{1}{2} \\ 0, & z > 0 \end{cases}$
Sigmoid	$f(z) = \frac{1}{1 + e^{-z}}$
Hyperbolic tangent	$f(z) = \frac{e^{z} + e^{-z}}{e^{z} + e^{-z}}$
Rectified linear unit (ReLU)	$f(z) = \max(0, z)$
Softplus	$f(z) = \ln(1 + e^{z})$

The working system of DL is based on a set of algorithms that attempt to create high-level abstraction models in data using a deep graph with multiple processing layers, consisting of multiple linear and non-linear transformations. The term "deep" often defines the number of hidden layers in a neural network. Conventional neural networks have only 2-3 hidden layers, whereas deep networks can have up to 150 hidden layers. DL models are like rocket engines and massive amounts of data are like fuel that we can feed to these algorithms.

Financial data is computationally intensive, extremely nonlinear, and could appear completely random at times. Deep neural networks (DNN) are adept at dealing with this type of data. However, the main issue arises in implementing these models in a live trading system, as there is no guarantee of stationarity on adding new data. This issue is also combated by using DNNs, which do not require stationarity. The supervised and unsupervised models are the widely used models in finance. State-based models, econometric models, or even stochastic models are marred by the problems of over fitting, heuristics, and poor out-of-sample results because the financial domain is hugely complex and nonlinear with a plethora of factors influencing each other. DL models are proven capable of learning from large-scaled unlabelled data, forming nonlinear relationships, forming recurrent structures, and can be easily tweaked to avoid over-fitting. These models are widely used in price forecasting, portfolio designing, portfolio optimization, fraud management, policy formulation, risk management, algorithm trading, and many other applications.

1.2 Fuzzy sets

Fuzzy set theory was firstly proposed by **Zadeh (1965)** as an extension of classic set theory. In classical set theory, the membership degree of an element in a set is evaluated based on the binary terms determining whether it belongs to the set or not. Human language is not easily transformed into the absolute terms 0 and 1. A fuzzy set is a mapping of a set of real numbers onto membership values that lie in the range [0, 1].

Definition (Fuzzy set): A fuzzy set A on a finite reference set X is characterised by a membership function $\mu_A: X \to [0,1]$ and is defined as the following mathematical object:

$$A = \{(x, \mu_A(x)) | x \in A\}$$

Fuzzy mention the things which are obscure. Working system of fuzzy sets is retraced uncertainty due to the imprecise and vague information but the centre of attention of the fuzzy sets is only the membership degree of vague events. Fuzzy set can't deal with falsity degree and indeterminacy degree of vague events as independent components. Several generalizations of fuzzy sets have developed such as intuitionistic fuzzy set (IFS), neutrosophic fuzzy set (NFS), type-2 fuzzy set, fuzzy multiset, pythagorean fuzzy set (PFS) and hesitant fuzzy set (HFS). IFS have three main parts: membership degree, non-membership degree and hesitancy degree. IFS can only deal with imperfect data but not with indeterminate and inconsistent data which exists very often in real world problems.

In NFS, the amount of indeterminacy is revealed and truth-membership, indeterminacy-membership, and falsity-membership are independent. Type-2 fuzzy set permits the membership of a given element as a fuzzy set. In fuzzy multiset, the elements could be encore more than once. PFS improves IFS by expanding the range of conditions to be modeled. When people make a decision, they usually hesitate for one thing or another, making it difficult to reach a final agreement. HFS is very powerful tool to get the optimal alternative in a decision making problem with multiple attributes and multiple persons. HFS permits the membership having a set of possible values. HFS has a unique characteristic that its basic element could manifest the assessment values of different decision makers on the same option under a certain criterion.

Different fuzzy sets have been successfully applied in the field of finance due to their ability to address imprecise, incomplete and vague data. Fuzzy logic based methodologies has also been used in the field of banking, although to a lesser extent, with particular relevance to areas such as risk management and credit scoring. However, with regard to the specific field of banking crises, the footprint of fuzzy logic has been almost non-existent. This absence could be seen as a contradiction, given the properties of fuzzy logic, which seem to fit particularly well in complex and uncertain environments. Some applications of fuzzy logic in finance are fuzzy averaging for forecasting stock prices, financial decision making in fuzzy environment, fuzzy logic control for business and so on.

1.3 Evolutionary algorithms

Evolutionary algorithms are simple problem-solving techniques, inspired by biological evolution. These algorithms are based on an artificially simulated process of natural selection, or survival of the fittest, known as Darwinian evolution. Evolutionary algorithms have proven to be a versatile general purpose search and optimization tool with applications in a variety of fields. On its way to identifying optimal solutions, a typical evolutionary algorithm takes a number of steps. Different researches describe different number of steps, but they all include the characteristic procedures that an evolutionary algorithm runs through. These will be covered in the following:

i. **Initialization:** The first step is to create an initial population of solutions, or chromosomes. Typically, the chromosomal population is chosen at random, such as by tossing a coin or letting a machine to generate random numbers.

ii. **Fitness evaluation:** The fitness function (also known as the objective function or evaluation function) is used to map an individual's chromosomes or bit strings into a positive integer, which represents that individual's fitness. The fitness function simulates the natural environment by assigning a fitness rating to solutions.

iii. **Selection:** In the third step, the evolutionary algorithm begins to reproduce. The individuals that are going to become parents of the next generation are selected from the initial population of chromosomes. Selection determines which individuals of the population will have all or some of their genetic material passed on to the next generation of individuals. The purpose of the selection procedure is to offer the most suited chromosomes a better chance of replicating.

Evolutionary algorithms appear to be particularly well-suited for financial modelling applications because of three reasons:

i. They are payoff-driven. Payoffs can mean improvements in predictive power or returns over a benchmark. There is an excellent match between the problem-solving tool and the issues to be addressed.

ii. Evolutionary algorithms are inherently quantitative. They are well-suited to parameter optimization.

iii. Evolutionary algorithms allow a wide variety of extensions and constraints those traditional methods unable to provide. They are robust, revealing a remarkable balance between efficiency and efficacy.

1.4 Stock price forecasting

Stock price forecasting is a prevalent and significant aspect of financial and academic studies since it assists investors in making better financial decisions. Financial time series forecasting is the most common and fundamental method used to perform this task. Stock prices reflect all unknown information, and the price movement responds to news or events. Even though some variations exist, the main focus is on predicting the next movement of the underlying asset. Due to the stock market's dynamics and ambiguity, the prediction of financial time series is described as one of the most challenging tasks of time series prediction. Many soft computing tools like neural networks, fuzzy systems, and wavelets have been popular over the decades to manipulate the data. But it has been challenging to handle big data and heterogeneous data for researchers and scientists. To overcome this problem deep learning, introduced in 2009-2010, becoming popular now a days. Figure 1.3 presents the flowchart of two widely used techniques for financial time series forecasting in recent years.

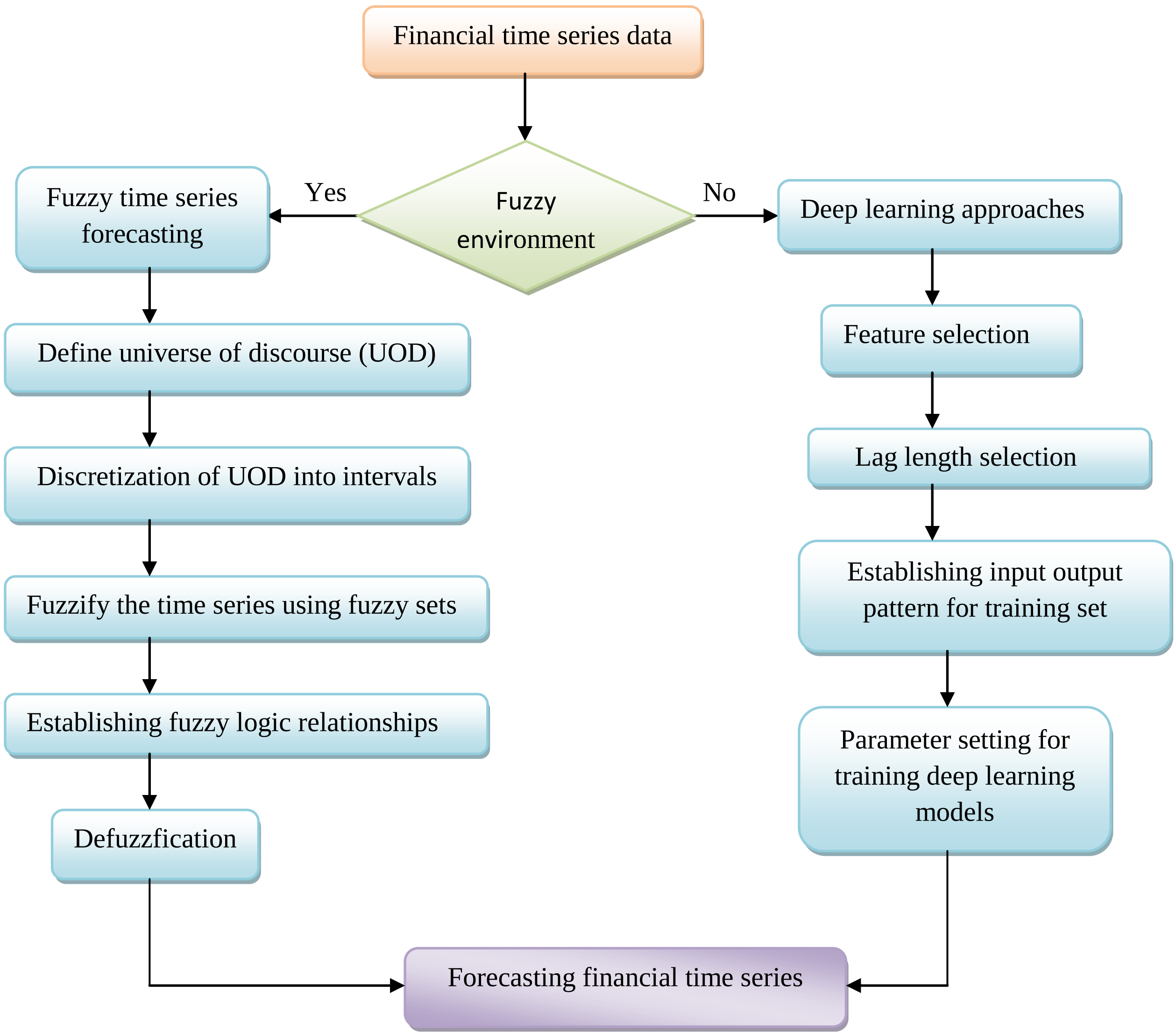


Figure 1.3: Flow chart of financial time series forecasting

1.4.1 Fuzzy time series forecasting

Fuzzy time series forecasting (FTSF) methods are non-parametric and very popular among researchers for predicting future values as they are not based on the strict assumptions of traditional forecasting methods. Non-stochastic methods of fuzzy time series forecasting have been preferred by researchers over the years because these methods are capable of dealing with real-life uncertainties and provide a significant forecast. There are generally four factors that determine the accuracy of the FTSF method (i) number of intervals (NOIs) and length of intervals to partition universe of discourse (UOD), (ii) fuzzification rules or feature representation of crisp time series, (iii) method of establishing fuzzy logic relations (FLRs), and (iv) defuzzification rule to get crisp forecasted value. Definitions of some terminologies used in FTSF are given below:

Definition (Universe of discourse): The universe of discourse (UOD) is a set defined as X containing all the elements of a time series. It could be discrete or continuous. If X is continuous, it can be defined as $X = [a, b]$ here, a be the lower bound and b be the upper bound of the time series.

Definition (Fuzzy set): Let X be a discrete and finite set such that $X = \{x_1, x_2, \ldots x_n\}$. A fuzzy set A of X is expressed as:

$$A = \mu_A(x_1)/x_1 + \mu_A(x_2)/x_2 + \ \ldots + \mu_A(x_n)/x_n \quad (1.1)$$

here, μ_A is the membership function of fuzzy set A, $f_A: X \to [0,1]$ and $f_A(x_i)$ denote the degree of belongingness of element x_i in the fuzzy set A.

For continuous X fuzzy set A of X is expressed as:

$$A = \int \mu_A(x_i)/x_i \quad (i = 1,2 \ldots n) \quad (1.2)$$

Definition (Fuzzy time series): Fuzzy time series (FTS) $F(t)$ is a series of fuzzy sets $f_i(t)(i = 1,2,3 \ldots n)$ defined on universe of discourse $Y(t)(t = 0,1,2,3 \ldots)$.

Definition (Fuzzy logic relation): Suppose $F(t)$ is caused by $F(t-1)$ then the relation $R(t, t-1)$ between $F(t)$ and $F(t-1)$ called fuzzy logic relation (FLR) which is defined as $F(t) = F(t-1)°R(t, t-1)$ here, ° is an arithmetic operator. This type of FLR can be represented as $F(t-1) \to F(t)$ and FTS with such relation called first order fuzzy time series.

Definition (High order fuzzy time series): If $F(t)$ is determined by $F(t-1), F(t-2), \ldots F(t-n)$, then FLR is defined as $F(t-n) \ldots F(t-2)\ F(t-1) \to F(t)$ and the FTS is called n^{th} order fuzzy time series.

Hybridization of ANN with FTS is a significant development in the domain of forecasting. It is an ensemble of the advantages of ANN and FTS, with the shortcomings of one technique being replaced by the strengths of the other. ANN can be used in different steps of the FTS modelling approach (such as defining FLRs, partitioning UOD and defuzzification).This incorporates ANN's significant benefits, such as parallel processing, big data set handling, and quick learning capabilities, among others. Fuzzy sets are used to handle imprecise/uncertain variables as well as linguistic variables. Aside from these benefits, the FTS-ANN hybridization aids in the development of sophisticated decision-making systems.

1.4.2 Time series forecasting using deep learning

The recent advent of DL models for financial prediction has strengthened the financial community. New deep learning methods have been developed using additional financial time series data and other deep architectures. Even though there are several subtopics of this general problem, such as stock price forecasting, index prediction, forex price prediction, commodity (oil, gold, etc.) price prediction, bond price forecasting, volatility forecasting, and crypto currency price forecasting, the inherent dynamics are the same in all of these applications. Some definition related to DL models for time series forecasting are given below:

Definition (Forecasting problem): Given a historic time series data $S = \{(X_1, y_1), (X_2, y_2) \ldots (X_t, y_t)\}$ here, y_t is the response variable to the forecast at time t, and $X_t = (x_{t1}, x_{t2}, \ldots x_{tm})$ represents the value of relevant factors. We assume that a function dependency $f: S \to y_{t+n}$ exists, here, y_{t+n} denotes the response variable n times later than time t. Then the forecasting problem is to find a proper f' approximating f, and use f' to forecast the values of the response variable in the future.

Definition (Feature selection): Given a historical time series dataset $S = \{(X_1, y_1), (X_2, y_2) \ldots (X_t, y_t)\}$ and denote $X = (x_1, x_2, \ldots x_m)$ as the original feature set. Feature selection is the process of generating an optimal feature subset $X' \subseteq X = (x_1, x_2, \ldots x_m)$, based on which f' will have the optimal forecast accuracy on the future dataset.

Definition (Lag length): Lag is the time length of the input vector in DL models (e.g., 30 d means the input vector has a 30 day window), and it shows how far into the future the model predicts.

Different kinds of DL models such as deep multilayer perceptron (DMLP), recurrent neural network (RNN), long-short term memory (LSTM), convolution neural network (CNN), restricted boltzmann machines (RBMs), support vector machine (SVM), deep belief network (DBN), autoencoder (AE) and their combinations are used for financial time series forecasting.

1.5 Decision making in stock market

In the stock market, there are different trading strategies (such as intraday trading, scalping trading, swing trading, position trading or long term investment). In terms of the decision-making process, these trading strategies can be divided into two categories: (short term trading and long term investment). The decision making process in both categories is given in Figure 1.4.

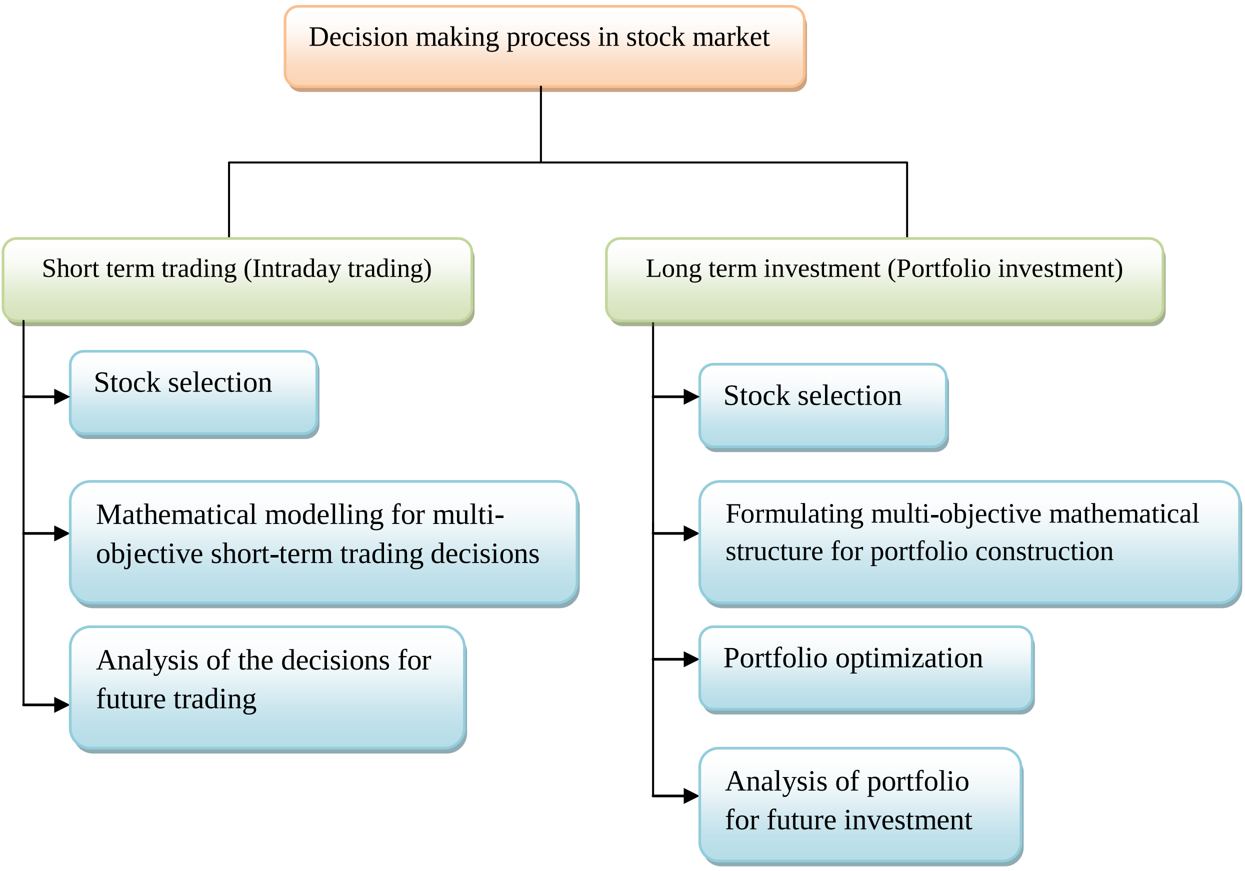


Figure 1.4: Flow chart of decision making in stock market

1.5.1 Stock selection

Stock selection is the first and foremost step in the stock market investment process. Numerous stocks are listed in the stock market and finding an appropriate stock that meets investment goals of an investor is a challenging task. The selection of stocks depends upon the type of trading strategy and the investor's goal. Stock selection is made by analyzing the stock's performance under different trading strategies. There are two widely popular ways of assessing the performance of the stock (i) Stock price forecasting, (ii) Fundamental analysis and/or technical analysis of the stock.

Fundamental analysis or technical analysis shifts the key ratio of a market to shape its financial health and takes several fundamental indicators (such as price to earnings ratio (P/E), price to book value (P/B), debt to equity ratio (D/E), return on equity (ROE), long term beta (LTB), earning per share (EPS) and many more) or technical indicators (such as simple moving average (SMA), relative strength index (RSI), moving average convergence divergence (MACD), stochastic oscillator, rate of change (ROC), commodity channel index (CCI), bollinger bands etc.) into account that determine the performance of a stock in future. It is possible to predict changes through a fundamental approach before they play on charts. The fundamental analysis (or performance) of a stock depends on several criteria based on the risk-return measure. It is a multi-faceted problem as there are many stocks to invest in and more than one criterion. So, a multi-criteria decision-making model is used to solve the underlying multi-criteria nature of this problem.

1.5.2 Multi-criteria decision making

Multi-criteria decision making (MCDM) is a significant part of operation research. When we make decisions, we go through different stages such as discerning the problem, identifying potential solutions, selecting criteria, determining the results of each solution, enumerating the solutions, and selecting the best solution, respectively. MCDM methods have been gaining phenomenal popularity and widespread applications. MCDM problems could be classified into two classes: Multiple-objective decision making (MODM) for designing the best solution and multiple-attribute decision making (MADM) for choosing the best alternative. The MODM method is preferably used for continuous optimization problems where the number of alternatives is infinite, i.e., the decision space is continuous. In general, it is suitable for designing the best alternative planning decision problems in which alternatives are not predetermined. However, instead, a set of objective functions is optimized subject to constraints. According to computational time and solution type, the MODM methods are solved utilizing mathematical programming and heuristic algorithms.

On the other hand, MADM methods refer to discrete representations of a problem with many conflicting criteria and a limited number of alternatives. MCDM is commonly used to describe the discrete MADM. There are two main goals for solving practical problems by MCDM methods: (i) calculating the optimum weight of the criterion; (ii) setting the rank of the alternatives. Scientists and researchers gave new insight into how to mend decision-making quality over the past decades. They cited several methods for weight processing of criteria and alternatives, and for ranking alternatives. Better decisions can be made by well-structuring complex problems and considering various rules. The inherent quality of MCDM makes it attractive and useful for real life. MCDM format is a controlled decision tool for calculating the weight of the evaluation criterion and ranking the alternatives present in problems with quantitative and qualitative criteria. When more than one decision makers are involved in MCDM, the problem is called as group MCDM or GMCDM. The terminologies which are used in MCDM can be described as follows:

- **Alternative:** One of the things, propositions, or courses of action that can be chosen. MCDM consists of a number of predefined, independent and limited options, and each of them contains the level of desired characteristics of the decision makers.
- **Criterion:** Criterion means “a standard of judging” based on which the effectiveness of a particular alternative can be measured. The application of MCDM depends on the calculation of the criterion weight, which is important for selection and sorting. It is divided into objective and attribute.

- **Attribute:** This is the quality that must be in an alternative. Based on the information of the decision makers, each alternative is associated with some related attributes.
- **Objective:** An objective is a goal that is pursued until it is achieved.
- **Decision matrix:** An MCDM problem with m alternatives and n criteria is described by a matrix $A = [\tilde{a}_{ij}]$ of order $m \times n$ in which $\tilde{a}_{ij}$ represents the performance of alternative i corresponding criterion j.

$$A = \begin{bmatrix} \tilde{a}_{11} & \tilde{a}_{12} & \tilde{a}_{13} & \dots & \tilde{a}_{1n} \\ \tilde{a}_{21} & \tilde{a}_{22} & \tilde{a}_{23} & \dots & \tilde{a}_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{a}_{m1} & \tilde{a}_{m2} & \tilde{a}_{m3} & \dots & \tilde{a}_{mn} \end{bmatrix}$$

- **Beneficial-criteria (profit-criteria):** Criteria with positive desirability from the decision maker's point of view; i.e., their higher amount is more favorable to the decision maker. For instance, income.
- **Non-beneficial criteria (cost-criteria):** Criteria with negative desirability from the decision maker's point of view; i.e., their lower amount is more favorable to the decision maker. For instance, cost.
- **Dependent criteria:** Criteria which are correlated to at least one of the other criteria.
- **Independent criteria:** Criteria that are absolutely not correlated to each other.
- **Qualitative criteria:** These are criteria that have no unit of measure and cannot be represented numerically.
- **Quantitative criteria:** These are criteria that have a unit of measure and can be displayed numerically.
- **Weight:** Weight defines the importance of the criteria. The more the weight, the more will be its importance. Methods of finding weights are classified into two parts: subjective and objective.
- **Subjective weight:** The determination of criteria weights is dependent on the preferences of decision-makers.

- **Objective weigh:** Objective weights are determined based on the initial data or decision matrix and removes the impact of direct judgments on the significance of criteria expressed by decision-makers.

1.5.2.1 Different methods of finding relative weights of criteria

The foremost important thing is obtaining the optimal criterion weights, which have been the foundation for introducing many MCDM methods. DEMETAL (Decision-Making Trial and Evaluation Laboratory), AHP (Analytic Hierarchy Process), ANP (Analytic Network Process), CRITIC (Criteria Importance Through Intercriteria Correlation), SWARA (Stepwise Weight Assessment Ratio Analysis), BWM (Best–Worst Method) and BCM (Base-Criterion Method) are some methods for criteria weight assignment.

The DEMATEL method was developed by **Gabus and Fontela (1972)** as a compensatory method and used in MCDM to detect complex relationships and construct interrelations between various criteria. The AHP method was proposed by **Saaty (1970)** which is a powerful but simple method for managing and analyzing complex decisions using mathematics and psychology. In AHP, the relative importance of criteria and alternatives are executed by decision-makers using pairwise comparison based on a 1-9 scale. Despite the simplicity and prominence of AHP, due to inconsistencies in the pairwise comparison matrix, it is incapable of adequately handling the ambiguity of expert preferences.

The ANP method proposed by **Saaty (2005)** is a more common form of AHP based on the analysis operated by the human brain for complex issues with non-hierarchical structures. In this method, independence of criteria is not necessary. CRITIC method is developed by **Diakoulaki *et al.* (1995)** as a compensatory method. This method is introduced to calculate the criteria weights using a decision matrix. The procedure of achieving objective weight involves the standard deviation of criteria and the correlation among decision criteria. The SWARA method is introduced by **Keršulienė *et al.* (2010)**, in which attributes are independent of each other and attributes are compensatory. In this method, the initial priority of the independent criteria is determined based on the opinion of the decision-makers. Then the relative weight of each decision criteria is calculated.

BWM has been developed by **Rezaei (2015)** to reform the consistency of pairwise comparisons by slackening the number of pairwise comparisons. This method calculates the weight of criteria based on the best and worst criteria. BWM outcomes are more real than AHP. However, there is an inconsistency in BWM that may affect decisions. BCM proposed by **Haseli *et al.* (2020)** is a preferable walkway to determine the weights of the criteria. Firstly, the decision-maker selects the base criterion (preferential), and after that, pairwise comparisons between base criteria and other criteria are attained based on a 1/9 to 9 scale. It can derive criteria weights with fewer pairwise comparisons than the existing MCDM methods. The final weights determined through BCM are authentic as the comparisons are fully consistent, whereas other traditional MCDM techniques such as BWM and AHP have a low inconsistency ratio.

1.5.2.2 Different methods for ranking the alternatives

The most widely applied methods developed by the researchers for ranking the alternatives are ELECTRE (ELimination Et Choice Translating REality), MAUT (The Multi-Attribute Utility Theory), TOPSIS (Technique for Preference by Similarity to the Ideal Solution), PROMETHEE (The Preference Ranking Organization METHod for Enrichment of Evaluations), COPRAS (Complex Proportional Assessment), VIKOR (VlseKriterijumskaOptimizacija I KompromisnoResenje), MOORA (The Multi-Objective Optimization Ratio Analysis), WASPAS (The Weighted Aggregates Some Product Assessment), MABAC (The Multi-Attributive Border Approximation area Comparison) and CoCoSo (Combined Compromise Solution).

The ELECTRE method was proposed by **Benayoun *et al.* (1965)** and was first applied to select the best option from a given set of options, but it was soon applied to three main problems: selecting, ranking and sorting. This method is famous for its cognate relationship to rank a set of options. MAUT method was developed by **Keeney and Raiffa (1976)**. This technique is very simple in solving multi-criteria decision making problems. TOPSIS method was first developed by **Hwang and Yoon (1981)** to select the best alternative from a set of alternatives. The underlying theory of the TOPSIS methodology is that the selected alternatives must have the minimum distance from the positive ideal solution (PIS) and the maximum distance from the negative ideal solution (NIS). PROMETHEE methods were first proposed by **Brans *et al.* (1984)** and have been widely employed. It is an efficient and outranking method. PROMETHEE I examine the positive and negative flows. PROMETHEE II determines the net flow as the final value and ranks the alternatives. In PROMETHEE III, the final ranking is done based on intervals.

The COPRAS method was developed by **Zavadskas *et al.* (1994)** which arises from the correlation of complicated relationships among elements of a decision matrix. It is a compensatory method. This method is used to estimate the maximum and minimum index values, and the effect of maximizing and minimizing indices of attributes on outcome evaluation is considered separately. The implementation of the COPRAS method needs less time and effort for calculation than other methods. A compromise method, VIKOR, proposed by **Opricovic (1998)**, is based on decrepit ideas of compromise programming and centered on choosing from a set of options with conflicting criteria. VIKOR ranks the options and determines a solution called compromise nearest to the ideal solution. In this method, the criteria are independent and qualitative criteria could be converted into quantitative criteria.

The MOORA method was proposed by **Brewers (2004)**, which is considered a non-subjective method. In addition, the desirable and undesirable criteria for ranking are used cumulatively to choose one of the preferable or supreme alternatives from different alternatives. In this procedure, the criteria are independent. The WASPAS method was introduced by **Zavadskas *et al.* (2012)**. This method combines the weighted sum model (WSM) and the weighted product model (WPM). In this, the relative preference of each criterion is determined, and then options are enumerated and prioritized. Also, the criteria are independent in this method. MABAC approach was introduced by **Pamučar and Ćirović (2015)**. In this approach, the distance of the alternatives defines from the border approximation area, and each alternative is ranked by specifying the difference between the distances. The Combined Compromise Solution (CoCoSo) method developed by **Yazdani *et al.* (2019)** is new and has a unique structure among several MCDM methods. This method is defined by a unified simple additive weighting and exponentially weighted product model. The weight of the alternatives is established in the decision-making procedure by three equations. At the final stage, an aggregated multiplication rule is applied to determine the ranking of the alternatives.

In addition to traditional methods, the Dempster-Shafer (D-S) theory, based on the Bayesian theory of subjective probability, is used to rank options in MCDM. It was proposed by **Dempster (1967)** and later extended by **Shafer (1976)**. It provides a mathematical foundation for modelling uncertainty and a robust approach for combining the degree of evidence collected from several sources. The criteria employed in the MCDM problem are considered sources of evidence for various hypotheses that are integrated to rank the alternatives.

1.5.2.3 Different methods for solving MODM problems

The MODM methods can be classified into mathematical programming models and heuristic algorithms based on computational time and solution. For example, mathematical methods include linear programming (LP), non-linear programming (NLP), mixed-integer linear programming (MILP), goal programming (GP), compromise programming (CP), and dynamic programming. Heuristic methods include ones such as simulated annealing (SA), genetic algorithm (GA), non-dominated sorting genetic algorithm (NSGA), and tabu search (TS).

Fuzzy sets and their extensions have been incorporated with the MCDM methods to deal with real-world problems' uncertain characteristics and the fuzziness of human minds. Therefore, applying fuzzy information to unfold the decision information can be reliable in many practical MCDM problems. Many fuzzy-based MCDM techniques have been developed, such as fuzzy TOPSIS, fuzzy VIKOR, fuzzy ELECTRE and fuzzy BWM, which have been used in many practical problems, such as situation assessments, weapon selection for defense systems, stock portfolio selection, and supplier selection under sustainability.

1.5.4 Linguistic variables

In the MCDM method, experts can express their opinions using crisp values. Imprecise information and subjective opinions of decision-makers, which often appear in the decision process of stock portfolio selection, crisp values are insufficient to solve such problems. A more realistic approach may be to use linguistic assessments instead of numerical values to model the decision-maker's decisions. A linguistic variable's values are words or sentences in a natural or artificial language. For instance, speed is a linguistic variable which can take the values “slow”, “fast”, “very fast”, and so on. The linguistic variable can be expressed by triangular fuzzy numbers (TFNs), trapezoidal fuzzy numbers (TrFNs), intuitionistic fuzzy numbers (IFN), hesitant fuzzy numbers (HFN), etc.

1.5.5 Preference relation

In MCDM problems, the decision maker feels comfortable expressing his choice by comparing each pair of given items and then making a preference relation. Roughly, there are two types of preference relations: fuzzy preference relation; and multiplicative preference relation. The key difference between these two types of preference relations is that the fuzzy preference relation conveys the decision maker's preference information about criteria or alternatives based on a 0.1–0.9 scale, which is asymmetric distribution around 0.5, whereas the multiplicative preference relation based on 1–9 scale of Saaty with non-symmetric distribution around 1.

Definition (Fuzzy preference relation): Let $A = \{A_1, A_2, .., A_n\}$ be a set of n alternatives, then $R = (r_{ij})_{n\times n}$ is called a fuzzy preference relation on $A \times A$, whose element r_{ij} estimates the preference of the alternative A_i over A_j and called fuzzy preference value. r_{ij} unfolds the maximum degree of preference for A_i over A_j. $r_{ij}\, \epsilon [0.5,1]$ implies definite preference for A_i over A_j. $r_{ij} = \frac{1}{2}$ indicates indifference between A_i and A_j.

Definition (Multiplicative preference relation): Let $A = \{A_1, A_2, .., A_n\}$ be a set of n alternatives, then $B = (b_{ij})_{n\times n}$ is called a multiplicative preference relation on $A \times A$, whose element b_{ij} estimates the preference of the alternative A_i over A_j, and is characterized by a ratio scale (such as Saaty’s ratio scale) such that $b_{ij} \in \left[\frac{1}{9}, 9\right]$, and $b_{ij} * b_{ji} = 1, i, j = 1,2, .., n$; here, $b_{ij} = 1$ implies indifference between A_j and

A_i; $b_{ij} > 1$ unfolds that A_i is preferred to A_j, especially, $b_{ij} = 9$ indicates that A_i is absolutely preferred to A_j; $b_{ij} < 1$ unfolds that A_j is preferred to A_i , especially, $b_{ij} = \frac{1}{9}$ reflects that A_j is absolutely preferred to A_i.

In today's real-world decision making problems, the grades of preference are not symmetric but distributed asymmetrically around some value. Therefore, Saaty's 1/9-9 scale is a useful tool to deal with such a situation given in Table 1.2. So far, several extensions of multiplicative preference relations have been proposed, such as intuitionistic multivariate preference relations, interval neutrosophic multiplicative preference relation, hesitant multiplicative preference relation.

Table 1.2 Comparison of 0.1-0.9 scale with the Saaty's Scale.

Saaty's Scale	0.1-0.9 Scale	Linguistic terms
1/9	0.1	Extremely not preferred
1/7	0.2	Very strongly not preferred
1/5	0.3	Strongly not preferred
1/3	0.4	Moderately not preferred
1	0.5	Equally preferred
3	0.6	Moderately preferred
5	0.7	Strongly preferred
7	0.8	Very strongly preferred
9	0.9	Extremely preferred
Values between 1/9 - 9	Values between 0.1 to 0.9	Intermediate values

1.5.6 Aggregation operator

The aggregation of information provided by decision makers is a fundamental need of an information processing system such as decision making. The purpose of aggregation operators is to convert a group of numbers into a unique representative value. Aggregation functions play a key role in MCDMs to slacken the dimensions of the criterion. According to the conventional aggregation operators, criteria are independent, and the efficacy of the criteria is additive (for instance, weighted averaging). Whereas, in real-world decision-making problems there are always different types of interrelationships between decision criteria, so this independent hypothesis cannot ordinarily be satisfied.

The most fundamental and plausible aggregation operators are the Choquet Integral (CI) & Sugeno Integral (SI), Power Average (PA), Bonferroni Mean (BM), and Heronian Mean (HM). Choquet and sugeno integral are two such fuzzy integrals that could be define on any type of fuzzy measure. CI is able to influence the importance of individual criteria and the importance of interrelationships among criteria. PA enables the criterion to reinforce each other between the two criteria, depending on the degree of support. The operators based on BM describes the interrelationships between different variables but does not explain the interrelationship between a decision criterion and itself and also does not differ the interrelationship between C_j and C_i from the interrelationship between criteria C_i and C_j. The HM operator has the peculiarity of catching the correlations of the aggregated arguments. Although the structure of the HM operator is similar to that of the BM, but it can effectively solve the two problems stated in the BM.

1.6 Portfolio optimization

In the portfolio construction process, a percentage of the total investment is obtained for each share using weighting schemes or optimization techniques. Weighting schemes are reckoned as an integral part of the portfolio selection process. The weightage assigned to some selected stocks in a portfolio selection process contributes to returns and is as important as stock selection and timing of investment decisions. Each return has an associated risk - as the weights vary, the risk and return also vary. The two main objectives of investors are- to maximize the return and minimize the risk. An optimization method produces a more appropriate portfolio concerning the portfolio's risk and return.

There are several weighting schemes to get asset weights, for instance, rank asset algorithm, equal ratio, Sharpe ratio, Sortino ratio, Treynor ratio, return-based, optimization techniques etc. The rank asset algorithm assigns weight to the shares according to their ranking. In equal ratio scheme, an equal value is distributed to each stock. Treynor, Sharpe and Sortino ratios measure an investment portfolio's risk-adjusted returns. The Treynor ratio evaluates the performance of a portfolio based on the systematic risk in its computation, while the Sharpe ratio measures total risk (symmetric and asymmetric). The Sortino ratio is the generalization of the Sharpe ratio. The Sortino ratio penalizes only negative volatility or downside deviation from the mean return as the upside deviation benefits the investor while the Sharpe ratio penalizes both upside and downside volatility equally. The high value of the Sortino ratio is considered a good investment. The Sortino ratio is used more to evaluate high-volatility portfolios, while the Sharpe ratio is applied to evaluate low-volatility portfolios. The return-based scheme establishes the weight of the asset based on its average return.

Many optimization techniques have been proposed, such as Markowitz optimization, genetic algorithm (GA), ant colony optimization (ACO), and particle swarm optimization (PSO) for portfolio optimization. The theory of modern portfolios was given by **Markowitz (1952)**, a mathematical framework for assembling a portfolio of assets that maximizes the expected return for a certain level of risk. GAs developed by **Holland (1975)**, are commonly used to generate high-quality solutions to optimization problems. ACO is a very popular meta-heuristic optimization technique first introduced by **Dorigo (1992)**, which can be used to find approximate solutions to difficult optimization problems and is inspired by the biological behavior of ants. ACO is particularly suitable for discrete optimization problems. **Kennedy and Eberhart (1995)** have invented a heuristic global optimization technique called PSO, based on the simulation of the social behavior of schooling birds, bees, or fish. Recently, deep learning approaches have also been used by researchers for portfolio optimization, ensuring robust and better results as compared to traditional approaches.

1.7 Objectives and plan of work

The objective of this research work is to develop such systems that yield high returns as well as provide the security of money for investment professionals and common investors. This study aims to understand the challenges of various investment strategies and propose comprehensive solutions that cover all stages of the investment process, including stock selection, portfolio optimization, and future investment analysis. Another goal of this research work is to build knowledge-based forecasting systems that ensure financial stability for future investments. To achieve our goals, this study focuses on designing hybrid models utilizing soft computing approaches (deep learning, fuzzy sets, and evolutionary algorithms) to handle the randomness and uncertainty of the stock market. Specific objectives of the present work are summarized as follows:

i. Developing comprehensive models that include all stages of the investing process (i.e., stock selection, portfolio construction, portfolio optimization, and future analysis).

ii. Designing multi-objective decision-making systems for both short-term trading and portfolio investment.

iii. Developing stock price forecasting models for stock portfolio selection in the fuzzy environment using deep architectures.

iv. Developing MCDM-based strategies for stock selection using fuzzy extensions that integrate multiple fundamental and technical financial indicators.

v. Construction and optimization of the portfolio using deep learning and evolutionary algorithms.

vi. Analyzing Indian economic sectors, investor sentiments, and developing investment models for the Indian stock market (NSE and BSE).

vii. Incorporating professional stock market expertise and neophyte investor concerns into decision making.

viii. Risk-return analysis and future investment evaluation of proposed models.

1.8 Structure of the thesis

This thesis is organized as follows.

Chapter 2 - Review of literature presents a contextual background of soft computing techniques used in financial decision making.

Chapter 3 - Materials and methods introduce the detailed methodology of different financial decision making models developed in this thesis. This chapter also presents the definitions and methods of other researches required for the development of the models.

Chapter 4 - Results and discussion present the simulation results of developed models when they were applied on the real data of Indian stock market (NSE and BSE). This chapter also include the comparative study that confirms the outperformance of the developed methods.

Chapter 5 - Conclusion summarizes the findings and gives the overall conclusions as well as the contribution and future scope of this research work.

Review of Literature

Chapter 2 REVIEW OF LITERATURE

Finance's status as a decision science only dates back to the 1950s with the publication of the paper on portfolio selection by **Markowitz (1952)**. This paper totally transformed the field of finance. In his work, he incorporated means, variance, and covariance into the newly emerging field of mathematical programming. Prior to 1952, there was essentially no theory of risk. Diversification was another area of frustration. Although people were aware about the dangers of putting "all of one's eggs in the same basket", there was no agreement about what exactly constituted diversification or ways to measure it. Moreover, the terms efficient portfolio and optimal portfolio had not become common terms yet.

Despite the massive success of the Markowitz model, it has endured many criticisms, one of which has possibly been the most persistent. It is because the fundamental model does not support additional criteria. While the criticism dates back at least to **Lee and Lerro (1973)**, the need for portfolio selection to include characteristics other than mean and variance has only grown, as attested to in this work. To facilitate the many new criterion ideas that have been introduced since 1973, it is now helpful to view this growing area of portfolio selection, which we will call multi-criteria portfolio selection, where "multiple" means more than two. After that, many studies have been done using various MCDM techniques for decision making in finance.

To serve the main objective of portfolio construction i.e. determination the optimal investment ratios for the stocks such that the expected returns is maximized under a bearable risk for a given period of investment is termed as portfolio optimization. For academics, evolutionary algorithms such as GA, ACO, PSO, and their hybrid approaches are highly popular heuristics. However, the current application of DL models has been observed to be quite promising for optimization proposals.

For financial time series forecasting many conventional statistical models have been developed using weighted moving average (WMA), moving average (MA), autoregressive integrated moving average (ARIMA), autoregressive moving average (ARMA) and exponential smoothing. But use of soft computing incorporating fuzzy set, neural network (ML and DL), evolutionary and nature based optimization techniques found very adequate and effective in handling big and heterogeneous stock market data.

This chapter presents literature related to applications of soft computing techniques for forecasting and decision making in stock market. Review of literature associated in this study is organized in three sections. Literature related to financial time series forecasting is presented in section 2.1. Section 2.2 discusses review of literature related to multi-criteria decision making techniques in stock market. Literature associated to portfolio optimization techniques is reviewed in section 2.3.

2.1 Financial time series forecasting

Atiya *et al.* (1997) developed a neural network forecasting model using fundamental information of companies such as earnings per share, price earnings ratio, dividends, sales, profit margin, and so on. These indicators and their ratios, particularly those relating to earnings, are considered the primary drivers of a stock's price. The simulation is performed on the Standard & Poor 500 (S&P 500) stocks using data from the 1993 calendar year, and the findings verified the practicality of the proposed model.

Corchado *et al.* (1998) presented an unsupervised neural network based technique using finite impulse response (FIR) to forecast financial time series. The proposed method is used to forecast the Dow Jones Index. When compared to multilayer perceptrons, radial basis function networks, and a regular ARIMA model, the results of the forecasts show that it has the lowest average error.

Chan *et al.* (2000) introduced the conjugate gradient learning algorithm to overcome the problem of slow convergence of the steepest descent in back propagation for time series forecasting. The technical analysis by neural network has been done on the daily trade data of the listed companies on the Shanghai Stock

Exchange. The results found that neural networks can model the time series satisfactorily, whatever the learning algorithm and weight initialization are adopted. However, the proposed conjugate gradient with multiple linear regression weight initialization has required a lower computation cost and learned better than the steepest descent with random initialization.

Walczak (2001) examined the effects of different sizes of training sample sets and suggested that those neural networks given an appropriate amount of historical knowledge can forecast future currency exchange rates with 60% accuracy while those neural networks which are trained on a larger training set have a worse forecasting performance.

Kodogiannis and Lolis (2002) presented improved neural network and fuzzy models used for exchange rate prediction. Several approaches, including multi-layer perceptions, radial basis functions, dynamic neural networks and neuro-fuzzy systems, had been discussed. A study has also been done using real daily exchange rate data of the US dollar vs. British pound to evaluate performance of all approaches for one-step and multiple-steps ahead predictions.

Wang *et al.* (2003) reinforced the significant bidirectional nonlinear causality between stock return and trading volume, and investigated whether trading volume can significantly improve neural network forecasting performance, or whether neural networks can adequately model such nonlinearity. The data of stock returns and trading volumes for the S&P 500 composite indices, as well as the Dow Jones Industry index, are used to train neural networks. Trading volume is used to train neural networks in order to see if it may aid in short-term predictions. According to empirical findings it concluded that trading volume leads to sporadic gains in predicting accuracy.

Zhora (2004) employed random subspace classifier for normalizing input data and to generate a sensitive classifier structure and variance structure selection. IBM stock data from 1998 to 2000 is used to test the proposed model. The best average success percentage in predicting stock price change direction is recorded at 58.1 percent.

Armano *et al.* (2005) presented a hybrid genetic-neural architecture for forecasting market indices. The activation of a feed forward artificial neural network for performing locally scoped forecasting has been controlled by a genetic classifier. Different types of information are fed to genetic and neural components: the former deals with inputs encoding information gleaned from technical analysis, and the latter processes other pertinent inputs, particularly past stock prices. The proposed method is tested on the COMIT index and the S&P 500 index, allowing realistic trading commissions to be considered. The results demonstrated the approach's superior predicting ability, as it consistently outperformed the buy and hold strategy.

Huarng and Yu (2006a) proposed a hybrid model using fuzzy time series and neural networks for forecasting known patterns, as well as a simple method for forecasting new patterns. They employed a back propagation neural network to solve the non-linearity problem. The Taiwanese Stock Index has been chosen as the forecasting goal for the years 1991-2003. The hybrid model outperformed both the basic and conventional fuzzy time series models, according to the empirical results.

Araújo *et al.* (2007) proposed a hybrid differential evolutionary system (HDES) for financial time series forecasting, which performs a differential evolutionary search for the minimum dimension to determine the characteristic phase space that generates the time series phenomenon. The ANN is combined with improved differential evolution (IDE) to create an intelligent hybrid model. IDE has searched for relevant time lags for proper time series characterization, as well as the number of processing units in the ANN hidden layer, the ANN training technique, and ANN modelling. It is observed that the proposed HDES model obtained better performance than a random walk model for the analyzed financial time series, overcoming the random walk dilemma.

Hussain *et al.* (2008) proposed a novel type of higher-order pipelined neural network: the polynomial pipelined neural network. Based on the engineering idea of divide and conquer, they built the network by concatenating a series of higher-order neural networks to forecast extremely nonlinear and non stationary inputs. The exchange rate between the US dollar and three other currencies is predicted using proposed neural network. Over a number of benchmarked neural networks, the network has shown better predictions and a higher signal-to-noise ratio.

Yu *et al.* (2009) discussed supervised neural networks as a meta-learning technique to design a financial time series forecasting system. They applied some data sampling approaches to produce distinct training subsets from the original datasets. Then, different neural networks and training procedures are used to create multiple prediction models or base models. The principal component analysis (PCA) technique is applied to enhance the accuracy of meta modelling predictions. The S&P 500 index, the New York Stock Exchange Index, US dollars vs. Euros, and US dollars vs. Japanese yen are all checked. The collected empirical findings show that the proposed strategy is effective.

Tarsauliya *et al.* (2010) investigated the utility of artificial neural networks for forecasting financial data series using various methods like as back propagation, radial basis function, and so on. Because of the x input variables, an x-y-1 network topology is chosen, with variable y defined by the number of hidden neurons during network selection with single output. Based on ANN architectures used to forecast daily S&P 500 and IBM indices, they stated that the number of hidden neurons initially drops and then raises the (root mean square error) RSME, while the number of input neurons first decreases and then increases the RSME.

Tarsauliya *et al.* (2011) ensembled of four neural networks: back propogation neural network (BPNN), radial basis neural network (RBNN), recurrent neural network (RNN), and elman neural network (EANN) to improve financial time series forecasting accuracy. To obtain the final output, the results of all four networks are combined using a gating integrator. The model has been used to forecast daily IBM and S&P prices, and it outperformed single neural networks.

Hamed *et al.* (2012) demonstrated an intelligent model for stock market signal prediction using multi-layer perceptron artificial neural network (MLP-ANN). The difficulties of prediction accuracy and lack of generalization are tackled by blind source separation technique from signal processing is incorporated with the learning phase of the generated baseline MLP-ANN. Because it converges quickly and allows generality in the learning mechanism, Kullback Leibler Divergence (KLD) is adopted as a learning method. The suggested model's accuracy and efficiency are confirmed using Microsoft stock from the Wall Street market, as well as several data sets from various sectors of the Egyptian stock market.

Aznarte *et al.* (2012) suggested fuzzy model evolved using a bio-inspired algorithm that produces accurate models for time series prediction. This model's performance is compared to that of a group of cutting-edge statistical models. To evaluate the proposal's merits, a detailed experimental research is carried out. Forecasting consistently of proposed model outperformed the other evaluated strategies in an experiment conducted on the Dow Jones Industrial Average Index.

Bhat and Kamath (2013) analyzed historical data to train the neural network that predicts movement for intraday nifty. The system has made prediction for every trading day with these methods to forecast if next day will be a positive day or negative. Buying and sell calls are also decided by the system, thus achieving full automation in stock trading.

João *et al.* (2014) investigated a possible ensemble organization, composed by neural networks (NNs) trained with the cascade correlation (CoNN) algorithm. Results obtained with data stream related with different stocks are analyzed and compared with those obtained with the traditional MLP-NNs, trained with back propagation. The proposed ensemble is proved an effective approach to non-stationary learning as it provided pre-defined rules that enable new learners with new knowledge to take part of the ensemble along data stream processing.

Adhikari and Agrawal (2014) developed a framework for asset price forecasting that included random walk (RW) and ANN models. They used the RW model to process the linear component of a financial dataset, and an ensemble of feed forward ANN (FANN) and EANN models to process the remaining nonlinear residuals. The suggested scheme's predicting ability has been tested using three prominent error statistics on four real-world financial time series. For all four financial time series, the obtained results clearly show that the proposed combined technique provided reasonably greater forecasting accuracies than each of the RW, FANN, and EANN models separately.

Ding *et al.* (2015) proposed a DL strategy for event-driven stock market prediction. First, using a novel neural tensor network, events are retrieved from news content and represented as dense vectors. Second, both short-term and long-term effects of events on stock price fluctuations are modeled using a deep convolution neural network. Experiments indicated that the model can enhance S&P 500 index prediction by almost 6% and individual stock prediction by nearly 6%, respectively.

Mustaffa *et al.* (2015) proposed a hybridization of the grey wolf optimizer (GWO) with the least squares support vector machines (LSSVM) algorithm for financial time series forecasting with the goal of better parameter tuning of LSSVM hyper parameters. The effectiveness of the suggested technique is confirmed using three statistical measures, namely mean absolute percentage error (MAPE), prediction accuracy, and root mean squared percentage error (RMSPE), on daily data of natural gas prices obtained from the barchart website.

Moghaddam *et al.* (2016) investigated the ability of ANN in forecasting the daily NASDAQ stock exchange rate. Several feed forward ANNs that are trained by the back propagation algorithm. The methodology considered the short-term historical stock prices as well as the day of week as inputs. Daily stock exchange rates from January 28, 2015 to June 18, 2015 are used to develop a robust model. Networks for NASDAQ index prediction for two type of input dataset (four prior days and nine prior days) are developed and validated.

Nayak *et al.* (2016) presented a virtual adaptive neuro-fuzzy inference system (VANFIS) for efficient forecasting of stock market indices. This model does not take any real-world data as input at any point in time, instead operating entirely in a virtual environment. 15 years of data from ten stock markets are used to validate the proposed model's performance, and five distinct performance indicators are analyzed. In compared to adaptive neuro-fuzzy inference system (ANFIS), simulation findings reveal that VANFIS greatly increases forecasting performance.

Rubio *et al.* (2017) introduced a novel weighting fuzzy time series technique for creating reliable ex-post forecasts and developed a fuzzy pattern operator. Under a fuzzy time series framework, a decision support system is developed for controlling the weights of the information provided by historical financial data. Several experiments are done to examine the historical performance of the time series, and it classified the properties of the series using a fuzzy operator, resulting in a trapezoidal fuzzy number as a one-step ahead forecast.

Korczak and Hemes (2017) discussed features of creating DL approaches for financial time series forecasting in reference to the A-Trader multi-agent stock trading system. The authors briefly examined the problem of financial time series on the FOREX market in the first half of the study. The advantages and disadvantages of traditional neural networks and DL models and their performance are discussed. This model reported better performance than MLP model. Conducted experiments have shown a significant drop in the error rate of time series forecasting with CNN compared to other models.

Bisht and Kumar (2018) proposed IFS-based fuzzy time series forecasting method using a simple computational algorithm to forecast using intuitionistic fuzzy logical relations. They forecasted the SBI share price, TAIEX, and the Dow Jones Industrial Average using the proposed method and discovered that it outperformed a number of existing methods.

Bas *et al.* (2018) introduced Pi-Sigma neural network for determining the fuzzy relationships for high order fuzzy time series forecasting. To train the Pi-Sigma network, they used a modified PSO algorithm. The model has been put to the test on two real datasets and compared to other methodologies, with the model coming out on top.

Panigrahi and Behera (2018) proposed a computationally efficient high order FTSF method. For expressing the FLR, they employed a generalized regression neural network (GRNN), which is a fast learning one-pass neural network. The model has been simulated fifty times on different time series independently, and the results are subjected to a thorough statistical examination. When compared to current alternatives, the results confirmed the suggested model's resilience and statistical superiority.

Selvamuthu *et al.* (2019) examined the outcomes of three different neural networks based on three different learning algorithms, namely Levenberg-Marquardt (LM), scaled conjugate gradient (SCG), and Bayesian regularization (BR), for stock market prediction based on tick data and 15-minutes data of an Indian firm. They discovered that using tick data, all three algorithms have a 99.9% accuracy rate. In comparison to results produced using tick data, the accuracy over a 15-minute dataset lowers to 96.2 percent, 97.0 percent, and 98.9 percent for LM, SCG, and BR, respectively.

Vlasenko *et al.* (2019) proposed a novel ensemble neuro-fuzzy model for financial time series forecasting to overcome limitations and improve the previously successfully applied a five-layer multidimensional Gaussian neuro-fuzzy model and its learning. The proposed solution avoided the error-prone hyper parameter selection process, resulting in higher accuracy in stock price predictions for Cisco, Alcoa, American Express, and Disney stocks.

Panigrahi and Behera (2020) tackled two major issues of FTSF: modelling FLRs and determining the effective length of intervals. They used three popular machine learning (ML) techniques: DBN, LSTM, and SVM for modelling FLRs. The proposed FTSF approaches outperform their crisp TSF equivalents statistically.

Niu *et al.* (2020) proposed a new forecasting framework based on a two-stage feature selection model: a DL model and an error correction model, with the goal of effectively capturing the nonlinearity inherent in multivariate financial time series. Case studies and associated sensitivity analyses are conducted on the Shangzheng composite index, Shangzheng fund index, and Dow Jones Industrial Average index to validate the performance of the forecasting framework, confirming its superiority over 16 benchmarks.

Mukherjee *et al.* (2021) employed deep feed-forward neural network (DFNN) and CNN to predict NSE stock prices. The DFNN model had a precision of 97.66 percent, while the CNN model had a precision of 98.92 percent, according to the study. The CNN model made predictions based on 2-D histograms created from a quantized dataset within a specific time frame. The model is tested on the recent COVID-19 epidemic data. The results found satisfactory, as it yielded a 91 percent accuracy rate.

Mehta *et al.* (2021) considered public sentiment, opinions, news and historical stock prices to forecast future stock prices. SVM, MNB classifier, linear regression, nave bayes, and LSTM among the machine-learning and deep-learning approaches are used in the research. The results show that the recommended methodology for applying BSE data is effective.

Syamala Rao *et al.* (2022) presented a financial time series forecasting model using multistage wavelet transform (WT). They processed the time series data through WT with different mother wavelet functions to extract high frequency and low frequency coefficients then used standard particle swarm optimization (PSO) algorithm to find optimal regression models in order to predict future samples.

Dawei *et al.* (2022) proposed a multi-modality graph neural network (MAGNN) to learn the multimodal inputs for financial time series prediction. The heterogeneous graph network is constructed by the sources as nodes and relations in the financial knowledge graph as edges. To ensure the model interpretability, a two-phase attention mechanism is utilized for joint optimization, allowing end-users to investigate the importance of inner-modality and inter-modality sources.

2.2 MCDM techniques in stock market

Lee and Lerro (1973) suggested more conditions beyond mean and variance for optimal portfolio selection. The goal programming model presented in this study allows various conditions to be simulated with relative ease and speed. The model is devised to recommend portfolios for a variety of conditions on entering the potential data once and doing simple changes in the priority levels of the certain goals.

Kumar *et al.* (1978) addressed the problem of goal conflicts in the portfolio selection of dual-purpose funds, and proposed an extension of standard methodology, in terms of the development of a goal programming model in conceptual form, which can be applied for the resolution of inherent clash of interests.

Lee and Chesser (1980) proposed a goal programming model to help investors in selecting efficient portfolio satisfying spectrum of investment desires. The mathematical formulation of the model include six investment goals that are capital budget, desired return from investment, establishment of accepted beta, maximum investment in a given security, investment preference for certain beta security and maximize the return of the portfolio. The model is applied for a hypothetical situation and the results indicate the achievement of objective of the study.

Alexander and Resnick (1985) expanded the model of immunizing a portfolio of default-free government bonds including default-grade corporate bonds. The immunizing equation is found to be slightly different. Both linear and goal programming are claimed the alternative techniques for identifying an investor's optimal immunizing portfolio.

Colson and Bruyn (1989) presented the first global computerized system for managing, in an integrated way, the three phases involved in all portfolio management: the information phase, the decision phase, and the control phase. They applied a single decision model (SDM) that used the Bayesian approach to process the information and to combine the forecasts supplied by internal analysts with the messages issued from external correspondents. The SDM output eight criteria for evaluating each security which is fed into simultaneous management model (SMM) and the decisional parameters are revised for the current portfolio based on a multiple criteria evaluation.

Khoury *et al.* (1993) suggested a multi-criterion approach of portfolio comparison and apply it to the selection of a global index fund portfolio using sixteen national stock market indices. The first algorithm used in the study (ELECTRE II) solves for the subset of portfolios among which the investor must limit his choice (the Kernel). The second algorithm (ELECTRE III) ranks all the portfolios considered from best to worst, on the basis of investor's preferences revealed through the various parameters used in the calculations.

Pardalos *et al.* (1994) provided an overview of various portfolio models, focusing on the corresponding optimization issues. They reported computational results for the standard Markowitz mean-variance model, using a dual algorithm for constrained optimization.

Konno and Suzuki (1995) proposed a mean-variance-skewness portfolio optimization model, a direct extension of the classical mean-variance model to the situation where the skewness of asset rate of return and the third order derivative of a utility function play significant roles in choosing an optimal portfolio, as well as three computational schemes for solving an associated non concave optimization model. The presented model is used to compute an estimated mean-variance-skewness efficient surface, which computed a portfolio with the highest predicted utility for any decreasingly risk-averse utility function.

Cooper *et al.* (1997) presented a portfolio planning model in which closeness to risk and return objectives has been measured in sums of absolute deviations. An appendix has shown how such use of dual variables may be applied to evaluate least absolute value (LAV) regressions relative to their sensitivity to data variations. Simple numerical examples are used to illustrate the potential uses of these dual variable values for evaluation in more complex situations, including determining whether an efficiency frontier is attained.

Jog *et al.* (1999) used trade-off information to create effective stock portfolio characterized by desired values of selected stock attributes. Basic notions behind such portfolio creation processes are discussed and also related the multi attribute analysis performed by evaluating compensations among the attributes' values. A framework to construct a portfolio using only compensatory information is presented and applied to the analysis of the stocks traded in the Toronto Stock Exchange.

Doumpos *et al.* (2000) briefly reviewed the main MCDA sorting techniques, and presented the multi group hierarchical discrimination method. The proposed sorting method is used to solve the portfolio selection problem. The proposed model is applied for stock selection among the 98 stocks traded in the Athens Stock Exchange. In a multiprocessing environment, this study provided good efficiency for solving large-scale problems.

Bouri *et al.* (2002) found PROMETHEE II method appropriate for evaluations according to each of the criteria for attractive portfolio selection. Five criteria are selected for stock analysis and applied the proposed methodology to the Tunisian Stock Market. The results are interpreted in perfect coherence with the Markowitz model.

Samaras *et al.* (2003) described multi-criteria decision support system presenting a complete and spherical evaluation of the stocks of Athens Stock Exchange. The system evaluates the stocks on the basis of three approaches: fundamental analysis, technical analysis and stock-exchange analysis utilizing multi-criteria analysis methods. The system is intended to support investment decisions while addressing both institutional and private investors with either a long-term or a short-term investment horizon.

Ehrgott *et al.* (2004) proposed a model for portfolio optimization based on multi attribute utility theory and the classical mean–variance model of Markowitz. The Markowitz model is extended formulating a hierarchy of objectives, which decomposes risk and return into five sub-objectives. Interpolation methods are used allowing individual investors preferences. Consequently, a non-convex mixed-integer programming model is formulated and solved by several Meta heuristics approaches. Numerical results showed that good solutions can be obtained for problem sizes relevant in practical applications in just a few seconds of CPU time.

Tiryaki and Ahlatcioglu (2005) suggested new method regarding fuzzy MCDM and applied it to the stocks selection on Istanbul Stock Exchange. The proposed approach used linguistic decision process considering the fuzzy relation on alternatives given by the decision makers. The model provided both ranking and weighting information to the investors and also found to be applied for other selection problems easily.

Huang *et al.* (2006) revised the conventional mean–variance method to determine the optimal portfolio selection under the uncertain situation. Possibilistic regression model and Mellin transformation are employed to obtain the mean and the risk by considering the uncertainty. Based on the results of the numerical example, the proposed method is found more flexible and accurate than the conventional method.

Albadvi *et al.* (2006) structured the stock selection model around two pillars: industry evaluation and company evaluation. PROMETHEE method has been used for solving the problem. The model is applied at Tehran Stock Exchange as a real case. The proposed model is found helpful for investor improving the structure of industry evaluation and company evaluation.

Gupta *et al.* (2008) presented fuzzy mathematical programming to develop comprehensive models of asset portfolio optimization for the investors' pursuing either of the aggressive or conservative strategies. An empirical study is done on a data set of daily closing prices in respect of 20 assets listed in NSE. The proposed model has provided satisfying portfolio selection strategies according to investors' vague aspiration levels, varying degrees of satisfaction, and varying importance of various objectives.

Cheung and Liao (2009) presented a decision framework based on the AHP to evaluate and select equity portfolios under a given investment horizon for Hong Kong stock market. The model offers a flexible and readily applicable addition to the financial practitioner's menu of equity selection techniques, and suggests fruitful extensions to optimize the use of knowledge under bounded rationality and to other areas of asset investment.

Ho *et al.* (2011) proposed a novel MCDM model, including DEMATEL, ANP, and VIKOR for exploring portfolio selection based on capital asset pricing model. This paper examined leading semiconductor companies spanning the hottest sectors of integrated circuit (IC) design, wafer foundry, and IC packaging. They determine that investing in a portfolio of wafer foundries is the best option.

Varma and Kumar (2012) identified and evaluated various criteria relevant for securities, which can aid in developing a framework for portfolio analysis. The companies listed in NSE are taken as the basis for the identification of relevant criteria to formulate a portfolio in an Indian context. DEMATEL method is applied to identify the importance and causal relationship among the identified criteria. The findings arrive expected to facilitate choice under multiple conflicting criteria scenario for portfolio selection.

Zopounidis and Doumpos (2013) analyzed the relevance of multi-criteria decision systems for financial decisions. A detailed discussion and up-to-date review on two important areas of financial decision support, namely portfolio selection and corporate performance evaluation, are given to highlight. The study also reported that the different multi-criteria modelling approaches can complement and enhance exiting techniques from the areas of finance and operations research.

Abdelaziz and Masmoudi (2014) modeled the portfolio's beta as a random variable and proposed a multiple objective stochastic portfolio selection model with random beta. A stochastic goal programming approach is applied to solve the model. A numerical example from the US stock exchange market is illustrated.

Aazam *et al.* (2015) applied ELECTRE III method for ranking the companies listed in Tehran Stock Exchange. Research results indicate ELECTRE III technique useful and efficient to select a portfolio. Moreover, value-based criteria as well as accounting criteria are found suitable to select a portfolio.

Yodmun and Witayakiattilerd (2016) presented a stock selection approach assisted by fuzzy procedures. Stocks are classified into groups according to business types. Within each group, the stocks are screened and then ranked according to their investment weight obtained from fuzzy quantitative analysis. Groups are also ranked according to their group weight obtained from fuzzy AHP and TOPSIS. As a demonstration, procedures are applied to a test set of data.

Mehlawat (2016) presented fuzzy multi-objective multi-period portfolio selection problems. A fuzzy credibility programming approach is used with multi-choice goal programming embedded in it. A real-world empirical application with data-set from an Indian stock market is presented to demonstrate usefulness of the proposed models and the solution approach for multi-period portfolio selection problem in fuzzy environment.

Masri (2017) developed a shariah-compliant optimization model for portfolio selection in an Islamic security market. The security return is considered stochastic and estimated based on the stochastic market return. The proposed model followed shariah principles by avoiding excessive risk and providing an ethical and socially responsible approach for portfolio selection. An empirical study from Bahrain Islamic Market is reported.

Si *et al.* (2018) proposed a multi-objective deep reinforcement learning model for intraday financial signal representation and trading that balanced profit and risk. They used deep neural networks to automatically extract dynamic market variables, and then used a reinforcement learning method to make continuous trading decisions using RNN. Experiments are carried out on three index-based future contracts traded in China, demonstrating it as a simple and reliable approach of profiting.

Thakur *et al.* (2018) used the fuzzy delphi method to identify the critical factors initially and applied Dempster–Shafer evidence theory to rank the stocks. A portfolio selection model that prefers stocks with higher rank is proposed. Illustration is done using stocks under BSE. Simulation is done by ACO. The performance of the outcome is found satisfactory when compared with recent performance of the assets.

Wang *et al.* (2018) introduced the Sharpe ratio (SR) in fuzzy and proposed fuzzy Value-At-Risk ratio (VR). On the basis of the two ratios (SR and VR), they built a multi-objective model to evaluate their joint impact on portfolio selection and used fuzzy simulation based multi-objective PSO algorithm to solve the model. The model effectiveness has been exemplified by three case studies using securities of New York Stock Exchange and Shanghai Stock Exchange for portfolio selection.

Li *et al.* (2019) constructed a new portfolio selection framework inspired by Maslow's need hierarchy theory using the bi-level optimization technique in which the lower-level need related to safety (low risk) while the upper-level need is concerned with self-actualization (high payoff). Empirical study is done on the American stock market and U. K. stock market which proved model applicability in determining optimal portfolios with moderate diversification.

Chen *et al.* (2020) proposed a multi-objective portfolio selection model using trapezoidal fuzzy numbers that includes a mean-semi variance model and a data envelopment analysis cross-efficiency model. They devised a cross-efficiency model based on the Sharpe ratio, as well as holdings bounds and cardinality constraints. A case study of 52 companies from Iran Stock Exchange is done to validate the suggested model.

Rahiminezhad *et al.* (2020) developed fuzzy ANP to assess and select portfolios on the Tehran Stock Exchange. The results indicate that profitability, growth, market, and risk are the most important criteria for portfolio selection.

Jiang and Wang (2021) investigated an uncertain multi-period multi-objective portfolio selection problem. They proposed that the multi-objective optimization issue be transformed using an ellipsoidal uncertainty set, and that a weighted-sum approach be used to find the problem's Pareto front. To illustrate the proposed strategy and validate the effectiveness and efficiency of the model established, numerical examples on ten distinct portfolios are presented.

Narang *et al.* (2021) presented an MCDM approach based on the combination of fuzzy set theory and COPRAS method to rank the alternatives in uncertain and ambiguous environment. A practical implementation of proposed method in stock selection is presented in the study. Portfolio is constructed by using the ranking obtained by proposed model. The weight allocation of the stocks is calculated using the ranked assets algorithm. Execution of the proposed model on NSE stocks is found to be dominant when compared with the previous studies.

Narang *et al.* (2022) developed two stage MCDM framework for stock portfolio selection. First, heronian mean operator is combined with the traditional CoCoSo method to present a new decision-making model for dealing with stock selection problems. Second, the base-criterion method is used to calculate the relative optimal weights of the specified decision criteria. A case study of stock selection for the portfolio under the National stock exchange (NSE) is discussed to validate the applicability of the proposed model. Different portfolios have been constructed using Particle swarm optimization (PSO).

Baydaş and Pamučar (2022) applied seven MCDM methods with different methodologies to evaluate companies' financial performance. Then, compared the obtained MCDM scores using two different objective verification mechanisms. The first validation criterion is the relationship of a MCDM method to real-life rankings (share price). The second criterion is the standard deviation (SD) technique used to discover the objective information content of MCDM final scores. According to the results of the study, PROMETHEE and FUCA (Faire Un Choix Adéquat) outperform other methods in terms of both SD values and strength of correlation with reference real-life rankings.

2.3 Portfolio optimization techniques

Shoaf and Foster (1998) were the first to use a GA to optimize a portfolio based on Markowitz model. In order to avoid infeasible solutions and penalty functions, the efficient set GA is adopted as an indirect representation method. Empirical studies proved the efficient set GA scales effectively for portfolios with up to $n = 100$ stocks with time complexity $O(n\,log\,n)$. This method is found more efficient than the classic quadratic programming method.

Chang *et al.* (2000) presented three heuristic algorithms based upon GAs, tabu search and simulated annealing for finding the cardinality constrained efficient frontier in order give optimal ratio to portfolio assets. The presented algorithms are applied on 225 assets of five different capital market indices (Hang Seng (Hong Kong), DAX 100 (Germany), FTSE 100 (UK), S&P 100 (USA) and Nikkei 225 (Japan)) and results are compared.

Karunamurthy (2003) demonstrated that a genetic programming approach to dynamic portfolio rebalancing under transaction costs and integral trade amounts is able to produce statistically significant beneficial trading strategies. These trading strategies are represented as paired S-expressions, with one S-expression devoted to rebalancing decisions and another devoted to individual trading decisions. They claimed that the computational complexity of achieving these results is not excessive in the 30-asset model, and it can be feasibly extended to 2000 asset or greater markets.

Oh *et al.* (2005) proposed a portfolio optimization scheme for index fund management using GA which is demonstrated for index fund designed to track Korea Stock Price Index 200 under various settings. GA process has outstanding advantages over the conventional portfolio mechanism. GA portfolio has reported the dominating performances in addition to more desirable properties during flat market.

Aranha and Hitoshi (2008) used a tree structure to represent a portfolio for the GA. They presented weights by intermediate nodes and assets by the leaves. They argued the tree approach for the preservation of building blocks, and accelerating the evolution of a good solution in portfolio optimization process. The initial experimental results supported presented new genome representation.

Cura (2009) presented used PSO for portfolio optimization problem using cardinality constrained mean-variance model. They used the weekly prices from March 1992 to September 1997 from Hang Seng (Hong Kong), DAX 100 (Germany), FTSE 100 (UK), S&P 100 (USA) and Nikkei (Japan) indices as test data. The results are compared with GA, simulated annealing and tabu search approaches. The results showed the successful applicability of proposed approach in portfolio optimization.

Anagnostopoulos and Mamanis (2010) used non-dominated sorting genetic algorithm II (NSGA-II), pareto envelope-based selection algorithm (PESA) and strength pareto evolutionary algorithm 2 (SPEA2), for solving the mixed-integer multi-objective portfolio optimization problem. SPEA2 is found to be the best algorithm for both the restricted and unconstrained optimization problems in the computational comparison, with PESA coming in second while being the fastest technique.

Chen *et al.* (2012) employed artificial bee colony algorithm for hybrid encoding of mixes integer and real variables to fulfill the characteristic of the portfolio optimization problem. The study is tested on four global stock market indices provided by the OR-Library and compared with simulated annealing, tabu search, and variable neighborhood search methods in the literature. ABC has performed better in terms of diversity, convergence, and effectiveness among all three algorithms; therefore, ABC has demonstrated its potential on portfolio optimization.

Sefiane and Benbouzian (2013) proposed a meta-heuristic ACO method to solve the portfolio optimization problem. They constructed a total cost function to optimize portfolio return while minimizing portfolio risk at the same time. A case study of five stock portfolios is presented, and the method is found to be capable and effective in determining the best portfolio return with the least amount of risk, as well as providing better solutions than other meta-heuristics based on GAs in terms of convergence time and efficiency.

Kamali (2014) employed the Markowitz mean–variance model for portfolio selection problem. They used two heuristic approach PSO and GA algorithm for portfolio optimization. The results claimed PSO approach more suitable for portfolio optimization.

Lei and Li (2015) proposed a Quantum-inspired Evolutionary Algorithm with Neighborhood Search (QEANS) to solve the project portfolios optimization problem with limited multiple resources and bounded risks for each project portfolio. They formulated the problem by a 0-1 linear programming model. Each problem solution is encoded by a Q-bits matrix, which has been updated by quantum-rotation gate. Randomly generated instances proved the effectiveness of the proposed QEANS.

Rezaei *et al.* (2016) investigate the application of invasive weed optimization (IWO) for portfolio optimization. The proposed model is compared with PSO and the reduced gradient method (RGM) using data envelopment analysis. They concluded that the IWO and PSO algorithms perform similarly in most significant criteria, although the IWO algorithm had a faster solving time and performs better in dominating inefficient solutions, whereas the PSO strategy performed better in total violation of restrictions.

Tzanetos *et al.* (2017) comprised ACO for detecting optimal combination of assets and a gravitational search algorithm (GSA), for optimal capital allocation in the portfolio. Experimental findings indicate that the proposed hybrid scheme yields a promising distribution of fitness values from independent simulation runs.

Zhao *et al.* (2018) proposed novel dynamic asymmetric copula model to capture the dependence structure across exchange-traded funds (ETFs) returns for obtaining weekly re-balanced portfolios. The proposed optimization is compared with the traditional mean–variance and the mean–CVaR portfolio optimization approach.

The proposed model is found providing significant improvements in the portfolio optimization process.

Costa *et al.* (2019) described a new adaptive meta-heuristic based on a vector evaluated approach for solving multi-objective problems. Tests have shown that the adaptive meta-heuristic reaches the best hyper-volumes in three of Z- directional tensile (ZDT) benchmarks functions and, also in two portfolios of a real-world problem called portfolio investment optimization. The proposed algorithm improved the Pareto curve when compared to the hyper volumes of each heuristic separately.

Aboussalah and Lee (2020) introduced the Stacked Deep Dynamic Recurrent Reinforcement Learning (SDDRRL) architecture to build a real-time optimal portfolio that addresses the difficulty of continuous action and multi-dimensional state spaces. SDDRRL is equipped with an automated Gaussian process (GP) with expected improvement for hyper parameter tuning. The algorithm is trained and tested on 10 equities from various sectors of the S&P 500. The suggested model outperformed three benchmarks: the rolling horizon mean-variance optimization (MVO) model, the rolling horizon risk parity model, and the uniform buy-and-hold (UBAH) index.

Zhang *et al.* (2020) used DL models to directly optimize the Sharpe ratio of the portfolio. They formed a portfolio by trading exchange-traded funds (ETFs) of market indices. Different asset class indices exhibited strong relationships, and trading them significantly narrows the scope of possible assets. They evaluated the model to a variety of algorithms, with the findings indicating that the suggested model performed the best over the testing period, which lasted from 2011 to the end of April 2020, including the financial instability of the first quarter of 2020.

Khan *et al.* 2021) formulated quantum beetle antennae search (QBAS), a meta-heuristic optimization algorithm, and a variant of beetle antennae search (BAS) for portfolio optimization. They applied QBAS on real-world stock market data and compared the results with other meta-heuristic optimization algorithms. The obtained results showed the outperformance of QBAS than swarm algorithms such as the PSO and the GA.

Ma *et al.* (2021) combined two machine learning models, i.e., random forest (RF) and support vector regression (SVR), and three DL models, i.e., LSTM neural network, deep multilayer perceptron (DMLP) and CNN for optimizing portfolio. Then incorporated their predictive results in advancing mean–variance and omega portfolio optimization models. Evaluation is done on historical data of 9 years from 2007 to 2015 of component stocks of China securities 100 index. MV and omega models with RF outperformed the other models.

Du (2022) presented a new mean variance portfolio optimization model using stationary portfolios composed of cointegrated stocks. SVM, RF, and attention-based LSTM network are used to predict expected returns. The proposed model is evaluated using data on stocks in the CSI 300 and the S&P 500, with 42 features over 8 years. The empirical results show that the portfolio constructed based on the stationary portfolios in both the Chinese and the US stock markets delivers significant profits.

Materials and Methods

Chapter 3 MATERIALS AND METHODS

This chapter has been classified into seven sections which contain the complete methodology and procedure of the following models:

Model [1]: Deep reinforcement learning based multi-objective systems for financial trading.

Model [2]: Stock portfolio selection based on fuzzy time series forecasting using fuzzy c-means clustering and deep learning techniques.

Model [3]: An effective hybrid MCDM approach of portfolio construction using modern portfolio theory.

Model [4]: A portfolio construction model based on sector analysis using Dempster-Shafer evidence theory and Granger causal network.

Model [5]: Stock portfolio selection hybridizing fuzzy base-criterion method and evidence theory in triangular fuzzy environment.

Model [6]: A method of intraday stock selection integrating fuzzy TOPSIS and belief divergence measure in evidence theory.

Model [7]: Single valued triangular neutrosophic MEREC-CoCoSo method for multi-criteria decision making and its application in portfolio construction.

3.1 Model [1]: Deep reinforcement learning based multi-objective systems for financial trading

The primary objective of live trading systems (i.e., intraday trading systems) is to maximize overall return; however, the return is influenced by risk. Accepting a higher risk can increase total return, but it can also result in a significant loss, whereas accepting a lesser risk means compromising total return. Risk plays a crucial role in the trading environment, so there is need to develop trading systems that make the maximum possible profit while restraining the risk. Therefore, making the most precise decision is important under dynamic market conditions. Reinforcement learning (RL) algorithms have attracted researchers to train trading systems as RL agent explore unknown environmental states and choose the most favorable action by past experiences. The learning agent takes input as different price patterns and outputs discrete trading action signals that optimize the profit and control the risk. In order to achieve the goal of varied kinds of learning problems, mainly two sorts of RL algorithms are getting used: critic-based algorithms and actor-based algorithms.

Critic based algorithms are the value-function based methods, e.g., Temporal Difference-learning (TD- learning) **(Sutton and Barto, 1998)**, Quality-learning (Q-learning) **(Tesauro, 1994)** and State-Action-Reward-State-Action (SARSA) **(Singh *et al.*, 2000, Corazza and Sangalli, 2015).** These are dynamic programming methods used for solving the optimization problem in discrete space. Many researchers often use Q-learning for financial trading. However, critic-based methods are not ideal for online trading problems because of the complexity of trading problems and the online manner of trading **(Gao *et al.*, 2000)**. Such situations are difficult to handle in discrete space.

Recently, researchers work with another algorithm, i.e., the actor-based method **(Deng *et al.*, 2017)**, which directly learns to make policy and take continuous actions. There is no need to describe discrete market conditions in actor-based methods, which makes optimization easier. **Moody and Saffell (2001)** introduced an actor-based learning algorithm and proposed an RL model for optimizing portfolios and risk control. Since then, many researchers have used the RL model with different objectives **(Deng *et al.*, 2017, Si *et al.*, 2018)**.

Training the RL agent over large datasets helps the agent inspect broader market situations. The hierarchical functioning of DL enables the agent to process the vast amount of unstructured datasets **(Goodfellow *et al.*, 2016)**. The blend of DL and RL, i.e., deep reinforcement learning (DRL) has blown the mind of researchers with its remarkable success in Alpha Go **(Silver *et al.*, 2016)**, playing Atari games **(Mnih *et al.*, 2013)**, natural language processing **(Sharma and Kaushik, 2017)**, healthcare **(Liu *et al.*, 2019)** and many more. Using DRL for training the trading systems can improve their performance to make better decisions during trading.

The problem of dealing with large datasets is overcome with DL, but the other challenges are the noise and uncertainty present in the data received online. Deep feature learning techniques are promising for discovering intricate structures and extracting relevant features. However, raw features could adversely affect the trading system's performance and give absurd results. Initially, a sparse coding model was introduced for feature representation **(Jianchao Yang *et al.*, 2010)**, but it was not considered for big data. Deep feature learning not only extracts features but also improves accuracy, reduces overfitting, speeds up training, and improves data visualization, as we can see in the field of image classification **(Ma *et al.*, 2016)** and speech recognition **(Graves *et al.*, 2013)**. **Deng *et al.* (2017)** proposed a DL framework for robust feature learning and combined it with the recurrent reinforcement learning (RRL) model for real-time financial signal processing. Therefore, deep feature learning ensures robust key feature summarization and extraction directly from the data, and then DRL ensures the robustness of the decision-making process.

There are different types of stocks in the financial market with different leverages, market patterns, and investment policies which require different parameters to analyze them. Therefore, in the context of dealing with different stocks and achieving the investor's ultimate goal, in this study, we proposed two multi-objective systems based on DRL for stock trading in the live market. The target of the RL agent is to give {buy, hold and sell} signals in order to optimize the total reward term as multi-objective functions. The whole structure of systems is designed combining two deep neural networks. The first is the LSTM autoencoder for robust feature extraction, and the second is DRL with LSTM recurrent neural network for decision making.

3.1.1 Preliminaries

This section comprises brief introductions about deep reinforcement learning and long short-term memory.

3.1.1.1 Deep Reinforcement Learning (DRL)

DRL is a combination of RL and DL. RL is a process in which an agent learns to make decisions through trial and error. This problem is often modelled as Markov decision process (MDP), where at each time step, an agent is in a state s, take actions a, receives a scalar reward r and transitions to the next state s' according to environment dynamics $\rho(s'|s,a)$. The agent attempt to learn the policy $\pi(a|s)$ or map the observations to actions, in order to maximize its return (expected sum of rewards). In RL the algorithm only has access to the dynamics $\rho(s'|s,a)$ through sampling.

In many practical decision making problem the state (s) of the MDP are high dimensional and cannot be solved by traditional RL algorithms. DRL algorithms incorporate DL to solve such MDPs, often representing the policy $\pi(a|s)$ or other learned functions as a neural network, and developing specialized algorithms that perform well in this setting. DRL is learning from existing knowledge and applying it to a new data set.

3.1.1.2 Long short-term memory (LSTM)

LSTM is a special kind of recurrent neural network capable of learning long time dependencies. It is designed to remember information for long period of time. An LSTM cell is composed of a memory unit c, a hidden state h and three types of gates - input gate i, output gate o and forget gate f. These gates regulate the flow of information through the cell. At each time-step t input x_t and previous hidden state h_{t-1} activates all three gates. Forget gate removes less important information from cell, input gate feed some additional information to the cell and output gate selected some important information and show it. Then, memory cell c_t and hidden state h_t get updated. The involved computations are given as follows:

$$i_t = \sigma(w_{xi}x_t + w_{hi}h_{t-1} + b_i)$$

$$f_t = \sigma\left(w_{xf}x_t + w_{hf}h_{t-1} + b_f\right)$$

$$o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + b_o)$$

$$c_t = f_t * c_{t-1} + i_t * \tanh(w_{xc}x_t + w_{hc}h_{t-1} + b_c)$$

$$h_t = o_t * \tanh c_t$$

here, $\sigma(x) = \frac{1}{(1+\exp(-x))}$ is logistic sigmoid function and w and b are the weights and biases respectively, h_t is the final output of LSTM cell. Figure 3.1.1 displays the LSTM cell.

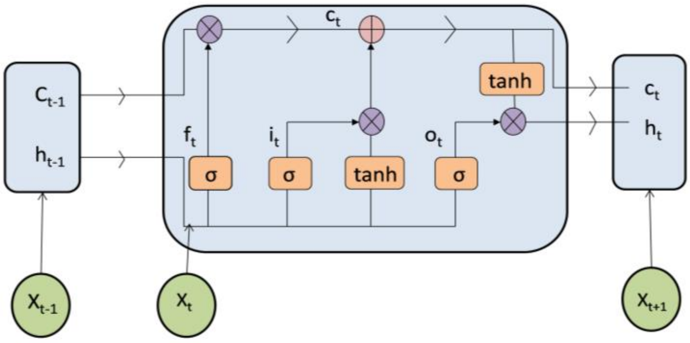


Figure 3.1.1: Long short-term memory (LSTM) cell

3.1.2 Proposed model

This model is developed in two sections. Section 3.1.2.1 presents proposed multi-objective structures and section 3.1.2.2 details the deep neural network setup for implementing proposed structures.

3.1.2.1 Multi-objective structures

Considering the basic framework of DRL proposed by **Moody and Saffell (2001),** which is essentially a single layer recurrent neural network. The historical price data of an asset can be taken as close price sequence $z_1, z_2, \ldots z_{t\ldots}$ and return at time t as $p_t = z_t - z_{t-1}$. The trading system has to train to make a decision (policy) $\theta_t = \{-1,0,1\}$ i.e. $\{long, neural, short\}$ or $\{sell, hold, buy\}$ position at the end of each time interval$(t-1, t)$. The decision (policy) making function θ_t is defined as

$$\theta_t = \tanh(\langle w, f_t \rangle + b + v * \theta_{t-1}) \tag{3.1.1}$$

In equation (3.1.1), f_t represents feature vector containing recent k return values at time t i.e. $f_t = [z_{t-k+1}, \ldots z_t] \in R_k$. $W = \{w, b, v\}$ is the set of learning parameters of DRL trading system. The profit P_t made by trading system at time t is defined by

$$P_t = \theta_{t-1} r_t - c|\theta_t - \theta_{t-1}| \tag{3.1.2}$$

In equation (3.1.2), the first term denotes the profit or loss obtained in the trading position θ_{t-1} at time $t-1$ and c denotes the transaction cost which has to be paid on changing the trading position at time t. Transaction cost also discourages frequently changing trading positions. Some researchers used risk-adjustment ratios such as Sharpe ratio and Sterling ratio to consider risk factors in maximizing total profit **(Deng *et al.* (2017), Moody and Saffell (2001))**. Sharpe ratio is the ratio of average return to standard deviation over the period $1, \ldots, T$.

$$ST = \frac{mean\ (P_1, \ldots, P_T)}{std\ (P_1, \ldots, P_T)} \tag{3.1.3}$$

Moving averages of returns are considered to get the value of the Sharpe ratio at each timestamp. The basic DRL objective is to optimize the moving Sharpe ratio that can accumulate maximum total profit over the period T defined as

$$U = \max \sum_{t=0}^{T} P_t \tag{3.1.4}$$

Our goal is to maximize profit while taking minimum risk throughout the trading period T. So, the first multi-objective structure is described as follows:

$$U_1 = \max \sum_{t=1}^{T} [\alpha\{mean\ (P_{t+k}, \ldots, P_t)\} - \beta\{std(P_{t+k}, \ldots, P_t)\}] \tag{3.1.5}$$

It is a risk-adjustable return function. The first term under the summation measures the average of recent k returns, and the second term measures the volatility of recent k returns. In order to maximize the function in equation (3.1.5), the second term has to minimize i.e. minimum risk is taken during the trading. α and β are adjustable parameters, here we take $\alpha = 1$ and $\beta = 0.01$. The DRL trading system tends to maximize the function by refining the learning parameters $= \{w, b, v\}$.

In the function in equation (3.1.5), the total volatility of the investment is considered, but some investments are more concerned with downside risk. So considering downside or negative volatility, here we describe another multi-objective structure as follows:

$$U_2 = \max \sum_{t=1}^{T} [\alpha\{mean\ (P_{t+k}, \ldots, P_t)\} - \beta\{std(N_{t+k}, \ldots, N_t)\}] \tag{3.1.6}$$

$$N_i = \min(P_i, 0) \quad i = t, \ldots., t+k$$

Other symbols in this function are taken same as in the equation (3.1.5).

3.1.2.2 Deep neural network for multi-objective trading systems

The whole neural network comprises two fully connected deep neural networks, first for robust feature learning and second for reinforcement learning. LSTM architecture is used in both networks because of its significance in handling sequential data. This section explains the deep feature learning (LSTM autoencoder) and deep reinforcement learning (LSTM recurrent neural network).

A. Deep feature learning (LSTM autoencoder)

LSTM autoencoder is a sequential data autoencoder system based on the LSTM architecture. We feed the raw dataset in sequences to LSTM autoencoder, it is configured to read the input data, encode it, decode it, compress it and extract robust features. It contains three main layers, input layers (l^{th} layer) to input raw features i.e. f_t , encoding LSTM layer ($(l+1)^{th}$ layer) to encode input features into less dimensional feature vector (say h_t^{enc}) i.e. $h_t^{enc} = encoder(f_t)$ then decoding LSTM layer ($(l+2)^{th}$ layer) read h_t^{enc} and predicts a feature vector h_t^{dec} i.e. $h_t^{dec} = decoder(h_t^{enc})$. LSTM autoencoder try to minimize the difference between f_t and h_t^{dec} :

$$loss = \min\left\|h_t^{dec} - f_t\right\|^2$$

Computation in encoder part:

$$i_t^{enc} = \sigma(w_{Fi}x_t + w_{hi}h_{t-1}^{enc} + b_i)$$

$$f_t^{enc} = \sigma\left(w_{Ff}x_t + w_{hf}h_{t-1}^{enc} + b_f\right)$$

$$o_t^{enc} = \sigma(w_{Fo}x_t + w_{ho}h_{t-1}^{enc} + b_o)$$

$$c_t^{enc} = f_t^{enc} * c_{t-1}^{enc} + i_t^{enc} * \tanh(w_{Fc}x_t + w_{hc}h_{t-1}^{enc} + b_c)$$

$$h_t^{enc} = o_t^{enc} * \tanh(c_t^{enc})$$

Computation in decoder part:

$$i_t^{dec} = \sigma\left(w_{Hi}h_t^{enc} + w_{hi}h_{t-1}^{dec} + b_i\right)$$

$$f_t^{dec} = \sigma\left(w_{Hf}h_t^{enc} + w_{hf}h_{t-1}^{dec} + b_f\right)$$

$$o_t^{dec} = \sigma\left(w_{Ho}x_t + w_{ho}h_{t-1}^{dec} + b_o\right)$$

$$c_t^{dec} = f_t^{dec} * c_{t-1}^{dec} + i_t^{dec} * \tanh\left(w_{Hc}h_t^{enc} + w_{hc}h_{t-1}^{dec} + b_c\right)$$

$$h_t^{dec} = o_t^{dec} * \tanh\left(c_t^{dec}\right)$$

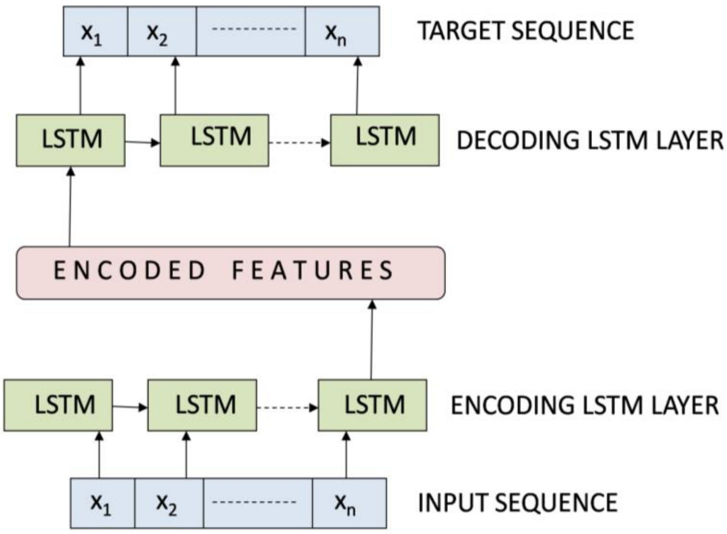


Figure 3.1.2: Overview of LSTM autoencoder

B. Deep reinforcement learning (LSTM recurrent neural network)

After learning robust features, we implemented LSTM recurrent neural network for decision making. Here, the LSTM network memories the previous trading decisions while taking successive decisions. It also helps in reducing transaction cost. The encoded features h_t^{dec} are taken as input in the LSTM, the whole procedure is govern by the following equations:

$$i_t^{rec} = \sigma(w_{Hi}h_t^{enc} + w_{hi}h_{t-1}^{rec} + b_i)$$

$$f_t^{rec} = \sigma\left(w_{Hf}h_t^{enc} + w_{hf}h_{t-1}^{rec} + b_f\right)$$

$$o_t^{rec} = \sigma(w_{Ho}h_t^{enc} + w_{ho}h_{t-1}^{rec} + b_o)$$

$$c_t^{rec} = f_t^{rec} * c_{t-1}^{rec} + i_t^{rec} * \tanh(w_{Hc}h_t^{enc} + w_{hc}h_{t-1}^{rec} + b_c)$$

$$h_t^{rec} = o_t^{rec} * \tanh(c_t^{rec})$$

The final output h_t^{rec} used to make trading decision (policy) θ_t at time t:

$$\theta_t = \tanh(\langle w, h_t^{rec}\rangle + b + v * \theta_{t-1})$$

w, b and v are the learning parameters. At $t = 0$, $\theta_0 = 0$ i.e. initial trading decision is to be in holding position.

The values of θ_t are ranged between -1 to 1, so here, parameter λ is defined to distinguish three decisions $(-1, 0, 1)$. We take $\lambda = 0.3$ and consider final decisions according to the function given in equation (3.1.7).

$$\theta_t^* = \begin{cases} -1, & \theta_t \leq -\lambda \\ 0, & -\lambda < \theta_t < \lambda \\ 1, & \theta_t \geq \lambda \end{cases} \tag{3.1.7}$$

The obtained discrete value of θ_t^* is further processed in the network to calculated U_1 and U_2 as given in equation (3.1.4) and equation (3.1.4), respectively.

3.2 Model [2]: Stock portfolio selection based on fuzzy time series forecasting using fuzzy c-means clustering and deep learning techniques

Forecasting of financial time series (majorly stock prices forecasting) is very helpful in selecting appropriate stocks for investing. Forecasting methods with better accuracy are always in high demand. Both stochastic and non-stochastic methods have been employed by researchers for financial time series forecasting in past years. Stochastic methods like moving average (MA), autoregressive integrated moving average (ARIMA), vector regression (VR) and exponential moving average (EMA) based models have limitations in handling complex and highly uncertain real-world forecasting problems. Due to these limitations, non-stochastic methods are preferred over stochastic methods. Fuzzy time series forecasting (FTSF) models are highly prevalent in the research field because of linguistic representation; these methods more closely illustrate real-world scenarios and generally give better results than traditional methods. **Song and Chissom (1993, 1994)** were the first to introduce time series forecasting models based on fuzzy sets. The whole process of FTSF has four major steps as follows:

1. Determining universe of discourse (UOD), number of intervals (NOIs) and length of intervals to divide UOD.
2. Obtaining fuzzy sets and fuzzifying the time series.
3. Establishing fuzzy logic relationships (FLRs) on fuzzifed time series.
4. Defuzzifying fuzzifed forecasted value to get crisp output.

Real time series are usually non-uniformly distributed, which encouraged researchers to develop methods that partition UOD into unequal intervals. Many researchers suggested different algorithms (such as ratio-based **(Huarng and Yu, 2006b)**, genetic algorithm based **(Chen and Chung, 2006)**, probability distribution methods **(Bisht and Kumar, 2016, Gupta and Kumar, 2019)**, clustering methods **(Cheng *et al.*, 2008, Egrioglu *et al.*, 2011, Li *et al.*, 2008)** to partition UOD into unequal intervals.

For the second step of the forecasting algorithm, most researchers develop models by using linguistic values to fuzzify the crisp time series according to the index of intervals obtained by UOD partitioning **(Cheng *et al.*, 2006, Rubio *et al.*, 2016, 2017)**. After fuzzification of crisp time series next step is to establish fuzzy logic relationships (FLRs). Distinct methods such as fuzzy relation matrix **(Wong *et al.*, 2010)**, fuzzy logic relation groups (FLRGs) **(Cheng *et al.*, 2016, Singh and Borah, 2013a, Ye *et al.*, 2016)**, artificial neural network (ANN) **(Bas *et al.*, 2018, Gu *et al.*, 2017, Singh and Borah, 2013b)** and recently machine learning techniques have been employed by the researchers for formulating the rules of fuzzy logic relation **(Panigrahi and Behera, 2018, 2020)**. Among theses, artificial neural network are computationally convenient and also state-of-the-art methods. Recently, **Panigrahi and Behera (2020)** implemented machine learning techniques for establishing FLRs. They suggested a modified average based method to find the length of intervals and used some popular machine learning techniques such as DBN, LSTM and SVM for modelling FLRs. **Pattanayak *et al.* (2020)** developed a new method of FTSF. They used membership values with the data and used a support vector machine to establish FLRs.

The findings of **Pattanayak *et al.* (2020)** inspired us to develop this model using fuzzy clustering and deep learning techniques to achieve better accuracy in forecasting financial time series. Considering the first two steps of FTSF, the proposed model uses interval index number and membership value as input features to predict future value. We suggested a rounding-of range and large step-size method to find the optimal NOIs and used fuzzy c-means clustering process to divide UOD into intervals of unequal length. Two deep learning techniques SVM and MLP are implemented to establish FLRs. The forecasted financial time series are obtained by defuzzification. This method is employed to forecast the stock prices of different stocks. Then, the stocks are ranked based on the return and risk ratio estimated by forecasted stock prices. A portfolio is constructed with top-ranked stock, and portfolio optimization is done using ant colony optimization (ACO).

3.2.1 Preliminaries

This section provides short notes on the fuzzy c-means clustering, support vector machine and multilayer perceptron.

3.2.1.1 Fuzzy c-Means Clustering (FCM)

Fuzzy c-means (FCM) clustering algorithm was developed by **Bezdek (1981)**. Suppose $X = \{x_1, x_2, \dots x_n\}$ is the set of data points which have to partition into $2 \le c \le n$ clusters. In FCM process, dataset is partitioned into c clusters, the degree of belongingness (u_{ij}) of each data point (x_i) to each cluster $(j = 1,2, \dots c)$ and center of the clusters (v_j) are calculated by equation (3.2.1) and (3.2.2) respectively:

$$u_{ij} = \frac{\left(1/d\left(x_i, v_j\right)^2\right)^{\frac{1}{p-1}}}{\sum_{k=1}^{c}(1/d(x_i, v_k)^2)^{\frac{1}{p-1}}} \quad (3.2.1)$$

$$v_j = \frac{\sum_{i=1}^{n} u_{ij}{}^{p} * x_i}{\sum_{i=1}^{n} u_{ij}{}^{p}} \quad (3.2.2)$$

here, $d\left(x_i, v_j\right)$ denotes euclidean distance between data point x_i and cluster center v_j and $p \in [1, \infty[$ parameter of fuzziness.

$u_{ij}'s$ and $v_j's$ are calculated iteratively in order to minimize sum squared error (SSE)

$$SSE = \sum_{i=1}^{n} \sum_{j=1}^{c} u_{ij}^{p} d^2\left(x_i, v_j\right) \quad (3.2.3)$$

subject to the following constraints:

$u_{ij} \in [0,1]$; $\sum_{j=1}^{c} u_{ij} = 1$; $\sum_{i=1}^{n} u_{ij} < n$

Minimization of SSE ensures the each data point belongs to cluster whose center is nearest to it and it also makes well separated clusters.

3.2.1.2 Support Vector Machine (SVM)

The use support vector machine for regression task was first documented by **Vapnik (1995)**. This regression technique is known as support vector regression (SVR). Let $X = \{(X_1, y_1), (X_2, y_2), \dots (X_n, y_n)\}$; $X_i \in \mathcal{R}^m, y_i \in \mathcal{R}$ be the training set. In SVR, we try to find the best fit function say $f(X)$which predict y for given X with following two assumptions: (i) the error between predicted $y_i's$ and actual values must be less than ϵ, and (ii) $f(X)$ must be as flat as possible. The function $f(X)$ is defined as follows:

$$f(X) = W.\varphi(X) + B \tag{3.2.4}$$

here, W is weight vector, B is bias and $\varphi(X)$ is a non-linear function. For linear regression we take X in place of $\varphi(X)$ in equation (3.2.4).

To ensure the flatness of the function in equation (3.2.4), we consider an optimization problem given below:

$$\text{minimize } \frac{1}{2}W^2 \tag{3.2.5}$$

subject to the constraints: $\begin{cases} y_i - W.\varphi(X_i) - B \leq \epsilon \\ W.\varphi(X_i) + B - y_i \leq \epsilon \end{cases}$

Usually it may difficult to find such regression function that completely fits the training set, may be some training data points fall outside the ϵ margin. In this case, we can reconsider optimization problem defined in equation (3.2.5) which find the best fit function $f(X)$ allowing some errors outside the ϵ margin. Two slack variables ξ_i and ξ_i^* are introduced and the optimization problem became:

$$\text{minimize } \frac{1}{2}w^2 + C\sum_{i=1}^{l}(\xi_i + \xi_i^*) \tag{3.2.6}$$

subject to the constraints: $\begin{cases} y_i - W.\varphi(X_i) - B \leq \epsilon + \xi_i \\ W.\varphi(X_i) + B - y_i \leq \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$

The parameter $C > 0$ maintains the flatness of the function $f(X)$ and the allowed ϵ error and also prevents over fitting. Lagrange dual form of the above optimization problem is easier to compute which is defined as follows:

$$\text{maximize } \left(-\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)K(x_i, x_j) - \epsilon\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) + \sum_{i=1}^{n} y_i(\alpha_i - \alpha_i^*)\right) \tag{3.2.7}$$

subject to the constraints: $\begin{cases} \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0 \\ 0 \leq \alpha_i, \alpha_i^* \leq C \end{cases}$

here, α_i and α_i^* are non-negative Lagrange multipliers and $K(x_i, x_j)$ is kernel function. We can define different types of kernel such as linear, polynomial as per the regression problem.

3.2.1.3 Multilayer perceptron neural network (MLP)

Multilayer perceptron neural networks are the feed forward neural networks basically consisting of three or more sequential layers: input layer, one or more hidden layers and output layer. The nodes in the input layer receive the data which processed and forwarded to the hidden layers. Hidden layers further process the received information and send it to the output layer. Layers are fully connected, different weights are assigned to every connection and activation functions are employed to hidden layers and output layer which activate the nodes. There can be different number of hidden layers required for different datasets according to the complexity of the task.

Input signal x_i which is impinging on the j^{th} node of hidden layer is multiplied to the connection weight and summed up. The sum passes through the activation function to activate the node. Similar procedure is followed to process the data from all hidden layers to the final output layer. The output of j^{th} node is defined as follows:

$$y_j = f\left(\sum w_{ij} x_i\right) \tag{3.2.8}$$

here, f is an activation function and w_{ij} is the weight of the connection of i^{th} node of preceding layer. The neural network is trained to minimize the sum squared error between the actual output and obtained output.

3.2.2 Proposed model

Model [2] is developed in three sections, and its overview is shown in Figure 3.2.1. In section 3.2.2.1, a FTSF method is developed for forecasting prices of different stocks. Section 3.2.2.2 discusses portfolio construction using forecasted results. Simulation of ACO for portfolio optimization is described in section 3.2.2.3.

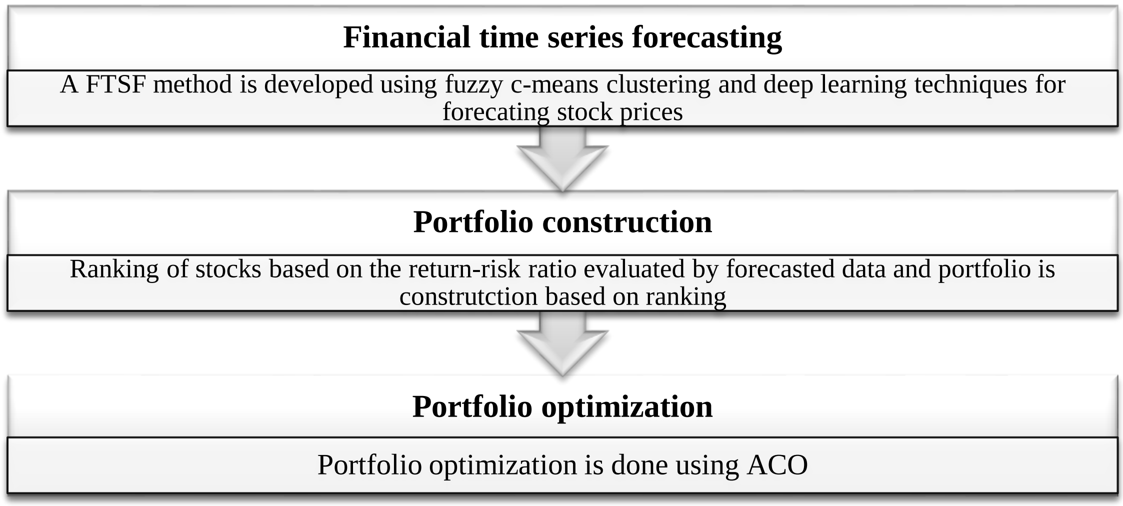


Figure 3.2.1: Overview of Model [2].

3.2.2.1 FTSF using fuzzy c-means clustering and deep learning techniques

This section presents developed FTSF method which includes following steps:

Step 1. Defining universe of discourse (UOD) for time series data:

Define universe of discourse (UOD) as $U = [y_{min} - d, y_{max} + d]$, here y_{min} and y_{max} are the maximum and minimum values of the real time series and d is a positive integer.

Step 2. Determining NOIs, clustering the time series data by FCM and generating cluster centers:

To apply FCM, first the optimal number of clusters (c) to partition the data into groups must be specified. For this, first we take round-off range of the time series and a large step-size according to the variation of the data. Then divide the range into equal intervals. And finally, the no. of clusters (c) is considered equal to the no. of intervals.

For example, time series data varies from y_{min} to y_{max}. The round-off range taken as the nearest values of y_{min} and y_{max} in multiple of 10, suppose $[y^*_{min}, y^*_{max}]$ is the round-off range. Taking a as step size, partition range into n equal intervals, i.e., $[y^*_{min}, y^*_{min} + a]$, $[y^*_{min} + a, y^*_{min} + 2a]$, $[y^*_{min} + 2a, y^*_{min} + 3a]$, … $[y^*_{min} + (n-1)a, y^*_{min} + na = y^*_{max}]$. Hence, take $c = n$ and divide the time series data into n clusters by FCM. Taking large step-size means less number of clusters which avoid over lapping in time series data and cluster them into well separated groups.

Step 3. Dividing UOD in unequal intervals:

For dividing UOD in unequal intervals midpoints of cluster centers are taken as boundaries of intervals. This method of determining intervals helps to find the appropriate NOIs according to the range and variation of the time series. Refining the size of intervals with the help of cluster centers ensure the density of the data in the particular interval.

Suppose $(c_1, c_2 \dots c_n)$ are cluster centers obtained in previous step, UOD is divided into n unequal intervals as: $[y_{min} - d, \frac{(c_1+c_2)}{2}]$, $[\frac{(c_1+c_2)}{2}, \frac{(c_2+c_3)}{2}]$, $[\frac{(c_2+c_3)}{2}, \frac{(c_3+c_4)}{2}]$, … $[\frac{(c_{n-1}+c_n)}{2}, y_{max} + d]$.

Step 4. Fuzzifying of the crisp data and obtaining their membership value:

This step fuzzify each data point to the interval index i which it belongs then find the membership value of the data point in that particular interval by min-max normalization operation. Let, y be the data point lie in the interval [a, b] then its membership value in interval is expressed as:

$$m = \frac{y-a}{b-a} \tag{3.2.9}$$

The interval number i and membership value m are taken as features of the data point.

Step 5. Normalizing the time series and setting an input output pattern for forecasting:

The crisp time series is normalized by min-max normalization and obtain the normalized time series. Suppose y be an element of crisp time series its normalized value $||y||$ is given as $||y|| = \frac{y-y_{min}}{y_{max}-y_{min}}$ here, y_{min} and y_{max} are the minimum and maximum observations of crisp time series. Then, we compose an input-output pattern between the features of crisp time series and normalized time series. The features of the previous $(t-1)$ time step are set as input to get the normalized value of t time step as output.

Step 6. Using SVM and MLP to establish relation between input and target values:

In this step, two deep learning techniques; SVM and MLP are applied to find the fuzzy logic relation between the input features and target values. Both techniques are trained using 80% of data and remaining 20% is used to test their performances. SVM is trained over the training dataset to find the best-fitted function $f(X)$ here, $X \in R_2$ is feature vector to predict the target value y. MLP is trained by backpropogation algorithm to minimize the sum squared error between actual and predicted output.

Step 7. Obtaining the test results and de-normalizing them:

After training, test features are fed to the trained SVM and MLP models and obtain the forecasted values. As the SVM and MLP models are trained over normalized target values so we get normalized predicted values for the test data as well. To obtain the actual values, the forecasted values are denormalized using min-max denormalization. Let, ‖y‖ be the normalized forecasted value the actual value y is obtained as follows:

$$y = \|y\|(b-a)+a \quad (3.2.10)$$

3.2.2.2 Stock portfolio construction by ranking forecasted time series

In this section, the stocks are selected based on their forecasted stock prices obtained by aforementioned method. The return and risk of stocks are estimated using forecasted stock prices with the help of the given formulae:

$$Return = \frac{Final\ stock\ price - Initial\ stock\ price}{Initial\ stock\ price} * 100$$

$$Risk = Standard\ deviation\ (r_1, r_2, r_3 \ldots r_n) * \sqrt{n}$$

here, n is the number observations and r_t are the daily returns.

The return-risk (RR) ratio is calculated, and stocks are ranked in increasing order of the RR ratio. Stocks with a higher RR ratio from a benchmark value are chosen as an asset of the portfolio.

3.2.2.3 Portfolio optimization using ACO

The main objective of portfolio construction is to find the best investment ratios for the assets so that the overall return is maximized while taking a reasonable risk for a given investment period. Sharpe ratio (SR) with some constraints is considered an objective function to find optimal investment ratios of the stocks in the portfolio constructed in the previous section. The objective function is defined as:

$$\left.\begin{aligned} & SR = max\ \frac{E\sum_{i=1}^{n}(r_i * x_i) - r_f}{\sigma_P} \\ & \text{subjected to the following conditions:} \\ & r_P > \alpha,\ v_P > \beta,\ x_i \geq x_j\ for\ i > j \\ & \textstyle\sum_{i=1}^{n} x_i = 1 \text{ and } m \leq x_i \leq M \quad \forall\ i \end{aligned}\right\} \qquad (3.2.11)$$

here, index i indicates the i^{th} rank stock,

r_i = expected return of the i^{th} stock,

x_i = fraction of the total investment to the i^{th} stock,

r_f = risk free rate,

$\sigma_P = \sqrt{\sum_{i=1}^{n} x_i^2 * \sigma_i^2 + \sum_{i=1}^{n}\sum_{i=1}^{n} x_i * x_j Cov_{ij}}$ = the standard deviation of portfolio return,

r_P = Portfolio return

v_P= portfolio risk or volatility

α = Minimum portfolio return

β = Maximum tolerable risk

The values of α, β, m and M are given as per the investor's choice. The constraint $x_i \geq x_j\ for\ i > j$ is added to give high investment ratio to high rank asset. The objective function in equation (3.2.11) is a risk adjusted return ratio of the difference of portfolio return and risk free rate to the portfolio standard deviation. Higher value of the objective function is considered as better investment. ACO is applied to optimize the objective function. The pseudo code of ACO algorithm for portfolio optimization is given below:

Algorithm 1: Pseudo code of ACO algorithm for portfolio optimization

1. ***Procedure ACO-Portfolio***
2. *Generate N random solution nodes based on equation (3.2.11);*
3. *Initialize the ACO;*
4. ***for** ITERATION=1 to I do*
5. ***for** ANT=1 to C do*
6. *Select the start node randomly;*
7. ***for** LIFETIME = 2 to L do*
8. *Select next node based on the heuristic information and pheromone concentration in the path. Move to the next node only if it is better than the current node.*
9. *Update pheromone on the selected path;*
10. ***End for***
11. *Store the objective value and the path details of the final node reached by each ant;*
12. ***End for***
13. *Identify the solution node where maximum number of ants have reached and consider that to be the optimum solution for the current iteration;*
14. *Update the pheromone on the path of each ants who have reached this optimum solution;*
15. *Evaporate the pheromone from all paths.*
16. ***End for***
17. ***End procedure***

3.3 Model [3]: An effective hybrid MCDM approach of portfolio construction using modern portfolio theory

Markowitz's (1952) modern portfolio theory (MPT) was based on the two main concepts: (i) Investors are risk-averse i.e. investors always prefer owning a portfolio with less risk. (ii) Risk can reduce by diversifying a portfolio through individual and unrelated assets. MPT quantifies the benefits of diversification. By selecting less and inversely correlated securities of the same category or different categories investors can own a diversified portfolio and reduce the risk of loss. Beside diversification or risk control, ultimately the goal of an investor is to gain maximum return. Hence, at the same time investor attempts to assemble the portfolio with the stocks having good possibility of growing in the future. Many researchers have proposed various MCDM based methodologies of building portfolio over the years **(Aouni *et al*., 2018, Thakur *et al.*, 2018, Narang *et al.*, 2021, Xidonas *et al.*, 2011, Xidonas and Psarras, 2009)**. Some of the researchers include diversification as an objective or constraint while formulating multi-objective portfolio optimization functions **(Ehrgott *et al.*, 2004, García *et al.*, 2013, Huang and Jiang, 2021)**. While MCDM ranking methods are adopted only for ranking-based selection of stocks, without concerning about the diversification in the selection. Table 3.3.1 shows some of the MCDM ranking-based portfolio construction models developed so far. Avoiding diversification at the stage of stock selection could be harmful for an investor. Addressing the two essential aspects of portfolio construction i.e. diversification and profit-oriented, in this paper we proposed a new hybrid MCDM approach that incorporates MPT assumptions.

There are two main objectives while solving MCDM problems first is the assignment of weights to different criteria and second is the evaluation of preference order of the alternatives **(Rezaei, 2015)**. So as in stock selection also, firstly determining the optimal relative weights of criteria is a crucial step. BCM method is employed for this study to get the leverage of its advantages in computing the criteria weights. In BCM, one of the criteria is chosen by the experts as a base-criterion. The pairwise comparisons $(a_{i,j})$ of other criteria depend on the base comparisons $a_{Base,i} * a_{i,j} = a_{Base,j}$ on the numerical scale of 1/9 to 9. Afterward, min-max problem is defined to obtain the optimal weights of the criteria. BCM method provides fully consistent solution with minimum consistency ratio $\xi = 0$, it require less pair-wise comparison hence it is less computational. But there are probable chances of having inconsistency and vagueness in the collected information. To deal with this, we extended BCM in neutrosophic environment and discussed the incorporation of single-valued triangular neutrosophic number with BCM for the first time. Under neutrosophic environment the relative importance of attributes recommended by the decision maker are characterized in three degrees (truthness, indeterminacy and falsity) and it unfolds the inconsistent, indeterminate and hesitant nature of the real information of a particular problem in stronger manner.

After calculating the weights of the criteria, another task is to rank alternatives or stocks. Researchers also used various MCDM ranking methods in the field of financial management and portfolio construction. Different from other ranking methods, the comparison of alternatives in PROMETHEE partial ranking method is seem to be more realistic and reliable for stock selection. This method is based on the pairwise comparisons between finite alternatives under finite criteria. Complete ranking methods reduce the multi-criteria problem into single criterion by a utility function for which every alternative gets a unique rank and preference order. The preference order could be unrealistic as it is completely relies on strong assumptions considered in the form of utility function. It may change the structure of MCDM problem. In PROMETHEE I method, the pairwise comparisons between the alternatives suggest whether “a is preferred over b” or “a and b are indifferent” or “a and b are incomparable”. Decision makers try to reduce incomparabilities in terms of making an optimal solution but withdrawing all the incomparabilities can make a disputable solution. PROMETHEE I provide a fair insight about the relationship between the alternatives, decision maker can make a useful and realistic decision aid according the demand of the particular MCDM problem. Fundamental indicators are commonly used as stock selection criteria. However, these criteria have very fragile definition in the dynamic market. Furthermore, PROMETHEE offers a number of preference functions to define different criteria. PROMETHEE method's comparison of alternatives helps to identify similar or dissimilar stocks, which can aid in diversification. Motivated by abovementioned facts, this model introduces a stock selection method that combines the PROMETHEE comparison relation with the concept of correlation between pairs of alternatives. Using the comparison relation and partial ranking obtained by PROMETHEE method, in this study we merge the concept of correlation between the pairs of alternatives to select the diverse stock for portfolio. The amalgamation of partial ranking system with correlation benefitted to make rank wise as well as diversified selection.

After selection of stocks, the main concern is to find the optimal investment portion of each asset out of total investment which satisfies the investor’s target. In this study, we formulated an optimization function for determining the rank-wise ratio allocation to the securities and simulate it by PSO technique. Preference wise weight allocation to the diverse stocks achieved the desired gain ensuring the limit of risk.

Table 3.3.1: MCDM based methods for portfolio construction or stock selection

Article	Portfolio construction approach	Used MCDM ranking methods	Diversification during stock selection	Real Source Data
Marasović and Babić (2011)	Two-step MCDM based model: choice the securities of different industries and choice the portfolio for each industry	Modified MCDM based on PROMETHEE II	Not Addressed	Zagreb Stock Exchange
Fazli and Jafari (2012)	Hybrid MCDM model for investment stock exchange	DEMATEL to build a relations-structure among criteria and AHP for criteria weight assessment, VIKOR for ranking	Not Addressed	Iran Stock exchange
Poklepović and Babić (2014)	Hybrid various MCDM rankings on the basis of Spearman’s rank correlation coefficient to evaluate best and worst stock	COPRAS, linear assignment, PROMETHEE, SAW and TOPSIS; AHP for criteria weight	Not addressed	Croatian capital market
Thakur *et al.* (2018)	Criteria having low correlation obtained by fuzzy Delphi method considered stock analysis and ant colony optimization are used for maximizing return and minimizing risk.	Dempster– Shafer evidence theory for ranking	Not addressed	Bombay Stock Exchange
Mills *et al.* (2020)	Hybrid grey MCDM based ranking based method	Integrating AHP and DEMATEL in grey environment	Not addressed	Shanghai stock exchange
Narang *et al.* (2021)	Hybrid MCDM rank based selection of stock	Integrating BCM and COPRAS in fuzzy environment	Not addressed	National stock exchange of India

3.3.1 Preliminaries

Single valued triangular neutrosophic numbers (SVTNs) and their properties are discussed in this section.

3.3.1.1 Single valued triangular neutrosophic numbers (SVTNs)

Let X be universe of discourse and a generic element $u \in X$. A single-valued neutrosophic set A in X is defined as follows:

$$A = \{u, < T_A(u), I_A(u), F_A(u) >\} \qquad \forall\, u \in X$$

$$T_A(u), I_A(u), F_A(u) \in (0,1)\ and\ {}^{-}0 \leq T_A(u) + I_A(u) + F_A(u) \leq 3^{+}$$

here, $T_A(u), I_A(u), F_A(u)$ represents the truth membership function, the indeterminacy membership function and the falsity membership function respectively, $T_A(u): X \to\ ]^{-}0, 3^{+}[$, $I_A(u): X \to\ ]^{-}0, 3^{+}[$ and $F_A(u): X \to\ ]^{-}0, 3^{+}[$.

A SVTN Ã is defined as following mathematical object:

$$\tilde{A} = \langle (a_1, b_1, c_1); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle$$

here, a_1, b_1, c_1 are lower, median and upper triangular neutrosophic number respectively, $\alpha_{\tilde{A}}$, $\beta_{\tilde{A}}$, $\gamma_{\tilde{A}}$ are the maximum truth membership function, the minimum indeterminacy membership function and the minimum falsity membership function respectively. $\alpha_{\tilde{A}}$, $\beta_{\tilde{A}}$, $\gamma_{\tilde{A}}$ are determined as follows:

$$T_{\tilde{A}}(u) = \begin{cases} \alpha_{\tilde{A}}\left(\dfrac{u - a_1}{b_1 - a_1}\right) & (a_1 \leq u \leq b_1) \\ \alpha_{\tilde{A}} & (u = b_1) \\ \alpha_{\tilde{A}}\left(\dfrac{c_1 - u}{c_1 - b_1}\right) & (b_1 \leq u \leq c_1) \\ 0 & otherwise \end{cases}$$

$$I_{\tilde{A}}(u) = \begin{cases} \left(\dfrac{b_1 - u + \beta_{\tilde{A}}(u - a_1)}{b_1 - a_1}\right) & (a_1 \leq u \leq b_1) \\ \beta_{\tilde{A}} & (u = b_1) \\ \left(\dfrac{u - b_1 + \beta_{\tilde{A}}(c_1 - u)}{c_1 - b_1}\right) & (b_1 \leq u \leq c_1) \\ 0 & otherwise \end{cases}$$

$$F_{\tilde{A}}(u) = \begin{cases} \left(\dfrac{b_1 - u + \gamma_{\tilde{A}}(u - a_1)}{b_1 - a_1}\right) & (a_1 \le u \le b_1) \\ \gamma_{\tilde{A}} & (u = b_1) \\ \left(\dfrac{u - b_1 + \gamma_{\tilde{A}}(c_1 - u)}{c_1 - b_1}\right) & (b_1 \le u \le c_1) \\ 0 & otherwise \end{cases}$$

3.3.1.2 Basic mathematical operators on SVTNs

Let $\tilde{A}_1 = \langle (a_1, b_1, c_1); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle$ and $\tilde{A}_2 = \langle (a_2, b_2, c_2); \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_2} \rangle$ are two SVTNs. The basic mathematical operations on $\tilde{A}_1$ and $\tilde{A}_2$ are as follows:

i. $\tilde{A}_1 + \tilde{A}_2 = \langle (a_1 + a_2, b_1 + b_2, c_1 + c_2); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle$

ii. $\tilde{A}_1 - \tilde{A}_2 = \langle (a_1 - a_2, b_1 - b_2, c_1 - c_2); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle$

iii. $\tilde{A}_1 * \tilde{A}_2 =$

$$\begin{cases} \langle (a_1 * a_2, b_1 * b_2, c_1 * c_2); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle & if c_1, c_2 > 0 \\ \langle (a_1 * c_2, b_1 * b_2, c_1 * a_2); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle & if c_1 < 0, c_2 > 0 \\ \langle (c_1 * c_2, b_1 * b_2, a_1 * a_2); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle & if c_1, c_2 < 0 \end{cases}$$

iv. $$\frac{\tilde{A}_1}{\tilde{A}_2} = \begin{cases} \langle \left(\frac{a_1}{c_2}, \frac{b_1}{b_2}, \frac{c_1}{a_2}\right); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle & if c_1, c_2 > 0 \\ \langle \left(\frac{c_1}{c_2}, \frac{b_1}{b_2}, \frac{a_1}{a_2}\right); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} \rangle & if c_1 < 0, c_2 > 0 \\ \left(\frac{c_1}{a_2}, \frac{b_1}{b_2}, \frac{a_1}{c_2}\right); \alpha_{\tilde{A}_1} \wedge \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_1} \vee \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_1} \vee \gamma_{\tilde{A}_2} & if c_1, c_2 < 0 \end{cases}$$

v. $\tilde{A}_1^{-1} = \langle \left(\frac{1}{c_1}, \frac{1}{b_1}, \frac{1}{a_1}\right); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle$

vi. $$\lambda \tilde{A}_1 = \begin{cases} \langle (\lambda a_1, \lambda b_1, \lambda c_1); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle if \lambda > 0 \\ \langle (\lambda c_1, \lambda b_1, \lambda a_1); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle if \lambda < 0 \end{cases}$$

vii. $$\frac{\tilde{A}_1}{\lambda} = \begin{cases} \langle \left(\frac{a_1}{\lambda}, \frac{b_1}{\lambda}, \frac{c_1}{\lambda}\right); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle if \lambda > 0 \\ \langle \left(\frac{c_1}{\lambda}, \frac{b_1}{\lambda}, \frac{a_1}{\lambda}\right); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle if \lambda > 0 \end{cases}$$

3.3.1.3 Score and accuracy degrees of SVTNs

Let $\tilde{A}_1 = \langle (a_1, b_1, c_1); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle$ be a SVTN then

$$S(\tilde{A}) = \frac{1}{8}[a_1 + b_1 + c_1][2 + \alpha_{\tilde{A}} - \beta_{\tilde{A}} - \gamma_{\tilde{A}}]$$

$$A(\tilde{A}) = \frac{1}{8}[a_1 + b_1 + c_1][2 + \alpha_{\tilde{A}} - \beta_{\tilde{A}} + \gamma_{\tilde{A}}]$$

here, $S(\tilde{A})$ and $A(\tilde{A})$ are score and accuracy degrees of$\tilde{A}$, rescpectively.

3.3.1.4 Comparison of two SVTNs

Let $\tilde{A}_1 = \langle (a_1, b_1, c_1); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1} \rangle$ and $\tilde{A}_2 = \langle (a_2, b_2, c_2); \alpha_{\tilde{A}_2}, \beta_{\tilde{A}_2}, \gamma_{\tilde{A}_2} \rangle$ are two SVTNs

1. If $S(\tilde{A}_1) < S(\tilde{A}_2)$, then $\tilde{A}_1 < \tilde{A}_2$
2. If $S(\tilde{A}_1) = S(\tilde{A}_2)$;
   a. $A(\tilde{A}_1) < A(\tilde{A}_2)$; then $\tilde{A}_1 < \tilde{A}_2$
   b. $A(\tilde{A}_1) = A(\tilde{A}_2)$; then $\tilde{A}_1 = \tilde{A}_2$

3.3.2 Proposed model

Model [3] comprises four sections and its overview is depicted in Figure 3.3.1. In section 3.3.2.1, neutrosophic base-criterion method (NBCM) is developed for weight determination of selection criteria. Section 3.3.2.2 is the PROMETHEE I procedure for ranking alternatives (stocks). Modern portfolio based stock selection procedure is proposed in section 3.3.2.3 and portfolio optimization using PSO is done in section 3.3.2.4.

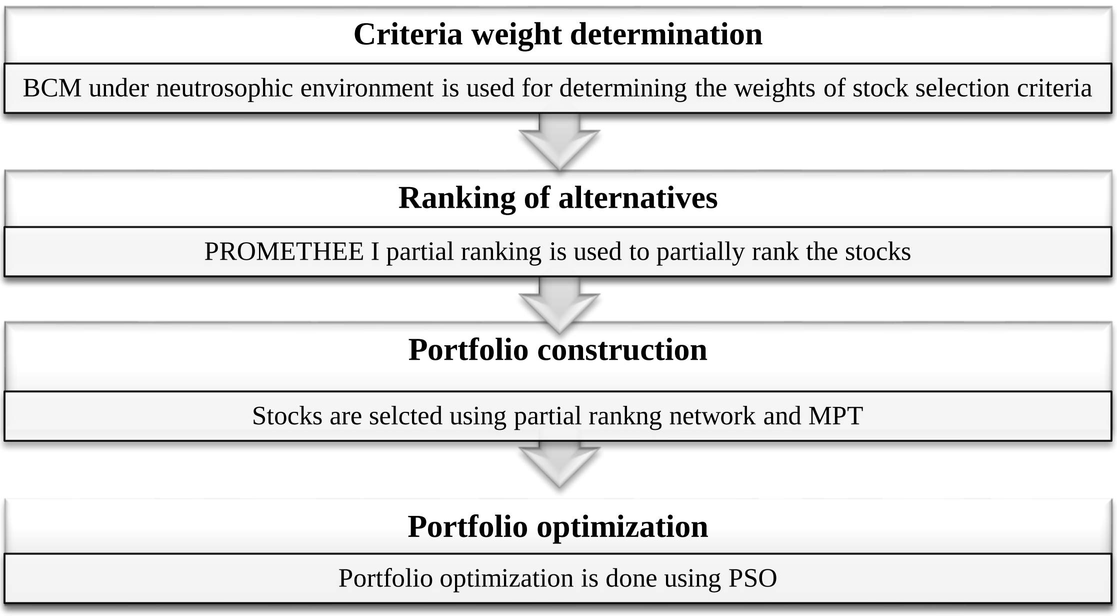


Figure 3.3.1: Overview of Model [3]

3.3.2.1 Neutrosophic base-criterion method (NBCM)

The step wise procedure of criteria weight determination through BCM under single valued triangular neutrosophic environment is as follows:

Step 1. Identify the decision making problem, specify the alternatives and criteria

First step is to identify the real life decision making problem and selecting a well experienced expert in the domain. Then specifying the preferably available alternatives $\{A_1, A_2, \ldots\ A_m\}$ and all the desired criteria $\{C_1, C_2, \ldots\ C_n\}$.

Step 2. Select the base-criterion and determine the relative importance of base-criteria to other criteria

In this step, one of the criteria has to be selected as the base-criteria. Then determine the pairwise comparison between the base-criterion and other criteria as per the expert opinion on the triangular neutrosophic scale given in Table 3.3.2.

Table 3.3.2: STVNs for pairwise comparison

Linguistic term	SVTNs	Crisp number (Saaty 1997)
Equally important	$\langle(1,1,1); 0.5,0.5,0.5\rangle$	1
Moderate important	$\langle(2,3,4); 0.3,0.75,0.7\rangle$	3
Strong important	$\langle(4,5,6); 0.8,0.15,0.2\rangle$	5
Very strong important	$\langle(6,7,8); 0.9,0.1,0.1\rangle$	7
Extreme important	$\langle(9,9,9); 1,0,0\rangle$	9
Intermediate values	$\langle(1,2,3); 0.4,0.65,0.6\rangle$	2
	$\langle(3,4,5); 0.6,0.35,0.4\rangle$	4
	$\langle(5,6,7); 0.7,0.25,0.3\rangle$	6
	$\langle(7,8,9); 0.9,1.0,1.0\rangle$	8
Reciprocal	$\langle(a^{-1}, b^{-1}, c^{-1}); \alpha_{\tilde{A}}, \beta_{\tilde{A}}, \gamma_{\tilde{A}}\rangle$	x^{-1}

Step 3. Determine triangular neutrosophic base comparison vector and transform to a deterministic term

In this step, a comparison vector in triangular neutrosophic scale between base and other criterion is obtained on the basis of expert's opinion such as:

$\tilde{A}_{Base} = (\langle(a_1, b_1, c_1); \alpha_{\tilde{A}}, \beta_{\tilde{A}}, \gamma_{\tilde{A}}\rangle, \langle(a_2, b_2, c_2); \alpha_{\tilde{A}}, \beta_{\tilde{A}}, \gamma_{\tilde{A}}\rangle \ldots \langle(a_n, b_n, c_n); \alpha_{\tilde{A}}, \beta_{\tilde{A}}, \gamma_{\tilde{A}}\rangle).$

The components of comparison vector are converted to deterministic value using are score accuracy degrees of Ã. The transformed comparison vector is defined as follows:

$$\tilde{A}^{*}_{Base} = \{a_{B1}, a_{B2} \ldots a_{Bn}\}$$

Step 4. Determine the optimum weights of criteria $\{w_1{}^*, w_2{}^* \ldots w_n{}^*\}$

The optimal weight w_j of the criterion c_j should satisfy $\frac{w_B}{w_J} = a_{Bj}$. It can be obtained by minimizing the absolute difference $\left|\frac{w_B}{w_j} - a_{Bj}\right|$. Considering, the non negative constraint and sum condition of weights we find the solution of the following min max problem:

$$\min \max \left|\frac{w_B}{w_j} - a_{Bj}\right| \quad \text{satisfying the following:} \tag{3.3.1}$$

$$\sum w_j = 1$$

and $w_j \geq 0$ for all j.

The above problem can be expressed in a minimization problem as follows:

$$\text{Min } \xi \quad \text{satisfying the following:} \tag{3.3.2}$$

$$\left|\frac{w_B}{w_j} - a_{Bj}\right| \leq \xi$$

$$\sum w_j = 1$$

and $w_j \geq 0$ for all j.

The pairwise comparison of criteria in BCM method done under the principle that $a_{Base,i} * a_{i,j} = a_{Base,j}$ and $\frac{1}{9} \leq a_{ij} \leq 9$. This ensure the fully consistent solution of the minimization problem in equation (3.3.2) and gives $\xi = 0$. Thus, the optimal weights $\{w_1{}^*, w_2{}^* \dots w_n{}^*\}$ are obtained.

3.3.2.2 PROMETHEE I partial ranking method

The PROMETHEE method start with evaluation table and two addition information is required for its implementation such as: (i) relative importance (weights) of the criteria (ii) preference function for pairwise comparisons of alternatives for each criterion separately.

Suppose a multi-criteria decision making problem as follows:

$$max\{f_1(a), f_2(a), \dots, f_k(a) | \, a \in A\} \tag{3.3.3}$$

here, $A = \{a_1, a_2, \dots, a_n\}$ is the set of alternatives and $\{f_1(a), f_2(a), \dots, f_k(a)\}$ is the set of evaluation of a alternative for k criteria. The preference relation between two alternatives a_i and a_j $(i \neq j)$ for a particular criterion is evaluated by preference function P measuring the deviation between the evaluations of a_i and a_j lies in the interval [0,1]. Hence, for k^{th} criterion, the preference function P_k for alternatives a_i and a_j is formulated as:

$$P_k(a_i, a_j) = G_k[d_k(a_i, a_j)]$$

such that: $0 \leq P_k(a_i, a_j) \leq 1$

here, G_k is a non-decreasing function of the deviation between a_i and a_j calculated as $d_k(a_i, a_j) = f_k(a_i) - f_k(a_j)$.

For cost criteria the preference function is reversed and defined as:

$$P_k(a_i, a_j) = G_k[-d_k(a_i, a_j)]$$

There are basically six types of preference functions proposed by **Brans *et al.* (1984)** (Usual, Quasi, V-shape, Level, Linear and Gaussian) for the convenience in decision making in order to select appropriate preference function for particular criteria. There are three threshold parameters namely, Preference threshold (p), Indifference threshold (q) and Gaussian threshold (s), out of these zero, one or two have to define for implementation the preference functions. Preference threshold (p) stands for the threshold value of strict preference i.e. $P_k(a_i, a_j) = 1$; it is the smallest deviation claiming the preference of one alternative to the other. Indifference threshold (q) stands for threshold value of indifference; it is the largest deviation considering the negligence of the decision maker for the particular criteria. Gaussian threshold (s) is defined only for Gaussian preference function and it is set for an intermediate value lying between q and p.

After defining the preference function for each criterion the aggregated preference indices are calculated. For the alternatives a_i and a_j , the aggregated preference indices are defined as:

$$\begin{cases} \pi(a_i , a_j) = \sum_{k=1}^{m} P_k(a_i, a_j) * w_k \\ \pi(a_j , a_i) = \sum_{k=1}^{m} P_k(a_j, a_i) * w_k \end{cases} \tag{3.3.4}$$

It holds the following properties:

i. $\pi(a_i , a_j) = 0$ for $i = j$

ii. $0 \le \pi(a_i , a_j), \pi(a_j , a_i) \le 1$

iii. $0 \le \pi(a_i , a_j) + \pi(a_j , a_i) \le 1$

After computing the aggregated preference indices for each pair of alternatives, following two types of outranking flows for each alternative are obtained:

i. **Positive outranking flow:** For an alternative a_i the positive outranking flow $\varphi^+(a_i)$ measures the outranking of a_i to the rest alternatives i.e. the dominance of alternative a_i to all other alternatives. Hence, the alternative a_i is better for high value of $\varphi^+(a_i)$.

$$\varphi^+(a_i) = \sum_{j=1}^{n} \pi(a_i , a_j) \tag{3.3.5}$$

ii. **Negative outranking flow:** For an alternative a_i the negative outranking flow $\varphi^-(a_i)$ measures how a_i is outranked by the rest alternatives i.e. the dominance of all other alternatives on alternative a_i. Hence the alternative a_i is better for low value of $\varphi^-(a_i)$.

$$\varphi^-(a_i) = \sum_{j=1}^{n} \pi(a_j , a_i) \tag{3.3.6}$$

The PROMETHEE I partially rank the alternatives on the basis of positive and negative ranking flows. The alternative a_i is preferred to alternative a_j, if positive outranking flow of a_i is greater than positive outranking flow of a_j and negative outranking flow of a_i is lesser than negative outranking flow of a_j.

$$(a_i\ P\ a_j) \text{ if : } \varphi^+(a_i) > \varphi^+(a_j) \text{ and } \varphi^-(a_i) < \varphi^-(a_j)\text{, or}$$

$$\varphi^+(a_i) > \varphi^+(a_j) \text{ and } \varphi^-(a_i) = \varphi^-(a_j)\text{, or}$$

$$\varphi^+(a_i) = \varphi^+(a_j) \text{ and } \varphi^-(a_i) < \varphi^-(a_j)$$

If the positive and negative outranking flows of alternative a_i are calculated equal to the positive and negative outranking flows of alternative a_j, respectively then the alternatives are consider as indifference alternatives.

$$(a_i\ I\ a_j) \text{ if : } \varphi^+(a_i) = \varphi^+(a_j) \text{ and } \varphi^-(a_i) = \varphi^-(a_j)$$

If the positive and negative ranking flows of the alternatives a_i and a_j are indicating the contradictory situations then the alternatives are considered as incomparable. This often happens when alternative a_i is better than alternative a_j for some criteria meanwhile alternative a_j is better than alternative a_i for remaining criteria and difficult to compare.

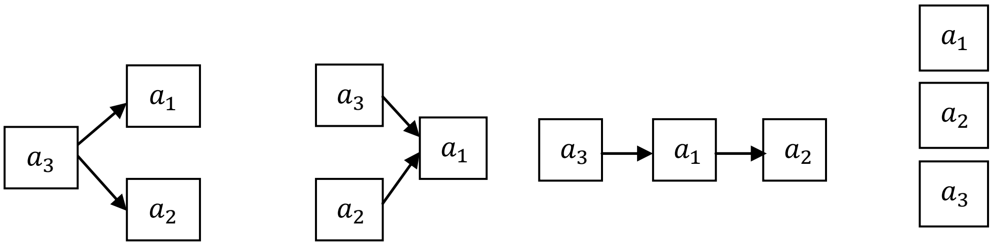
$$(a_i\ R\ a_j) \text{ if : } \varphi^+(a_i) > \varphi^+(a_j) \text{ and } \varphi^-(a_i) > \varphi^-(a_j)\text{, or}$$

$$\varphi^+(a_i) < \varphi^+(a_j) \text{ and } \varphi^-(a_i) < \varphi^-(a_j)$$

3.3.2.3 Stock selection procedure using MPT

In this section, the stock selection procedure is proposed for constructing a diversified and profitable portfolio by applying the principles of MPT with comparison results obtained by PROMETHEE I. The correlation coefficient (r) between the returns of two stocks reflects relation between their movements. It lies between -1 to +1. Two stocks are said to be perfectly correlated if the correlation coefficient is +1, which means both the stocks move in the same direction (both gain and loss simultaneously). Correlation coefficient -1 reflects perfect negative correlation, i.e., gain of one stock proportional to the loss of another stock. If the correlation coefficient is 0, there is no predictive relationship between the assets. MPT emphasizes selecting uncorrelated securities (having a correlation coefficient near 0) to limit the risk.

We take two actions for double assurance of diversification in the selection, (i) not selecting incomparable stocks simultaneously and (ii) selecting less correlative stocks among the incomparable stocks. The comparison of stocks by PROMETHEE can be visualized by PROMETHEE partial ranking network. The stocks have the very least possibility of being indifferent; therefore, we can ignore the condition in the partial ranking network. In the partial ranking network, the following four cases can occur:



Case 1	Case 2	Case 3	Case 4
$\{(a_3\ P\ a_1), (a_3\ P\ a_2), (a_1\ R\ a_2)\}$	$\{(a_3\ R\ a_2), (a_3\ P\ a_1), (a_2\ P\ a_1)\}$	$\{(a_3\ P\ a_1), (a_3\ P\ a_2), (a_1\ P\ a_2)\}$	$\{(a_3\ R\ a_1), (a_3\ R\ a_2), (a_1\ R\ a_2)\}$

Case 1: It depicts the situation in which an alternative a_3 is preferred over two incomparable options, a_2 and a_1. In such case, the selection of stocks from the set is done based on the flow chart given in Figure 3.3.2.

Case 2: It displays the network which starts with two incomparable options a_3 or a_2 preferred over a_1 . In this situation, the selection of stocks from the set is done based on the flow chart given in Figure 3.3.3. The reason for selecting the stock with the highest net flow in equivalent correlation situations is to select a stock that ranks higher by considered criteria analysis.

Case 3: It shows the situation where all the alternatives are comparable. Here a_3 is chosen first, then the stock a_2 or a_1 with the lowest correlation with a_3 is selected. If both a_2 and a_1 have same correlation with a_3 then $\mu = mean(\varphi(a_1), \varphi(a_2), \varphi(a_3))$ is computed. After that, the stock from the remaining stocks (i.e. a_1, a_2) is selected if it meets the condition $|\mu - \varphi(a_i)| < \frac{\mu}{2}$. This ensures to choose the stocks with sufficient net flow differences and high net flow to provide some diversification in the selection.

Case 4: It displays the last possible scenario where all the alternatives are incomparable. In this case, firstly the alternative with highest net flow is picked then other alternative with least correlation with previously selected alternative is chosen. In the case of same correlation, the condition given in case 3 is checked.

The best fitted case among all these four cases is used for $n \geq 3$ alternatives, likely to result in a diverse and profitable selection.

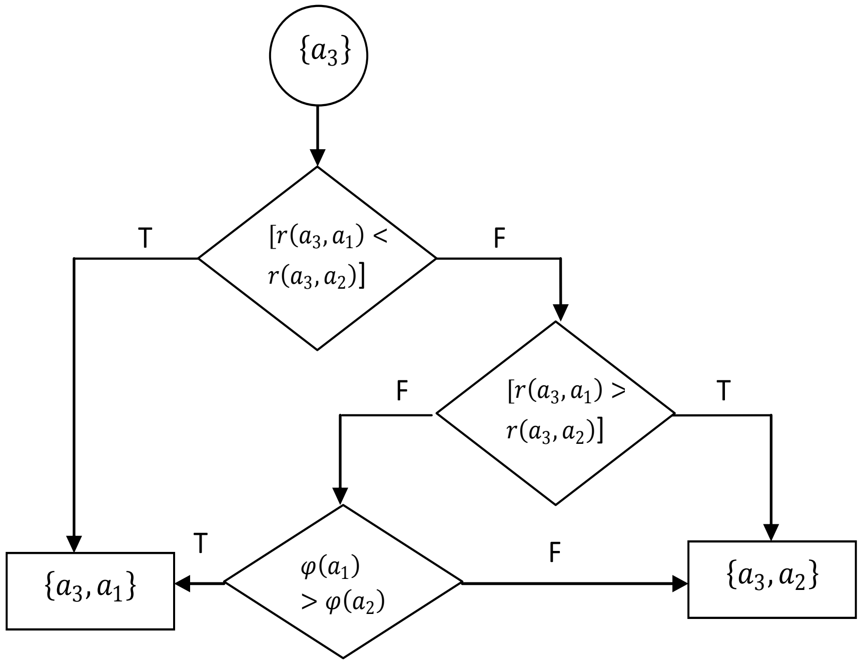


Figure 3.3.2: Stock selection process for case 1.

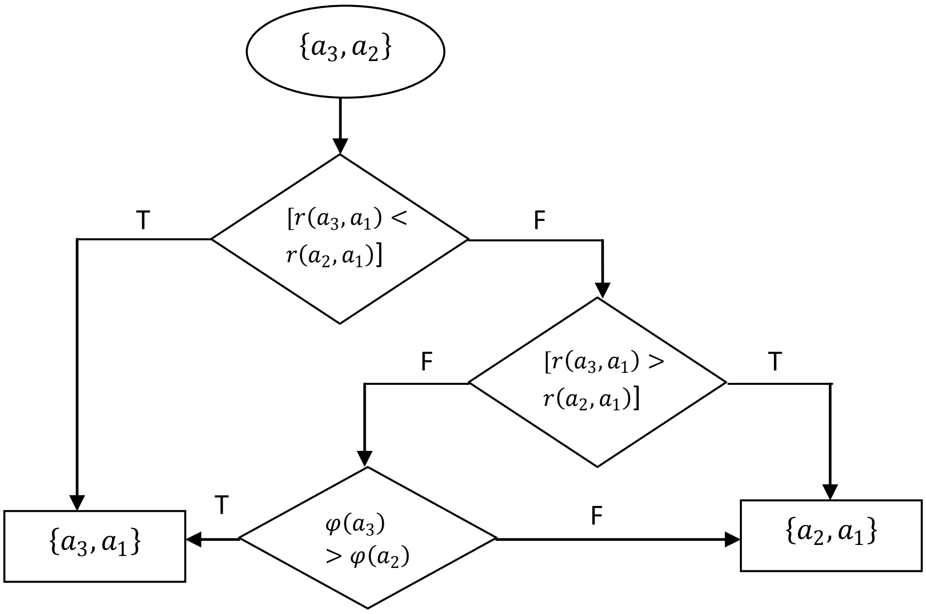


Figure 3.3.3: Stock selection process for case 2.

3.3.2.4 Portfolio Optimization using PSO

In this model, we again consider the SR as an optimization function with the same constraints as in section 3.2.2.3 of Model [2] to obtain the rank wise optimal investment ratios for the stocks in the proposed portfolio. An algorithm using PSO is implemented to optimize SR. Pseudo code of the implemented algorithm is given below.

Algorithm 2: Pseudo code of PSO algorithm for portfolio optimization

1. *fitness function* $(f(x) = -F(x)$ $(defined\ in\ equation$ 3.2.11)
2. ***for*** *i =1 to n (number of particles): (at t=0)*
3. *Initialize random solution* $p_i = (x_1, x_2, \ldots, x_d)$ *of* $f(x)$ *as local position and velocity* (v_i)
4. ***End for***
5. *Identify the best solution* p_g *of the fitness function*
6. *set* ω, c_1 *and* c_2
7. ***While*** *(t<max iteration):*
8. ***for*** *i =1 to n (number of particles):*

9. $v_i^{t+1} = \omega * v_i^t + c_1 rand_1(p_i^t - x_i^t) + c_2 rand_2(p_g^t - x_i^t)$

10. $x_i^{t+1} = x_i^t + v_i^{t+1}$

11. $p_i^{t+1} = p_i^t$

12. $p_g^{t+1} = p_g^t$

13. *Evaluate fitness function* $\left(f(x_i^{t+1})\right)$*;*

14. ***if*** $f(p_i^{t+1}) < f(x_i^{t+1})$*:*

15. *Update local best position* p_i^{t+1}

16. ***End if***

17. ***if*** $f\left(p_g^{t+1}\right) < f(p_i^{t+1})$*:*

18. *Update global best position* p_g^{t+1}

19. ***End if***

20. ***End for***

21. *Update* $t = t + 1$

22. ***End while***

3.4 Model [4]: A portfolio construction model based on sector analysis using Dempster-Shafer evidence theory and Granger causal network

There are thousands of companies traded in the stock market and the companies having similar business activities or services are grouped together in a sector. For investing in the stock market, there could be two approaches of portfolio selection. First, by analyzing the performance of stocks individually and making a selection believing in the fact that the selected stocks will be profitable in the future **(Bagheri, 2019, Huang, 2012, Poklepović and Babić, 2014, Vuković *et al.*, 2020)**. Second, by analyzing the sectors performance under the current economic conditions and selecting the stocks from leading sectors **(Antonakakis *et al.*, 2018, Coelho *et al.*, 2007, Gupta and Basu, 2009).** The second method of portfolio selection is preferred by economists over the first since it is based on the direct impact of developing and emerging local or global economies on the sectors. All the sectors of economy do not grow with the same rate. As a result, the performance of sectors can have a significant impact on portfolio selection. It is also a more convenient approach as compared to digging out some potential stocks from a large number of stocks enlisted in an exchange. Stocks of leading sectors could provide ample opportunities to earn good profit.

In **Markowitz (1952)** modern portfolio theory (MPT), when the number of stocks increases, the mean-variance model of MPT becomes computationally difficult due to quadratic utility functions and covariance matrix **(Yunusoglu and Selim, 2013)**. So, in this case also portfolio construction based on sector analysis is found more comfortable and it can also provide diversity by selecting stocks from non-correlated industries.

In this model, we aim to do portfolio selection involving two stages: (i) selecting strong and diversified sectors, and (ii) selecting the best stocks from those sectors. The first stage further involves two sub-stages: first is the selection of strong or leading economic sectors, and the second is to identify diversification among the strong sectors. Different types of factors such as government policies, politics, natural calamities, investor sentiments, supply and demand, exchange rates, etc. affect the sectors in different ways **(Joshi, 2013)**. But some common criteria or technical indicators based on the prior performances of sectors can be helpful to identify potential sectors of the future. Therefore, the problem of selection of the best sectors to invest could be framed as a MCDM problem. D-S theory of evidence has been a popular concept for aiding the MCDM problem among researchers **(Beynon *et al.*, 2001, Claussmann *et al.*, 2018, Dutta, 2015, Silva and Filho, 2016)**. However, its use in portfolio construction has received little attention. Its mechanism is based on the evidences collected by different sources which are efficiently handled and effectively combined to draw a conclusion out of it. Therefore, its application for ranking the dynamic and uncertain financial sectors is completely justified.

The sector-based approach is considered the more diversified method of investing which avoids the risk of targeting an individual company and it also helps in reducing unsystematic risk. Two sectors could relate to each other in many possible ways. The movement of one sector could drive the movement of another (i.e., prices of one sector can forecast prices of another), both sectors could move in the same direction or in opposing directions, or both sectors could have no influence on each other. In time series econometrics, the concept of Granger causality has been widely adopted for verifying the interdependencies between the sectors using historical financial time series. Granger causality has been used in a number of studies for assessing the dependencies in financial systems at local and global levels **(Ahmed *et al.*, 2018, Al-yahyaee *et al.*, 2019, Campbell, 2004, Hong *et al.*, 2009, Marisetty, 2021, Papana *et al.*, 2017, Peng *et al.*, 2020, Stawiarski, 2021, Tang *et al.*, 2019, Výrost *et al.*, 2015)**.

In this study, we introduce D-S theory for ranking different sectors based on different evidences. To diversify our choices the interrelation between strong sectors are determined by analyzing their prices movements in the past years and with the help of the Granger causal network the non-dependable sectors are identified. Following that, we determined the investment portion or how much to invest in a particular sector according to their ranks obtained by D-S theory. Stocks of top companies listed in each sector are analyzed to find the leading ones. Then, some leading companies from the selected sectors are grouped as a portfolio. At last, deep learning based recurrent neural network is applied to optimize the portfolio.

3.4.1 Preliminaries

This section provides the brief notes on Dempster-Shafer theory and Granger causality test.

3.4.1.1 Dempster- Shafer theory (D-S theory)

D-S theory also known as the theory of evidence or the theory of belief functions is a generalization of the Bayesian theory of subjective probability. It provides a mathematical framework for modelling uncertainty and a powerful method for combining the degree of evidences collected from different sources. Suppose, $\Theta = \{\theta_1, \theta_2 \ldots \theta_n\}$ referred as the frame of discernment (FOD), is a finite set of mutually exclusive and exhaustive propositions. The set of the subsets of Θ, i.e. 2^Θ include all the events and each subset or proposition will be assigned a value like probability based on evidence. The value 0 indicates no belief in the proposition; the value 1 indicates total belief in the proposition and the value lying between 0 and 1 shows partial belief in the proposition. Hence, a basic probability assignment (BPA) or mass function (m) over Θ described as a mapping $m: 2^\Theta \rightarrow [0,1]$ satisfying the following condition:

$$m(\emptyset) = 0 \text{ and } \sum_{A \epsilon 2^\Theta} m(A) = 1$$

here, A is one the propositions in 2^Θ and called focal element if $m(A) > 1$.

The belief function for A represented by $Bel(A)$ measure the total belief committed to A by adding the mass of all proper subsets of A.

$$Bel(A) = \sum_{B \subset A} m(B) \tag{3.4.1}$$

and, the plausibility function $pl(A)$ is defined as

$$Pl(A) = \sum_{B \cap A \neq \emptyset} m(B) \tag{3.4.2}$$

Shafer interpreted $Bel(A)$ and $Pl(A)$ as the bounds for the probability of A i.e. $Bel(A) \leq P(A) \leq Pl(A)$. Hence, suggested to assign an interval $[Bel(A), Pl(A)]$ to each set of hypotheses within degree of belief of each hypothesis must lie. The evidences collected from different sources are combined by using Dempster's rule of combination. Suppose, m_1 and m_2 are the mass functions of two evidences A and B respectively, the Dempster's rule of combination is defined as:

$$m_3(C) = \frac{\sum_{A\cap B=C} m_1(A)m_2(B)}{\sum_{A\cap B=\emptyset} m_1(A)m_2(B)} \quad (3.4.3)$$

3.4.1.2 Granger causality test

Granger causality test was proposed by **Granger (1969)** arguing that it is not necessary to have a correlation between two financial time series, rather prior values of a financial time series could predict the future value of another time series. A time series X is said to Granger causes time series Y if lagged values of time series X (with the lagged values of Y also included) help to forecast the future value of time series Y through the statistical hypothesis tests (i.e. t-test or F-test). Let, X_t and Y_t are two stationary time series then the Granger causality is estimated by the following vector auto-regression (VAR) model:

$$Y_t = \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \cdots + \alpha_p Y_{t-p} + \beta_1 X_{t-1} + \beta_2 X_{t-2} + \cdots + \beta_q X_{t-q} + \varepsilon_t \quad (3.4.4)$$

here, $\alpha_1, \alpha_2, \ldots \alpha_p, \beta_1, \beta_2, \ldots \beta_q$ are coefficients of the model, p and q are the length of lags and ε_t is error series. Then, testing the hypothesis:

$$\left.\begin{array}{l} H_0: \beta_1 = \beta_2 = \cdots = \beta_q = 0, \text{i.e. } X_t \text{ does not granger cause } Y_t \\ H_1: \text{Not } H_0 \end{array}\right\} \quad (3.4.5)$$

The existence of causality is tested by following F-test statistics based on residual sum of squares (RSS):

$$F = \frac{\big(RSS(p) - RSS(p,q)\big)(T-p-q-1)}{RSS(p,q)*q} \quad (3.4.6)$$

If the time series X_t and Y_t are co-integrated, then there must be Granger causality between them but the converse is not true. For co-integrated series, the VECM (vector error correction model) is suggested for testing granger causality rather than the VAR model. VECM (vector error correction model) is the modified form of VAR which allows short-run dynamics to converge to their co-integrating relation. Granger causality estimated by VECM model follows the given relation:

$$Y_t = \lambda e_{t-1} + \alpha_1 Y_{t-1} + \alpha_2 Y_{t-2} + \cdots + \alpha_p Y_{t-p} + \beta_1 X_{t-1} + \beta_2 X_{t-2} + \cdots + \beta_q X_{t-q} + \varepsilon_t \quad (3.4.7)$$

here, $e_{t-1} = Y_{t-1} - \theta - \varphi * X_{t-1}$, $Y_t = \theta + \varphi * X_t$ is a co-integration relationship between the variable and λ is error correction parameter.

3.4.2 Proposed model

Model [4] is developed in four sections and Figure 3.4.1 depicts its overview. In section 3.4.2.1, D-S theory for ranking sectors is presented. Section 3.4.2.2 is about determining causal relationship among strong sectors using Granger causal network. Sector based portfolio construction procedure is proposed in section 3.4.2.3 and portfolio optimization using deep recurrent neural network is done in section 3.4.2.4.

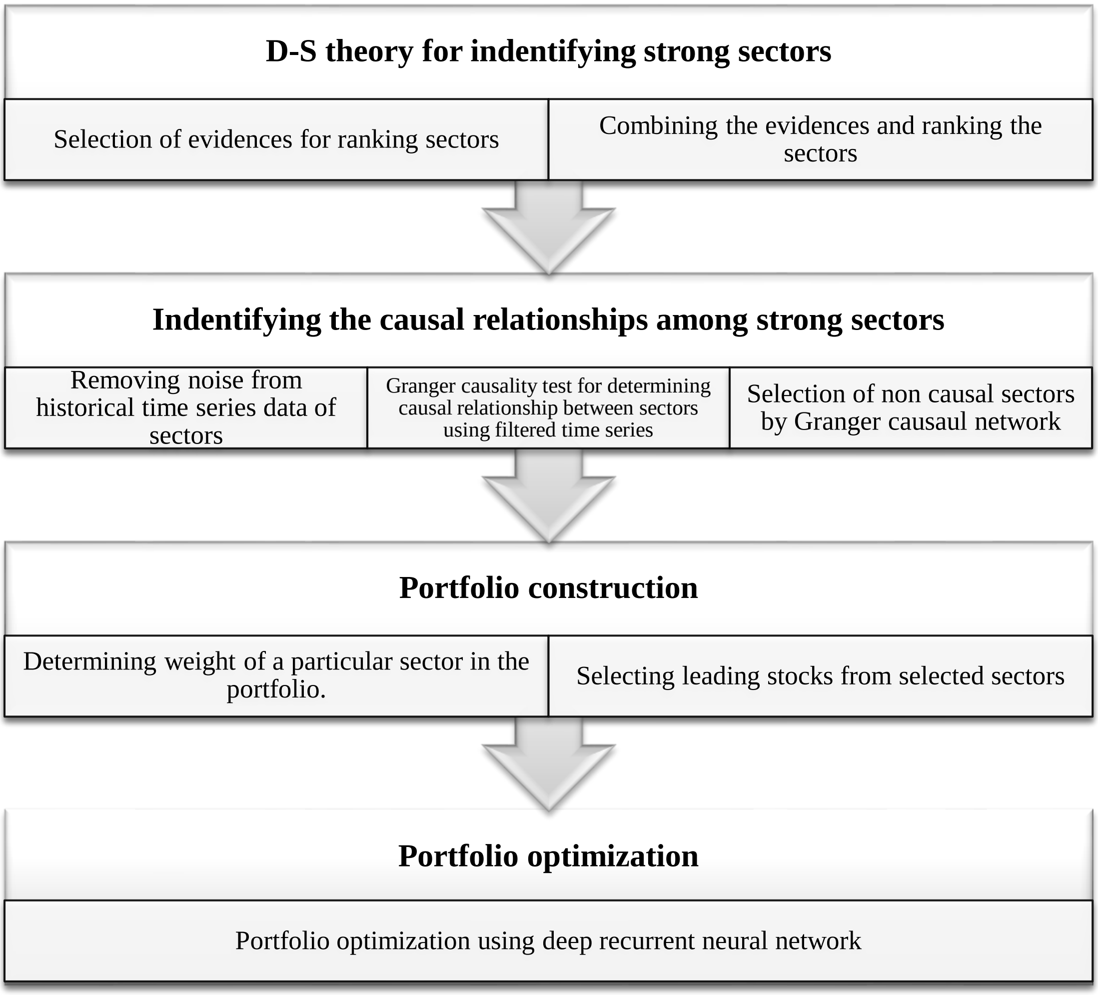


Figure 3.4.1: Overview of Model [4].

3.4.2.1 D-S theory for ranking sectors

This section is the application of D-S theory for indentifying strong sectors which includes section of evidences and combination of evidences.

A. *Selection of evidences for ranking the sectors*

The D-S theory is used to rank sectors in which some technical indicators signifying sector’s performance based on past trends have been be utilized as evidences. Among various kinds of technical indicators, breadth indicators are one of the most extensively used indicators for measuring health of a sector. Breadth indicators quantify the number of advancing stocks relative to the declining stocks belonging to a particular sector in order to gauge the amount of participation in the sector. Breadth indicators help in confirming the trend of a particular sector. A large number of advancing stocks in a sector is a signal of bullish market sentiment and confirms the uptrend of the sector whereas a large number of declining stocks signs bearish sentiment aligning with the downtrend of the sector. To analyze the performance of a sector, we employed three breadth indicators and defined a sector's relative strength index, which were treated as evidences. The four evidences and their importance in the sector analysis are discussed below:

i. **Average directional index (ADX):** ADX is a technical indicator that determines the strength of a trend. A trend can be upward or downward in nature. ADX above 25 indicates a strong trend, while ADX below 25 indicates a weak trend. The trend's direction is determined by the negative direction indicator (-DI) and the positive directional indicator (+DI). If the (+DI) line crosses above the (-DI) line and the ADX is above 25, the stock is in a strong upward trend; if the (-DI) line crosses above the (+DI) line and the ADX is above 25, the stock is in a strong downward trend. For this study, we determine the number of stocks in a sector for which the +DI line is above –DI line and ADX is above 25. To calculate ADX, 30-days time frame is used. This quantifies the percentage of stocks in a specific sector that were in a strong uptrend over a given time period.

ii. **Relative strength indicator (RSI):** RSI is a momentum indicator that gauges how quickly price movements change. It oscillates between 0 and 100, indicating bullish and bearish price momentum. When the RSI is expected to stay between 40 and 90, it suggests bullish or uptrend momentum, but when it is likely to stay between 10 and 40, it signals downtrend momentum. The standard is to use 14 periods to calculate the initial RSI value, we use the same. By filtering the stocks whose RSI is above the average line, we determined how many stocks in a sector were in bullish momentum at different time periods over past few years.

iii. **Simple moving average (SMA):** Simple moving average is used as a technical indicator that estimates the average price (usually the closing price) of a stock over a specific period. It is applicable for determining whether a stock continues its trend or not. For example, if the actual stock prices tend to remain above the simple moving average price of the previous 50 days then the stock has a better chance of staying in an uptrend. We used it as a breadth indicator in this work to see how many stocks in a given sector traded above the 200 days simple moving average.

iv. **Sector's relative strength (SRS):** We defined SRS indicator to evaluate the strength of a given sector in comparison to other sectors. If a specific sector outperforms n other sectors at a given period, then the sector's relative strength is assumed to be n. It assesses the sector's dominance in the market.

To ensure independency between the evidences as required to apply the D-S theory, the evidences are examined for different time frames.

B. Combining evidences and ranking sectors

The frame of discernment (Θ) is defined considering three hypotheses: "*sector is weak*" (WS), "*sector is medium or average*" (MS) and "*sector is strong*" (SS). Hence, $(\Theta) = \{WS, MS, SS\}$. Based on the collected four evidences for a sector, the basic probability assignment (BPA) is done to each evidence which is a belief value between [0, 1]. BPA value equal and less than 0.4 is considered as belief towards the hypothesis "*sector is weak*", BPA value lying between 0.4 to 0.6 is supposed to advocates the hypothesis "*sector is medium or average*", and BPA value equal and greater than 0.6 is supposed to support the hypothesis "*sector is strong*". All four evidences supporting different hypotheses are combined using Dempster's rule of combination given in equation (3.4.3). After combining the evidences, sectors are ranked based on the belief value corresponding the hypothesis "*sector is strong*". Finally the sectors having more than 80% belief value of being strong are selected.

3.4.2.2 Granger causal network for finding causal relationships among strong sectors

Proceeding with selected strong sectors identified in the previous section, this section discovers relationships among them in order to achieve diversification. The relationship between the price movements of the strong sectors can be determined by using historical daily closing prices. The denoising of required time series data (i.e. historical closing prices) is done first for further analysis.

A. Removing noise from real time series data

Financial time series obtained in raw form often contain considerable noise. Such noise could be the consequence of the highly volatile nature of the market or unexpected events. The noise entangled with the clean data distorts the pattern or all over trend which can mislead the analysis of time series **Street and Lawley (2014)**. Here, we employ Kalman filter to extract the dynamic pattern and reduce the noise from the collected time series data of different sectors. Kalman filter is based on Bayesian linear regression that uses a series of measurements over time having noise and inaccuracies to estimate the hidden state of a variable **(Rajan and Mathew, 2012)**. It is a recursive estimator which assumes the true state of the variable at time t can be estimated by the previous state at time $t-1$ as:

$$x_t = F_t x_{t-1} + B_t u_{t-1} + w_{t-1} \tag{3.4.8}$$

here, F_t is the state transition matrix applied to the previous state x_{t-1}, B_t is control-input matrix applied to the control vector u_{t-1} and w_{t-1} is the process noise which is drawn from zero-mean normal distribution with covariance Q i.e. $w_{t-1} \sim N(0, Q)$. At time t, the measurement z_t of true state x_t is calculated as:

$$z_t = H_t x_t + v_t$$

here, H_t is the observation matrix which maps true state to the measurement and v_t is the measurement noise assumed as zero mean normal distribution with covariance R i.e. $w_{t-1} \sim N(0, Q)$. Kalman filter estimates x_t at time t, provided the initial state x_0, series of measurements $z_1, z_2 \dots z_t$, and other information about F, B, H, Q and R as described in the system. In this work, Kalman filter is simulated for every time series providing initial state same as in the real data.

B. Granger Causality test for determining causal relationship between sectors

After denoising the data, Granger causality test is conducted for each pair of sectors to indentify the causal relation between each sector pair.

Initially, the first difference of all time series is taken to transform them into stationary time series. The next task is to determine the suitable lag length by using the VAR (vector auto regression) lag length selection model. The lag length is determined according to sequential modified LR test (LR), final prediction error (FPE), akaike information criterion (AIC) and Schwarz information criterion (SC). Then, the pairwise co-integration test is performed for each sector-pair. After the pairwise co-integration test, VECM (vector error correction model) Granger causality test is performed for co-integrated pairs and VAR Granger causality test for other pairs. Then a Granger causality network is prepared showing the dependencies of sectors on each other. In the network, each vertex represents a sector and a directed edge from sector A to another sector B interpreted as A Granger causes B. Finally, we sort a sub-network of the obtained Granger causal network having the least causal relations or no causal relations and also considering the ranking order of sectors too. The component of the sub network will be the strong as well as independent sectors of economy.

3.4.2.3 Portfolio construction

After selecting the diverse and strong sectors for investment, the next task is to construct a portfolio by selecting leading stocks from these sectors. Suppose, a portfolio of n number of stocks has to be constructed. The probability belief value of the hypothesis "*sector is strong*" is used to establish the weightage of each sector in the portfolio, or we can say to determine how many stocks out of n must be taken from a sector. The ranking wise portion of the sector in the portfolio is obtained by dividing the final mass $m(SS)$ of a sector by the total of the final masses of the hypothesis (SS) of all selected sectors and approximating the division to an integer.

Now, the next problem is to select the leading stocks of a particular sector. For this, the top enlisted companies in the sector are analyzed based on their performances in the last financial years. The companies with the highest profit margins in each category are chosen to build the portfolio.

3.4.2.4 Portfolio optimization using deep recurrent neural network

In this model, a framework for optimizing the Sharpe ratio using gradient ascent is designed to obtain optimal weights for the portfolio's assets. The standard formula of Sharpe ratio is defined as:

$$SR = \frac{E(r_p)-r_f}{Std(r_p)} \tag{3.4.9}$$

here, $E(r_p)$ and $Std(r_p)$ are the estimates of the mean and standard deviation of portfolio returns and r_f risk free rate. Specifically, for a investment period of $t = \{1,\cdots,T\}$, we can maximize the following objective function:

$$SR_T = \frac{E(r_{p,t})-r_f}{Std(r_{p,t})} = \frac{\frac{1}{T}\sum_{t=1}^{T} r_{p,t}-r_f}{\sqrt{E\left(r_{p,t}^2\right)-\left(E(r_{p,t})\right)^2}} \tag{3.4.10}$$

here, $r_{p,t}$ is realized portfolio return over n assets at time t denoted as:

$$r_{p,t} = \sum_{i=1}^{n} w_{i,t-1} * r_{i*t} \tag{3.4.11}$$

here, r_{i*t} is the return of asset i. The allocation ratio (position) of asset i is represented as $w_{i,t}, t \in [0,1]$ and $\sum_{i=1}^{n} w_{i,t} = 1$. In our approach, a neural network f with parameters θ is adopted to model $w_{i,t}$ for constructed portfolio:

$$w_{i,t} = f\left(\frac{\theta}{x_t}\right)$$

here, x_t represents the initial day market information and we bypass the next day step by linking the inputs with positions to maximize the Sharpe over trading period T, namely SR_T . However, portfolio imposes constraints that require weights to be positive and summed to one, softmax outputs is used to fulfill these requirements:

$$w_{i,t} = \frac{\exp(\tilde{w}_{i,t})}{\sum_j^n \exp(\tilde{w}_{j,t})} \tag{3.4.12}$$

here, $\tilde{w}_{i,t}$ are the raw weights. Such a framework can be optimized using unconstrained optimisation methods. Particularly, we use gradient ascent to maximize the Sharpe ratio. The gradient of SR_T with respect to parameters θ is readily calculable, with an excellent derivation presented in **Molina (2016)**. Once $\frac{\partial(SR_T)}{\partial\theta}$ is obtained, we can repeatedly compute this value from training data and update the parameters by using gradient ascent:

$$\theta_{new} \coloneqq \theta_{old} + \alpha * \frac{\partial(SR_T)}{\partial\theta} \tag{3.4.13}$$

here, α is the learning rate and the process can be repeated for many epochs until the convergence of Sharpe ratio or the optimization of validation performance is achieved.

We used deep recurrent neural network for optimization process. The neural network comprises three types of layers; input layer, hidden layers or recurrent neural layers, and output layers. Suppose, the portfolio is made up of n assets and each asset is denoted as A_i. A single input is prepared by concatenating information from all assets. For example, the input features of one asset can be its past prices and returns with a dimension of $(k, 2)$ here, k represents the look back window. By stacking features across all assets, the dimension of the resulting input would be $(k, 2 \times n)$. Then this input is fed to the network and expect non-linear features are extracted. The data is further processed in hidden layers which are simple recurrent neural layers and weights are initialized for each node of the neural network. The output layer consists of number of nodes equal to number of assets in portfolio with SOFTMAX activation function which naturally satisfies the constraints of assets weights in the portfolio to be positive and sum to one. The weights obtained by the output layer calculate the value of the objective function. Adam optimizer is used to updates the parameters of the model by back-propagating the errors after each epoch. The process is repeated for given numbers of epochs or till the convergence of the optimization function. The structure of implemented neural network is displayed in Figure 3.4.2.

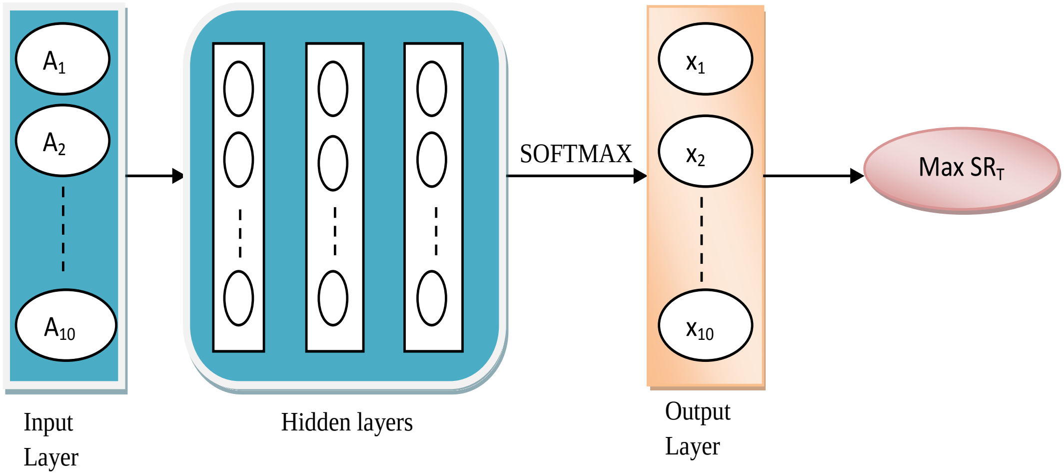


Figure 3.4.2: Overview of deep recurrent neural network used for optimization

3.5 Model [5]: Stock portfolio selection hybridizing fuzzy base-criterion method and evidence theory in triangular fuzzy environment

Investing in the stock market has grown in popularity worldwide over the last few decades. However, in India, only 2% of the population invests in the stock market, whereas in the United States the number is more than 50% of the population **(Sand and Mankotia, 2018)**. Due to the lack of basic knowledge and risky nature of stock market Indians are reluctant to invest their money in stocks. The stock market is influenced by a variety of direct and indirect causes, making it opaque and highly unpredictable. As a result, before investing in a stock, investors are advised to thoroughly study it. In today's global market, analyzing massive amounts of data to make an investment decision is quite challenging task. Domain experts, including successful and experienced investors, examine market-listed companies on a variety of factors, including revenue, risk, growth, solvency, debt, cash flow, valuation and so on, in order to make smart decisions and minimize the risk of loss **(Nti *et al.*, 2020)**. On the other hand, novice investors (common man) having limited knowledge of a company's financials is primarily concerned with two factors: return and risk.

Thus, there is a need to frame the problem of stock selection simultaneously considering the perspectives of a domain expert and a beginner investor which helps to identify the importance of their respective decisions in selection process. In this model, we presented a group MCDM approach of stock selection with two decision makers: a domain expert and a novice investor. The decision maker's opinions and recommendations could be hazy and ambiguous. Therefore, the model is developed using in triangular fuzzy numbers (TFNs) that increases the quality of the final selection. Firstly, fuzzy delphi method (FDM) is presented to identified most effective factors from plenty of factors of stock analyses for both decision makers. BCM in triangular fuzzy environment is developed to determine the weights of decision making attributes. Then, D-S evidence theory is employed for analyzing and ranking the companies based on historical data. Some limitations of **Thakur *et al.* (2018)** work on D-S theory of stocks, (such as giving equal weight to all selection criteria, ranking companies solely based on stock market experts' perspectives, and assigning belief values only for one preposition from the frame of discernment for all factors) are also addressed in the proposed model. On the basis of the historic data, evidences are collected for the frame of discernment consisting three prepositions whether the company is *"good for investment"*, *"average for investment"* or *"bad for investment"*. Then evidences collected for different factors in support of all three prepositions are combined. At last, final evaluation of the companies is done by combining the results obtained by each decision maker's assessment. At last, the companies are ranked in the decreasing order of the belief value supporting the preposition *"good for investment"*. The portfolio of top securities obtained by ranking is constructed. The optimal ratio allocation to the securities is done by using long-short term memory (LSTM) recurrent neural network for maximum Sharpe ratio.

3.5.1 Preliminaries

This section describes triangular fuzzy numbers (TFNs) and their basic operations.

3.5.1.1 Triangular fuzzy numbers (TFNs)

A fuzzy number $\tilde{a}$ on A is defined as a TFN if its membership function $\mu_{\tilde{a}}(x): A \to [0,1]$ is equal to

$$\mu_{\tilde{a}}(x) = \begin{cases} 0, & x < l \\ \frac{x-l}{m-l}, & l \leq x \leq m \\ \frac{u-x}{u-m} & m \leq x \leq u \\ 0, & x > u \end{cases}$$

here, l, m and u are respectively unfolds the lower, middle and upper values of the support of $\tilde{a}$, all of which are crisp numbers $(-\infty < l \leq m \leq u < \infty)$. A TFN can be shown as a triplet (l, m, u).

3.5.1.2 Arithmetic operations on TFNs

Let $T_1 = (l_1, m_1, u_1)$.and $T_2 = (l_2, m_2, u_2)$ are TFNs. The basic mathematical operations on T_1 and T_2 are as follows:

i. $T_1 + T_2 = (l_1 + l_2, m_1 + m_2, u_1 + u_2)$

ii. $T_1 - T_2 = (l_1 - u_2, m_1 - m_2, u_1 - l_2)$

iii. $T_1 * T_2 = (l_1 * l_2, m_1 * m_2, u_1 * u_2)$

iv. $\frac{T_1}{T_2} = (l_1/u_2, m_1/m_2, u_1/l_2)$

v. $T_1{}^{-1} = \langle\left(\frac{1}{u_1}, \frac{1}{l_1}, \frac{1}{l_1}\right); \alpha_{\tilde{A}_1}, \beta_{\tilde{A}_1}, \gamma_{\tilde{A}_1}\rangle$

vi. $\lambda T_1 = \begin{cases} (\lambda l_1,\ \lambda m_1,\ \lambda u_1) & if \lambda > 0 \\ (\lambda u_1,\ \lambda m_1,\ \lambda l_1) & if \lambda < 0 \end{cases}$

vii. $\frac{T_1}{\lambda} = \begin{cases} \left(\frac{l_1}{\lambda}, \frac{m_1}{\lambda}, \frac{u_1}{\lambda}\right) if \lambda > 0 \\ \left(\frac{u_1}{\lambda}, \frac{l_1}{\lambda}, \frac{m_1}{\lambda}\right) if \lambda < 0 \end{cases}$

3.5.2 Proposed model

Model [5] is divided into four sections, as shown in Figure 3.5.1. In section 3.5.2.1, fuzzy delphi method for identifying critical criteria for both decision makers is given. Section 3.5.2.2 is FBCM for weight assessment of criteria. D-S theory of giving ranking to stocks is presented in section 3.5.2.3 and portfolio optimization using deep recurrent neural network is done in section 3.5.2.4.

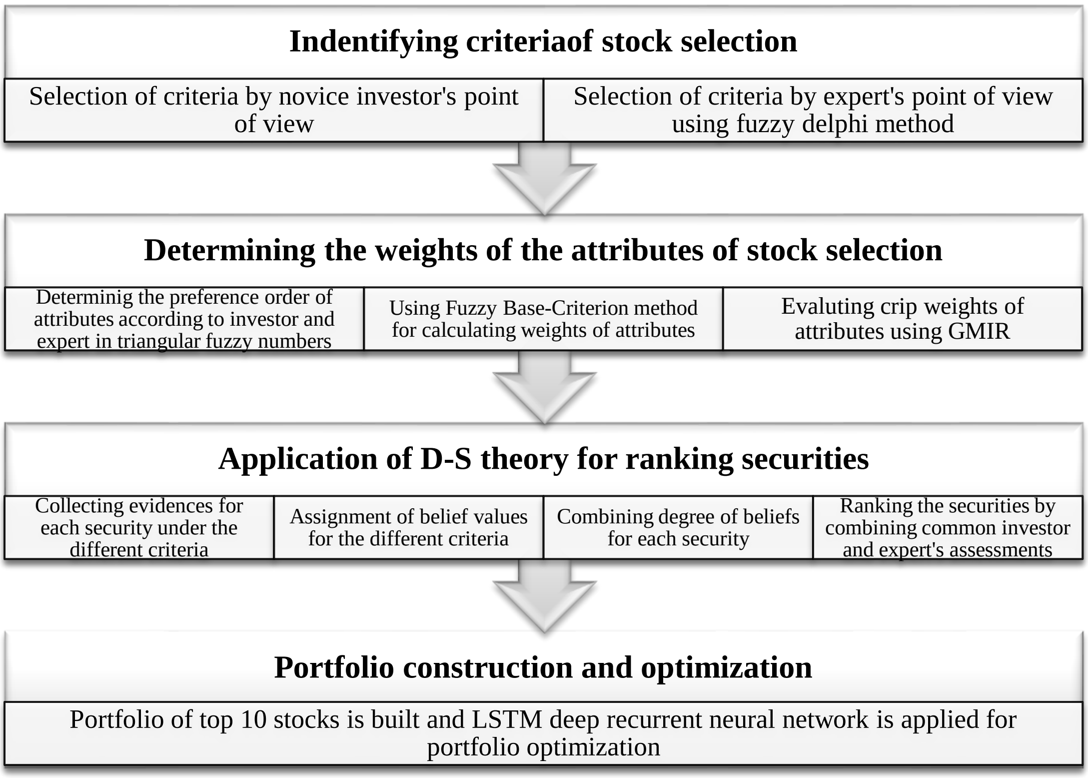


Figure 3.5.1: Overview of Model [5].

3.5.2.1 Fuzzy delphi method (FDM) for identifying criteria of selection

FDM was proposed by **Ishikawa *et al.* (1993)**, and it was derived from the traditional delphi technique and fuzzy set theory. **Noorderhaben (1995)** indicated that applying FDM to group decision can solve the fuzziness of common understanding of expert opinions. As for the selection of fuzzy membership functions, previous researches were usually based on TFNs, TrFNs and Gaussian fuzzy number. This study applied the triangular membership functions and the fuzzy theory to solving the group decision. FDM is used for the screening of factors of the first stage. The fuzziness of common understanding of experts could be solved by using the fuzzy theory, and evaluated on a more flexible scale. Thus, more objective evaluation factors could be screened through FDM involving following steps:

Step 1. Collect opinions of decision group: Find the evaluation score of each factor's significance given by each expert by using linguistic variables in questionnaires.

Step 2. Set up TFNs: Calculate the evaluation value of TFN of each factor given by experts, find out the significance TFN of the factor. In this model we have used geometric mean to find out the common understanding of group decision. The computing formula is illustrated as follows:

Assuming the evaluation value of the significance of j^{th} factor given by i^{th} of n experts is $E_{ij} = (l_{ij}, m_{ij,} u_{ij}), i = 1,2, \dots, n , j = 1,2, \dots m$, then the fuzzy weight $\tilde{E}_j$ of j^{th} alternative is $\tilde{E}_j = (l_j, m_{j,} u_j), j = 1,2,3 \dots m$ here,

$$l_j = \min_i (l_{ij}), \qquad m_j = \sqrt[n]{\prod_{i=1}^{n} m_{ij,}} \qquad , u_j = \max_i (u_{ij})$$

Step 3. Defuzzification: To derive the statistically unbiased effect and reduce the impact of extreme values geometric mean is taken to defuzzify the fuzzy weight $\tilde{E}_j$ of each factor to definite value S_j as:

$$S_j = \sqrt[3]{(l_j * m_j * u_j)}, j = 1,2, \dots m \qquad (3.5.1)$$

Step 4. Screen evaluation indices: Finally proper factors can be screened out from numerous factors by setting the threshold α. The principle of screening is as follows:

If $S_j \geq \alpha, j^{th}$ factor is the evaluation index.

If $S_j < \alpha$, then delete j^{th} factor.

3.5.2.2 Fuzzy base-criterion method (FBCM) of criteria weight determination

In this section, BCM method using TFNs is moved for determining optimal weights of the criteria. The steps of FBCM are as follows:

Step 1. Selected decision criteria of the ultimate objective of this study are termed as $\{C_1, C_2, \dots\ C_n\}$.

Step 2. One of the criteria has to be selected as the base-criteria. Then determine the pairwise comparison between the base-criterion and other criteria as per the decision maker opinion using the triangular fuzzy scale given in Table 3.5.1.

Table 3.5.1: TFNs for pairwise comparison

Linguistic term	TFNs
Equally important	(1, 1, 1)
Moderate important	(1, 1, 3/2)
Strong important	(1, 3/2, 2)
Very strong important	(3/2, 2, 5/2)
Extreme important	(2, 5/2, 3)

Step 3. The optimal fuzzy weights of criteria have to satisfy the condition $\frac{\tilde{w}_B}{\tilde{w}_j} = \tilde{a}_{Bj}$ for all j, here, $\tilde{w}_B$ is the weight of base-criterion and $\tilde{a}_{Bj}$ shows the triangular fuzzy degree of preference of the base-criterion over the j criterion. The maximum absolute odds $\left|\frac{\tilde{w}_B}{\tilde{w}_j} - \tilde{a}_{Bj}\right|$ should be minimized for all j. As the weight of the element is non-negative, so we can obtain the optimal fuzzy weights on solving following optimization problem.

$\min \max_j \left|\frac{\widetilde{w}_B}{\widetilde{w}_j} - \tilde{a}_{Bj}\right|$, such that

$$\begin{cases} \sum_{j=1}^{n} R(\widetilde{w}_j) = 1 \\ l_j^w \leq m_j^w \leq u_j^w \\ l_j^w \geq 0 \; for \; all \; j \end{cases} \tag{3.5.2}$$

In the terms of TFNs and $\tilde{\xi} = (k^*, k^*, k^*)$, the equation (3.5.2) can be expressed as follows:

$\min \tilde{\xi}$, such that

$$\begin{cases} \left|\frac{(l_B^w, m_B^w, u_B^w)}{(l_j^w, m_j^w, u_j^w)} - (l_{Bj}, m_{Bj,} u_{Bj})\right| \leq (k^*, k^*, k^*) \\ \sum_{j=1}^{n} R(\widetilde{w}_j) = 1 \\ l_j^w \leq m_j^w \leq u_j^w \\ l_j^w \geq 0 \; for \; all \; j \end{cases} \tag{3.5.3}$$

Step 4. After solving the above equation 3.5.3, the optimal triangular fuzzy weights are obtained which are converted to the optimal crisp weights by using the graded mean integration representation (GMIR) method defined as follows:

Definition: GMIR $\left(R(\tilde{A})\right)$ of a triangular fuzzy number $\tilde{A}$ represents the ranking of triangular fuzzy number. Let, $\tilde{A}_i = (l_i, m_i, u_i)$, then

$$R(\tilde{A}_i) = \frac{l_i + 4m_i + u_i}{6} \tag{3.5.4}$$

3.5.2.3 D-S theory for ranking stocks

The hierarchical framework of stock selection is depicted in Figure 3.5.2, here, D_i represents the decision makers level, for n number of attributes $a_i^{1\ldots n}$ represents the attribute level of i^{th} decision maker, $w_i^{1\ldots n}$ represents the weights of the attributes of i^{th} decision maker and $m(a_i^{1\ldots n})$ represents the basic probability of different attributes. $S(a_i^{1\ldots n})$ is evaluation of the attributes. The step by step procedure of D-S theory applied for this study is as follows:

Step1. The frame of discernment (FOD) denoted by Θ, is a finite set of mutually exclusive and exhaustive propositions. The set of the subsets of Θ, i.e. 2^{Θ} include all the events. For each attribute selected for stock analysis, the frame of discernment is considered as $(\Theta) = \{BI, AI, GI\}$ regarding the prepositions about the company is *"bad for investment"*, *"average (mediocre) for investment"* or *"good for investment"*, so the power set is represented as

$$2^{\Theta} = \{\varphi, \{BI\}, \{AI\}, \{GI\}, \{BI, AI\}, \{AI, GI\}, \{BI, GI\}, \{BI, AI, GI\}\}$$

Each subset of 2^{Θ} is called focal element. The focal element having two elements represents the in-between situation of the prepositions. The focal elements $\{BI, GI\}$ and $\{BI, AI, GI\}$ does not give any valid information regarding any attribute used in this study; therefore it can be completely ignored.

Step 2. The belief values for j^{th} attribute of i^{th} decision maker corresponding to the frame of discernment $(\Theta) = \{BI, AI, GI\} = \{f_1, f_2, f_3\}$ are presented by $\{\beta_i^{1,j}, \beta_i^{2,j}, \beta_i^{3,j}\}$. So, the $S(a_i^j)$ can be expressed as:

$$S(a_i^j) = \{(f_1, \beta_i^{1,j}), (f_2, \beta_i^{2,j}), (f_3, \beta_i^{3,j})\} \tag{3.5.5}$$

here, $0 \leq \beta_i^{1,j}, \beta_i^{2,j}, \beta_i^{3,j} \leq 1$, $(\beta_i^{1,j} + \beta_i^{2,j} + \beta_i^{3,j}) \leq 1$. For crisp observations, $(\beta_i^{1,j} + \beta_i^{2,j} + \beta_i^{3,j}) = 1$ which means there is no uncertainty in the belief.

Step 3. BPA of the attributes is obtained by multiplying with their respective weights and $(1 - \sum BPA)$ is considered as the uncertainty or ignorance (λ).

The BPA of the attributes are represented as $m(a_i^{1....n}) = \{m(a_i^{f_k,1....n}), \lambda(a_i^n)\}$

Step 4. Suppose $\{a_i^1, a_i^2\}$ are two attributes of i^{th} decision maker, for combining corresponding evidences the aggregation $\left(P = \frac{1}{K}\right)$ is calculated as follows:

$$P(1 \oplus 2) = \left[1 - \sum_{k=1}^{3}\{\sum_{l=1}^{3} m(a_2^{f_k,1}) * m(a_2^{f_l,2})\}\right]^{-1} \tag{3.5.6}$$

The new BPAs by combining $m(a_i^1)$ and $m(a_i^2)$ are as follows:

$$m\left(a_i^{f_k,(1\oplus 2)}\right) = P(1 \oplus 2) * \{m(a_i^{f_k,1}) * m(a_i^{f_k,2}) + m(a_i^{f_k,1}) * \lambda(a_i^2) +$$
$$m(a_i^{f_k,2}) * \lambda(a_i^1)\}, \ \ k = 1,2,3$$

$$\lambda\left(a_2^{(1\oplus 2)}\right) = P(1 \oplus 2) * \{\lambda(a_2^1) * \lambda(a_2^2)\}$$

The obtained BPAs are combined with 3rd attribute, and in the similar manner new generated BPAs are combined with 4th attribute so on upto n^{th} attribute. After combining all attributes the final mass values for decision makers are obtained as:

$$\{BI\} = \frac{m\left(a_i^{f_1,(1\oplus 2\ldots n)}\right)}{\sum_{k=1}^{3} m\left(a_i^{f_k,(1\oplus 2\ldots n)}\right)} \;;\quad \{AI\} = \frac{m\left(a_i^{f_2,(1\oplus 2\ldots n)}\right)}{\sum_{k=1}^{3} m\left(a_i^{f_k,(1\oplus 2\ldots n)}\right)} \;;\quad \{GI\} = \frac{m\left(a_i^{f_3,(1\oplus 2\ldots n)}\right)}{\sum_{k=1}^{3} m\left(a_i^{f_k,(1\oplus 2\ldots n)}\right)} \tag{3.5.7}$$

Finally assessment of decision makers D_1 and D_2 is combined as follows:

$$U = \alpha D_1 + \beta D_2 \tag{3.5.8}$$

here, α and β represents the weightage of the decision makers in the final decision.

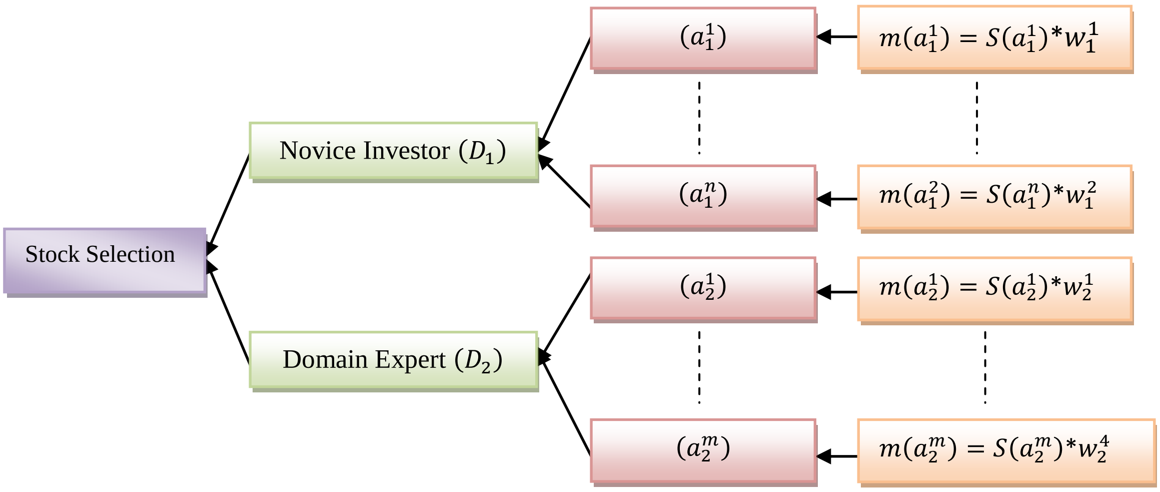


Figure 3.5.2: Hierarchical structure of stock selection process

3.5.2.4 Portfolio construction and optimization

A portfolio is constructed by selecting top ranking companies based on the value of U determined in previous section. We utilized the same algorithm for maximizing Sharpe ratio using neural network as described in section 3.4.2.4 of Model [4], but instead of a simple recurrent neural network, LSTM cells are employed in hidden layers and the rest of the neural network design is kept as shown in Figure 3.4.2.

3.6 Model [6]: A method of intraday stock selection integrating fuzzy TOPSIS and belief divergence measure in evidence theory

Intraday trading refers to trade for quick profits by microstructure implicit in the movement of stock indices **(Loginov *et al.*, 2021)**. This practice of trading gaining popularity as it offers to generate a lot of money, if a well-structured strategy is applied. In a down market, day traders can earn profit by short selling, which is another advantage of intraday. The first and most important step of intraday trading is to select appropriate stocks. Basically in intraday, trader buys and sells a particular stock multiple times a day and earns profit by fast price variations. Therefore, day traders aspires to pick such stock whose price movements offer good number of buying and selling opportunities in live market **(Zhang, 2018)**. Intraday trading is quite riskier as compared to long term investment. Uncertainty of stock market and unexpected market fluctuations can incur great loss. Long-term stock selection is usually done by analyzing past years performances of stocks. Hence adequate amount of data with less ambiguity is available for inspection. While for intraday, some previous days' data is used, which is limited, very uncertain, and highly contradictory **(Sharma and Habib, 2019)**. These intraday considerations necessitate a different stock selection methodology than the existing long-term stock selection algorithms.

The main conditions for intraday trading are that a stock should have ample liquidity, moderate to high volatility, and a large number of group followers. However, in addition to these basic factors, the use of technical indicators is extremely crucial for intraday stock selection. Technical indicators are heuristic or mathematical computations that anticipate future price movements based on the price, volume, or open interest of a stock. These are useful in understanding market sentiments and trends for the next trading day. When all of the fundamental aspects and technical indicators are taken into account, intraday stock selection can be defined as a MCDM problem. Although extensive studies have been done on portfolio selection for long-term investment using various MCDM algorithms but these algorithms do not guarantee optimal stock selection due to the vast differences in the nature of both types of trading strategies. In this research, we attempt to address the requirements and constraints of intraday trading and describe the stock selection process using a hybrid MCDM algorithm.

The proposed methodology is an integration of fuzzy TOPSIS and D-S evidence theory using belief divergence measure. Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) is a very popular MCDM method originally proposed by **Hwang and Yoon (1981)**. Its applications in variety of real-world problem as well as in stock selection have been discussed **(Behzadian *et al.*, 2012, Ece and Uludag, 2017, Nguyen *et al.*, 2020, Yaakob and Gegov, 2016, Zou *et al.*, 2011)**. In intraday stock selection, some criteria values are determined in linguistic term which has some fuzziness as well. To deal with it triangular fuzzy numbers are used in this study for better comparison among alternatives. The principle advantage of D-S theory is that it applies a degree of confidence known as mass functions to all simple and composition classes while also accounting for degree of ignorance of the information. It delivers a reliable outcome with less number of evidences. However, in the case of extremely conflicting evidences in intraday stock selection, fundamental D-S theory may produce counter-intuitive conclusions. The proposed method employs the belief divergence measure to address this issue. Belief divergence measure, which is extensively used in uncertain information processing, is a measure of how one probability distribution differs from another **(Fei and Deng, 2016, Song and Deng, 2019)**. Based on the assessment of belief divergence measure, weights are assigned to the evidences. The belief divergence measure has been proved to be exceptionally effective in dealing with highly uncertain and conflicting data **(Wang *et al.*, 2021)**.

3.6.1 Preliminaries

TFNs and their basic operations are covered in Model [5]'s preliminaries section 3.5.1, and the brief introduction of D-S theory is given in Model [4]'s preliminaries section 3.4.1. Remaining essentials used to develop this model are discussed below:

3.6.1.1 Fuzzy TOPSIS

Fuzzy TOPSIS method is developed in the following steps:

Step 1. Identify the potential alternatives $\{A_i : i = 1,2, \ldots n\}$ and critical criteria $\{C_j : j = 1,2, \ldots m\}$. Assign weights to the criteria $\{w_j : j = 1,2, \ldots m\}$.

Step 2. Determine the initial fuzzy decision matrix $\left\{D_{ij} = \left[x_{ij}\right]_{n*m}\right\}$

$$D_{ij} = \begin{bmatrix} x_{11} & \cdots & x_{1j} \\ \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} \end{bmatrix} \tag{3.6.1}$$

here, $x_{ij} = \left(l_{ij}, m_{ij}, u_{ij}\right)$ is a triangular fuzzy number representing evaluation of i^{th} alternative corresponding j^{th} criteria.

Step 3. Determine weighted normalized fuzzy decision matrix

The weighted normalized fuzzy decision matrix denoted by $\{E_{ij}\}$ is calculated as:

$$E_{ij} = \left[v_{ij}\right]_{n*m} = \left[w_j * \tilde{X}_{ij}\right]_{n*m} \tag{3.6.2}$$

here, $\tilde{X}_{ij} = \left(\frac{l_{ij}}{u_j{}^+}, \frac{m_{ij}}{u_j{}^+}, \frac{u_{ij}}{u_j{}^+}\right)$ is the normalized evaluation value obtained on dividing the components by $u_j{}^+ = max_i\left\{u_{ij} : i = 1,2 \dots . n\right\}$

Step 4. Determine fuzzy positive ideal solution (FPIS) and fuzzy negative ideal solution (FNIS)

FPIS (P^+) and FNIS (P^-) are defined as aspiring and worst value of the criteria and denoted as:

$$P^+ = \{v_1^*, v_2^*, \dots v_m^*\} \tag{3.6.3a}$$

$$P^- = \{v_1^-, v_2^-, \dots v_m^-\} \tag{3.6.3b}$$

Step 5. Determine the distance of each alternative from P^+ and P^-

The Euclidean distances d_i^+ and d_i^- from P^+ and P^- of i^{th} alternative is calculated as:

$$d_i^+ = \sqrt{\left(\frac{1}{3m}\right) \sum_{j=1}^{m} \{(l_i - l_i^*)^2 + (m_i - m_i^*)^2 + (u_i - u_i^*)^2\}} \quad i = 1,2, \dots , n \tag{3.6.4a}$$

$$d_i^- = \sqrt{\left(\frac{1}{3m}\right) \sum_{j=1}^{m} \{(l_i - l_i^-)^2 + (m_i - m_i^-)^2 + (u_i - u_i^-)^2\}} \quad i = 1,2, \dots , n \tag{3.6.4b}$$

Step 6. Calculate closeness coefficients and rank the alternatives

The closeness coefficient of i^{th} alternative to ideal solution is calculated as:

$$C_i = \frac{d_i^-}{d_i^- + d_i^+} \tag{3.6.5}$$

Finally, the alternatives are ranked in the decreasing order of closeness coefficient.

3.6.1.2 Belief divergence measure in evidence theory

Suppose, m_1 and m_2 are the mass functions of two evidences A and B respectively, the belief divergence measure between m_1 and m_2 is defined as follows:

$$D(m_1, m_2) = \sum_i \frac{1}{2^{|F_i|}} m_1(F_i) \ln\left(\frac{m_1(F_i)}{m_2(F_i)}\right) \tag{3.6.6}$$

here, $|F_i|$ is the cardinality of the focal element F_i. In addition, in the calculation of divergence measure, to avoid the denominator to be zero, a very small value such as 10^{-8} is used to replace zero.

The symmetric belief divergence measure between m_1 and m_2 is calculated as:

$$D_{12} = D_{21} = \frac{D(m_1, m_2) + D(m_2, m_1)}{2} \tag{3.6.7}$$

3.6.2 Proposed model

The step wise procedure of Model [6] of ranking alternative is as follows:

Step 1. Collecting the preliminary information to set up mathematical structure of decision making problem

1. Identify the potential alternatives $\{A_i: i = 1,2, \ldots n\}$ and critical criteria$\{C_j: j = 1,2, \ldots m\}$
2. Determine the initial decision matrix $\left\{D_{ij} = [x_{ij}]_{n*m}\right\}$ here, $x_{ij}'s$ are the crisp evaluation value of i^{th} alternative corresponding j^{th} criteria

$$D_{ij} = \begin{bmatrix} x_{11} & \cdots & x_{1j} \\ \vdots & \ddots & \vdots \\ x_{i1} & \cdots & x_{ij} \end{bmatrix} \tag{3.6.8}$$

Step 2. Transforming the initial decision matrix to fuzzy decision matrix

The fuzzy decision matrix $\left\{E_{ij} = \left[\tilde{X}_{ij}\right]_{n*m}\right\}$ is obtained by converting the entries of initial decision matrix $\{D_{ij}\}$ to TFNs with the help of corresponding linguistic terms given in the Table 3.6.1.

$$E_{ij} = \begin{bmatrix} \tilde{X}_{11} & \cdots & \tilde{X}_{1j} \\ \vdots & \ddots & \vdots \\ \tilde{X}_{i1} & \cdots & \tilde{X}_{ij} \end{bmatrix} = \begin{bmatrix} (l_{11}, m_{11}, n_{11}) & \cdots & (l_{1j}, m_{1j}, n_{1j}) \\ \vdots & \ddots & \vdots \\ (l_{i1}, m_{i1}, n_{i1}) & \cdots & (l_{ij}, m_{ij}, n_{ij}) \end{bmatrix} \quad (3.6.9)$$

Table 3.6.1: Linguistic terms and corresponding TFNs

Linguistic terms	**TFNs**
Extremely Low (EL)	(0.1,0.1,0.1)
Very Low (VL)	(0.1,0.2,0.3)
Low (L)	(0.2,0.3,0.4)
Medium Low (ML)	(0.3,0.4,0.5)
Medium (M)	(0.4,0.5,0.6)
Medium High (MH)	(0.5,0.6,0.7)
High (H)	(0.6,0.7,0.8)
Very High (VH)	(0.7,0.8,0.9)
Extremely High (EH)	(0.8,0.9,0.9)

Step 3. Identify the best and worst TFN value vectors of each criterion.

The best and worst TFN values of each criterion are consulted by the domain experts and presented as vectors (V^{+}) and (V^{-}), respectively.

$$(V^{+}) == \left\{\left(l_j^*, m_j^*, n_j^*\right): \left(l_j^*, m_j^*, n_j^*\right) is\ the\ best/desiable\ TFN\ value\ of\ j^{th}\ criterion\right\}$$

(3.6.10a)

$$(V^{-}) == \left\{\left(l_j^-, m_j^-, n_j^-\right): \left(l_j^-, m_j^-, n_j^-\right) is\ the\ worst/undesiable\ TFN\ value\ of\ j^{th}\ criterion\right\}$$

(3.6.10b)

Step 4. Calculate distance between best (worst) criterion value and initial TFN value

The distance between the best (worst) criterion value and initial TFN value is calculated by the help of Euclidian distance between two TFNs defined as:

$$ED(TFN_1, TFN_2) = \sqrt{\left(\frac{1}{3}\right)\{(l_1 - l_2)^2 + (m_1 - m_2)^2 + (u_1 - u_2)^2\}} \quad (3.6.11)$$

and positive decision matrix (PDM) and negative decision matrix (NDM) are determined as:

$$(PDM) = \begin{bmatrix} ED(\tilde{X}_{11}, (l_1^*, m_1^*, n_1^*)) & \cdots & ED(\tilde{X}_{1j}, (l_j^*, m_j^*, n_j^*)) \\ \vdots & \ddots & \vdots \\ ED(\tilde{X}_{i1}, (l_1^*, m_1^*, n_1^*)) & \cdots & ED(\tilde{X}_{ij}, (l_j^*, m_j^*, n_j^*)) \end{bmatrix} = \left[(l_{ij}^P, m_{ij}^P, n_{ij}^P)\right]_{n*m}$$

(3.6.12a)

$$(NDM) = \begin{bmatrix} ED(\tilde{X}_{11}, (l_1^*, m_1^*, n_1^*)) & \cdots & ED(\tilde{X}_{1j}, (l_j^*, m_j^*, n_j^*)) \\ \vdots & \ddots & \vdots \\ ED(\tilde{X}_{i1}, (l_1^*, m_1^*, n_1^*)) & \cdots & ED(\tilde{X}_{ij}, (l_j^*, m_j^*, n_j^*)) \end{bmatrix} = \left[(l_{ij}^N, m_{ij}^N, n_{ij}^N)\right]_{n*m}$$

(3.6.12b)

Step 5. Define frame of discernment (Θ) and basic probability assignment (BPA)

All criteria are categories into two frames of discernment such as Low (L) and High (H). The basic probabilities supporting the evidences the two events are considered as follows:

High (H) = closeness coefficient of the criterion to the aspired value (CC)

$$= \frac{(l_{ij}^N, m_{ij}^N, n_{ij}^N)}{(l_{ij}^N, m_{ij}^N, n_{ij}^N) + (l_{ij}^P, m_{ij}^P, n_{ij}^P)} \quad (3.6.13a)$$

Low (L) $= 1 - CC$ (3.6.13b)

Step 6. Obtain symmetric belief divergence matrix (SBDM) for each alternative

$$\text{SBDM} = \begin{pmatrix} 0 & D_{12} \ldots\ldots & D_{1k} \\ D_{21} & 0 \ldots\ldots & D_{2k} \\ D_{k1} & D_{k2} \ldots\ldots & 0 \end{pmatrix} \quad (3.6.14)$$

here, D_{1k} is belief divergence between m_1 and m_2 and k is the number of criteria.

Step 7. Calculate the supporting degree of evidences

$$S_i = \frac{1}{\sum_{j=1, j\neq k}^{k} D_{ij}} \quad (3.6.15)$$

Step 8. Calculate weight evidences

$$W_i = \frac{S_i}{\sum_{j=1}^{k} S_i} \quad (3.6.16)$$

Step 9. Calculated weighted BPA of evidences

$$M_i(A) = W_i * m_i(A) \quad (3.6.17)$$

Step 10. Combining evidences

Suppose, the weighted BPAs of first two evidences of A are represented as:

$$M_1 = \{M_1(L), M_1(H), \lambda_1\}$$

$$M_2 = \{M_2(L), M_2(H), \lambda_2\}$$

here, λ_i represents the degree of ignorance. The aggregation $\left(P = \frac{1}{K}\right)$ is calculated as follows:

$$P(1 \oplus 2) = [1 - \{M_1(L) * M_2(L) + M_1(L) * M_2(H) + M_1(H) * M_2(L) + M_1(H) * M_2(H)\}]^{-1}$$

New BPAs generated by combining M_1 and M_2 are as follows:

$$M_{1\oplus 2}(L) = P(1 \oplus 2) * \{M_1(L) * M_2(L) + M_1(L) * \lambda_2 + M_2(L) * \lambda_1\}$$

$$M_{1\oplus 2}(H) = P(1 \oplus 2) * \{M_1(H) * M_2(H) + M_1(H) * \lambda_2 + M_2(H) * \lambda_1\}$$

$$\lambda_{1\oplus 2} = P(1 \oplus 2) * \lambda_1 * \lambda_2$$

The obtained BPAs are combined with 3rd evidence, and in the similar manner upto j^{th}evidence. After combining all evidences the final mass values are obtained as:

$$\{L\} = \frac{M_{1\oplus 2\oplus \ldots j}(L)}{M_{1\oplus 2\oplus \ldots j}(L) + M_{1\oplus 2\oplus \ldots j}(H)} \quad (3.6.18a)$$

$$\{H\} = \frac{M_{1\oplus 2\oplus \ldots j}(H)}{M_{1\oplus 2\oplus \ldots j}(L) + M_{1\oplus 2\oplus \ldots j}(H)} \quad (3.6.18b)$$

Similarly, final belief values for all alternatives are calculated.

Step 11. Ranking alternatives

At last, the alternatives are ranked based on the final belief values obtained in the previous step.

3.7 Model [7]: Single valued triangular neutrosophic MEREC-CoCoSo method for multi-criteria decision making and its application in portfolio construction

Recently, a new ranking method called MEthod based on the Removal Effects of Criteria (MEREC) has been proposed by **Ghorabaee *et al.* (2021).** MEREC is less calculative as well as is capable of achieving relative objective weights of several conflicting criteria. The method is rest on the expulsion impact of each criterion on the collective performance of the alternatives for computing the criterion weighting. Criteria that have a greater impact on performance are given more importance. Causality is the basic idea behind this approach. In this method, a logarithmic function is used to calculate the aggregate performance of the alternatives. The absolute deviation measure is used to identify the effect on expulsion of each criterion. This measure unfolds the difference among the overall performance of the alternatives and its performance after removing the criteria. MEREC is very flexible and gives stable and reliable objective weights of criteria.

The Combined Compromise Solution (CoCoSo) method developed by **Yazdani *et al.* (2018)** which is new and having a unique structure among several MCDM methods. It is highly capable of working with incomplete and uncertain data. Foundation of the CoCoSo method is based on the idea of SAW, MEW and WASPAS methods. Currently, the utility of CoCoSo method is increasing in many fields **(Peng *et al.*, 2020, Wen *et al.*, 2019, Yazdani *et al.*, 2019)**.

In the proposed work, we offer a method for ranking alternatives in a SVTN environment by combining the MEREC and CoCoSo methods. The two approaches are integrated to produce a single step wise methodology in which criteria weights are generated using MEREC methodology and the process ranking is completed using the CoCoSo method. In the MEREC method, a parabolic measure is employed to calculate the overall performance of alternatives, and to demonstrate the applicability, a simple decision matrix is studied in a fuzzy environment. We devised the method for selecting stocks for portfolio construction. Uncertainty and predictability are common characteristics of qualitative assessments in stock market. Therefore, the proposed hybrid model is developed in SVTN environment. This hybrid approach

(STVN-MEREC-CoCoSo) for solving stock selection problem has been discussed for the first time.

The following are the main motivations for developing the proposed model for stock selection:

i. Despite uncertainties, it can make complex stock selection processes in multi-dimensional decision analysis systems much easier and more efficient.

ii. During the selection process, MEREC ensures that a selection criterion has a higher weight when its removal has a greater impact on the aggregate performance of alternatives, hence eliminating the reliance on external decision makers for examining criteria.

iii. The CoCoSo approach provides a comprehensive solution by pooling the results of various methodologies, making the final judgment more resilient as it includes randomness of stock market data.

3.7.1 Preliminaries

SVTNs, their operations, score and accuracy functions, and comparison of two SVTNs are defined in preliminaries section 3.3.1 in Model [3]. The averaging operators of SVTNs and traditional working procedure of MEREC and CoCoSo method are described below:

3.7.1.1 Single valued triangular neutrosophic averaging operators

Let $\tilde{A}_j = \left\{\langle(a_j, b_j, c_j); \alpha_{\tilde{A}_j}, \beta_{\tilde{A}_j}, \gamma_{\tilde{A}_j}\rangle\right\}, (j = 1,2,3 \ldots . n)$ be the set of SVTNs and $W_j = (w_1, w_2 \ldots w_n)^T$, $w_j \geq 0$, $\sum_{j=1}^{n} w_j = 1$ be the weight vector associated with $\tilde{A}_j$.

Single valued triangular neutrosophic weight averaging operator (SVTNWAO) is defined as:

SVTNWAO $(\tilde{A}_1, \tilde{A}_2 \ldots \tilde{A}_n)$

$$= \langle g^{-1}\left(\sum_{j=1}^{n} w_j g(a_j)\right), g^{-1}\left(\sum_{j=1}^{n} w_j g(b_j)\right), g^{-1}\left(\sum_{j=1}^{n} w_j g(c_j)\right); \wedge_{j=1}^{n} \alpha_{\tilde{A}_j}, \vee_{j=1}^{n} \beta_{\tilde{A}_j}, \vee_{j=1}^{n} \gamma_{\tilde{A}_j}\rangle$$

Single valued triangular neutrosophic weight power averaging operator (SVTNWPAO) is defined as:

SVTNWPAO $(\tilde{A}_1, \tilde{A}_2 \ldots \tilde{A}_n)$

$= \langle g^{-1}\left(\sum_{j=1}^{n} g(a_j)^{w_j}\right), g^{-1}\left(\sum_{j=1}^{n} g(b_j)^{w_j}\right), g^{-1}\left(\sum_{j=1}^{n} g(c_j)^{w_j}\right); \wedge_{j=1}^{n}\alpha_{\tilde{A}_j}, \vee_{j=1}^{n}\beta_{\tilde{A}_j}, \vee_{j=1}^{n}\gamma_{\tilde{A}_j}\rangle$

here, g is a continuous strictly monotone increasing function.

3.7.1.2 MEREC method

The following steps are used to calculate criteria weights by MEREC method:

Step 1. Suppose, there are m alternatives and n criteria, and the form of the decision-matrix is as follows:

$$D = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \tag{3.7.1}$$

Step 2. Normalized decision matrix is obtained by using a simple linear normalization to scale the elements of the decision-matrix as follows:

$$n_{ij}^{x} = \begin{cases} \dfrac{\min_k x_{kj}}{x_{ij}}, & if\ j\ is\ benificial\ criteria \\ \dfrac{x_{ij}}{\max_k x_{kj}}, & if\ \ j\ is\ non-benificial\ criteria \end{cases} \tag{3.7.2}$$

Step 3. According to the normalized values obtained from the previous step, it is sure that smaller values of n_{ij}^{x} yield greater values of performances. The following equation is used to calculate the overall performance of the alternatives (S_i):

$$S_i = \ln\left(1 + \left(\frac{1}{n}\sum_j \left|\ln\left(n_{ij}^{x}\right)\right|\right)\right) \tag{3.7.3}$$

Step 4. The difference between this step and step 3 is that the alternatives' performances are calculated based on removing each criterion separately. Let, S'_{ij} denotes the overall performance of i^{th} alternative concerning the removal of j^{th} criterion. The following equation is used for the calculations of this step:

$$S'_{ij} = \ln\left(1 + \left(\frac{1}{n}\sum_{k,k\neq j} \left|\ln\left(n_{ik}^{x}\right)\right|\right)\right) \tag{3.7.4}$$

Step 5. In this step, the removal effect of the j^{th} criterion $\left(E_j\right)$ based on the values obtained from step 3 and step 4 is calculated as:

$$E_j = \sum_i \left|S'_{ij} - S_i\right| \quad (3.7.5)$$

Step 6. In the final step, each criterion's objective weight is calculated using the removal effects $\left(E_j\right)$ of step 5.

$$w_j = \frac{E_j}{\sum_k E_k} \quad (3.7.6)$$

4.7.1.3 CoCoSo method

The following steps are used to rank alternatives by CoCoSo method:

Step 1. For m alternatives and n criteria, the initial decision-matrix is defined as follows:

$$D = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \quad (3.7.7)$$

Step 2. Normalized decision matrix is obtained by normalizing of criteria values based on compromise normalization equation given below:

$$r_{ij} = \begin{cases} \frac{x_{ij} - \min_i x_{ij}}{\max_i x_{ij} - \min_i x_{ij}}; for\ benificial\ criteria \\ \frac{\max_i x_{ij} - x_{ij}}{\max_i x_{ij} - \min_i x_{ij}}; for\ cost\ criteria \end{cases} \quad (3.7.8)$$

Step 3. The sum of the weighted comparability sequence (S_i) and an amount of the power weight of comparability sequences for each alternative (P_i) are calculated using the following expressions:

$$S_i = \sum_{j=1}^{n} (w_j * r_{ij}) \quad (3.7.9a)$$

S_i value is achieved according to grey relational generation approach.

$$P_i = \sum_{j=1}^{n} \left(r_{ij}\right)^{w_j} \quad (3.7.9b)$$

P_i value is achieved according to the WASPAS multiplicative attitude.

Step 4. In this step, three appraisal score strategies are used to generate relative weights of other options, which are derived using following formulae:

$$k_{ia} = \frac{P_i + S_i}{\sum_{i=1}^{m}(P_i + S_i)} \tag{3.7.10a}$$

$$k_{ib} = \frac{S_i}{\min_i S_i} + \frac{P_i}{\min_i P_i} \tag{3.7.10b}$$

$$k_{ic} = \frac{\lambda * S_i + (1-\lambda) * P_i}{\max_i \lambda * S_i + \max_i (1-\lambda) * P_i};\ 0 \le \lambda \le 1 \tag{3.7.10c}$$

Step 5. The final ranking of the alternatives is determined according to the value k_i,

$$k_i = \frac{(k_{ia} + k_{ib} + k_{ic})}{3} + (k_{ia} * k_{ib} * k_{ic})^{1/3} \tag{3.7.11}$$

3.7.2 Proposed model

The step by step procedure of Model [7] is elaborated as follows:

Step 1. Identification of criteria and alternatives

In the first step, a set of criteria $\{c_1, c_2, c_{3,} \dots, c_n\}$and alternatives $\{a_1, a_2, a_{3,} \dots, a_m\}$ are determined and evaluated according to the opinion of the decision makers.

Step 2. Determination of the decision matrix

In this step, initial decision matrix $\widetilde{D}_{ij} = [\varepsilon_{ij}]_{m \times n}$ is obtained here, ε_{ij} presents the evaluation of i^{th} alternative corresponding j^{th} criterion. The entries of $\widetilde{D}_{ij}$ are presented in SVTN, i.e., $\varepsilon_{ij} = \langle (a_{ij}, b_{ij}, c_{ij}); \alpha_{\tilde{A}_{ij}}, \beta_{\tilde{A}_{ij}}, \gamma_{\tilde{A}_{ij}} \rangle$ according the defined linguistic terms in Table 3.7.1.

Table 3.7.1: STVNs for evaluation

Linguistic term	STVNs
Very Low (VL)	<(0,0,1);0,1,1>
Moderate Low (ML)	<(0,1,3);0.17,0.85,0.83>
Low (L)	<(1,3,5);0.33,0.75,0.67>
Medium (M)	<(3,5,7);0.5,0.5,0.5>
High (H)	<(5,7,9);0.67,0.25,0.33>
Medium High (MH)	<(7,9,10);0.83,0.15,0.17>
Very High (VH)	<(9,10,10);1,0,0>

Step 3. Normalize fuzzy decision matrix:

The initial decision matrix is converted to normalized decision matrix $\tilde{N}_{ij} = [\delta_{ij}]_{m\times n}$. As the decision matrix in step 2 already step up according to linguistic terms of Table 3.7.1 so there is no need of different formulae for normalizing beneficial criteria and cost criteria values. Here, the largest SVTN $\left(\varepsilon_{ij}{}^{max}\right)$ is indentified by the help of score function and the entries are normalized as follows:

$$\delta_{ij} = \frac{\varepsilon_{ij}}{\varepsilon_{ij}{}^{max}} = \langle(\frac{a_{ij}}{c_{ij}{}^{max}}, \frac{b_{ij}}{b_{ij}{}^{max}}, \frac{c_{ij}}{a_{ij}{}^{max}}); \alpha_{\varepsilon_{ij}} \wedge \alpha_{\varepsilon_{ij}{}^{max}}, \beta_{\varepsilon_{ij}} \vee \beta_{\varepsilon_{ij}{}^{max}}, \gamma_{\varepsilon_{ij}} \vee \gamma_{\varepsilon_{ij}{}^{max}}\rangle =$$
$$\langle(a_{ij}{}^{*}, b_{ij}{}^{*}, c_{ij}{}^{*}); \alpha_{\tilde{A}_{ij}}{}^{*}, \beta_{\tilde{A}_{ij}}{}^{*}, \gamma_{\tilde{A}_{ij}}{}^{*}\rangle \quad (3.7.12)$$

Step 4. Compute overall performance of the alternatives:

A parabolic measure is employed to obtain the overall performance of the alternatives. It is a non-linear increasing function. The following equation is applied for the calculation.

$$\tilde{P}_i = \frac{1}{n}\sum_{j=1}^{n}\left(\delta_{ij}\right)^2$$

$$\tilde{P}_i = \langle(\tilde{p}_i, \tilde{q}_i, \tilde{r}_i); \tilde{\alpha}_{\tilde{A}_i}, \tilde{\beta}_{\tilde{A}_i}, \tilde{\gamma}_{\tilde{A}_i}\rangle =$$
$$\langle\left(\frac{1}{n}\sum_{j=1}^{n}\left(a_{ij}{}^{*}\right)^2, \frac{1}{n}\sum_{j=1}^{n}\left(b_{ij}{}^{*}\right)^2, \frac{1}{n}\sum_{j=1}^{n}\left(c_{ij}{}^{*}\right)^2\right); \min\left(\alpha_{\tilde{A}_{ij}}{}^{*}\right), \max\left(\beta_{\tilde{A}_{ij}}{}^{*}\right), \max\left(\gamma_{\tilde{A}_{ij}}{}^{*}\right)\rangle$$
(3.7.13)

Step 5. Enumeration of the performance of alternatives by deleting each criterion:

The alternative's performance is calculated on the basis of deletion of every criterion separately. For this, parabolic measure (depicted in Figure 3.7.1) is used as in the prior step. The collective performance of i^{th}alternative concerning the deletion of j^{th} criterion is calculated as follows:

$$\tilde{P}'_{ij} = \langle\left(\tilde{p}'_{ij}, \tilde{q}'_{ij}, \tilde{r}'_{ij}\right); \tilde{\alpha}_{\tilde{A}_{ij}}, \tilde{\beta}_{\tilde{A}_{ij}}, \tilde{\gamma}_{\tilde{A}_{ij}}\rangle$$

$$= \left(\frac{1}{n}\sum_{k,k\neq j}(a^{*}_{ih})^2, \frac{1}{n}\sum_{k,k\neq j}(b^{*}_{ih})^2, \frac{1}{n}\sum_{k,k\neq j}(c^{*}_{ih})^2; \min\left(\alpha_{\tilde{A}_{ih}}{}^{*}\right), \max\left(\beta_{\tilde{A}_{ih}}{}^{*}\right), \max\left(\gamma_{\tilde{A}_{ih}}{}^{*}\right)\right)$$
(3.7.14)

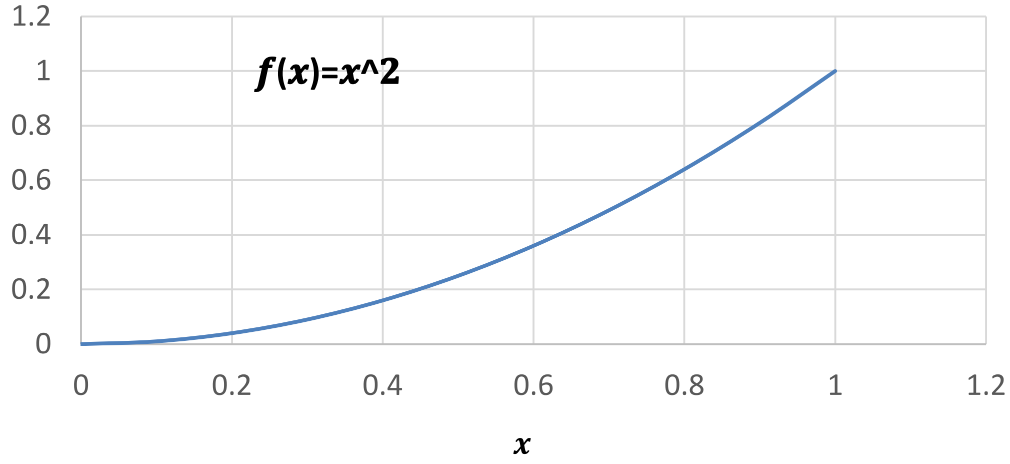


Figure 3.7.1 Parabolic measure

Step 6. Computation of the summation of Euclidean distances:

The expulsion impact of the j^{th}criterion on the basis of the values received from step 3 and step 4 is calculated as follows:

$$\tilde{E}_j=\langle\left(\sqrt{\sum_i(p'_{ij}-\tilde{p}_i)^2},\sqrt{\sum_i(\tilde{q}'_{ij}-\tilde{q}_i)^2},\sqrt{\sum_i(\tilde{r}'_{ij}-\tilde{r}_i)^2}\right);\tilde{\alpha}_{\tilde{A}_i},\tilde{\beta}_{\tilde{A}_i},\tilde{\gamma}_{\tilde{A}_i}\rangle \quad (3.7.15)$$

Step 7. Assessment of final weights of the criteria:

The objective weight of each criterion is calculated using the following equation:

$$w_j=\frac{\tilde{E}_j}{\sum_k \tilde{E}_k}, \quad (3.7.16)$$

After solving equation (3.7.16), the objective weights are transferred into crisp numbers using score function.

Step 8. Calculate the total of the weighted sum and whole of the power weight comparability sequences:

The total of the weighted sum measure (WSM)(S_i) for each alternative, i.e., the sum of the weighted comparability sequence is computed by

$$S_i = \text{SVTNWAO}\ (\delta_{i1},\delta_{i2}\ldots\delta_{in}) = \langle(a_i^{(1)},b_i^{(1)},c_i^{(1)});\alpha_{\tilde{A}_i}{}^{(1)},\beta_{\tilde{A}_i}{}^{(1)},\gamma_{\tilde{A}_i}{}^{(1)}\rangle$$

$$=\langle g^{-1}\left(\sum_{j=1}^{n} w_j g\left(a_{ij}{}^*\right)\right),g^{-1}\left(\sum_{j=1}^{n} w_j g\left(b_{ij}{}^*\right)\right),g^{-1}\left(\sum_{j=1}^{n} w_j g\left(c_{ij}{}^*\right)\right);\wedge_{j=1}^{n}\alpha_{\tilde{A}_{ij}}{}^*,\vee_{j=1}^{n}\beta_{\tilde{A}_{ij}}{}^*,\vee_{j=1}^{n}\gamma_{\tilde{A}_{ij}}{}^*\rangle$$

$$\forall i \quad (3.7.17a)$$

The whole of the weighted product measure (WPM)(P_i) for each alternative, i.e., the amount of the power weight comparability sequence is computed by

$$P_i = \text{SVTNWPAO}\ (\delta_{i1}, \delta_{i2} \dots \delta_{in})\ \forall i{=}\langle (a_i{}^{(2)}, b_i{}^{(2)}, c_i{}^{(2)});\ \alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(2)} \rangle$$

$$= \langle g^{-1}\big(\textstyle\sum_{j=1}^{n} g\big(a_{ij}{}^{*}\big)^{w_j}\big), g^{-1}\big(\sum_{j=1}^{n} g\big(b_{ij}{}^{*}\big)^{w_j}\big), g^{-1}\big(\sum_{j=1}^{n} g\big(c_{ij}{}^{*}\big)^{w_j}\big);\ \wedge_{j=1}^{n}\alpha_{\tilde{A}_{ij}}{}^{*}, \vee_{j=1}^{n}\beta_{\tilde{A}_{ij}}{}^{*}, \vee_{j=1}^{n}\gamma_{\tilde{A}_{ij}}{}^{*} \rangle$$

$\forall i$ (3.7.17b)

Step 9. Compute relative weights or balanced compromise scores of the alternative

In this step, three appraisal score degrees are utilized to create relative weights of the option, given as

$$U_i^{(1)} = \frac{S_i + P_i}{\sum_{i=1}^{m}(S_i + P_i)}$$

$$= \frac{\langle (a_i{}^{(1)} + a_i{}^{(2)}, b_i{}^{(1)} + b_i{}^{(2)}, c_i{}^{(1)} + c_i{}^{(2)});\ \alpha_{\tilde{A}_i}{}^{(1)}, \beta_{\tilde{A}_i}{}^{(1)}, \gamma_{\tilde{A}_i}{}^{(1)};\ \alpha_{\tilde{A}_i}{}^{(1)}\wedge\alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(1)}\vee\beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(1)}\vee\gamma_{\tilde{A}_i}{}^{(2)} \rangle}{\sum_{i=1}^{m}\big(\langle (a_i{}^{(1)} + a_i{}^{(2)}, b_i{}^{(1)} + b_i{}^{(2)}, c_i{}^{(1)} + c_i{}^{(2)});\ \alpha_{\tilde{A}_i}{}^{(1)}, \beta_{\tilde{A}_i}{}^{(1)}, \gamma_{\tilde{A}_i}{}^{(1)};\ \alpha_{\tilde{A}_i}{}^{(1)}\wedge\alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(1)}\vee\beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(1)}\vee\gamma_{\tilde{A}_i}{}^{(2)} \rangle\big)}$$

(3.7.18a)

$$U_i^{(2)} = \frac{S_i}{\min_i S_i} + \frac{P_i}{\min_i P_i} =$$

$$\frac{\langle (a_i{}^{(1)}, b_i{}^{(1)}, c_i{}^{(1)}); \alpha_{\tilde{A}_i}{}^{(1)}, \beta_{\tilde{A}_i}{}^{(1)}, \gamma_{\tilde{A}_i}{}^{(1)} \rangle}{\min_i \langle (a_i{}^{(1)}, b_i{}^{(1)}, c_i{}^{(1)}); \alpha_{\tilde{A}_i}{}^{(1)}, \beta_{\tilde{A}_i}{}^{(1)}, \gamma_{\tilde{A}_i}{}^{(1)} \rangle} + \frac{\langle (a_i{}^{(2)}, b_i{}^{(2)}, c_i{}^{(2)}); \alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(2)} \rangle}{\min_i \langle (a_i{}^{(2)}, b_i{}^{(2)}, c_i{}^{(2)}); \alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(2)} \rangle} \quad (3.7.18b)$$

$$U_i^{(3)} = \frac{\lambda * S_i + (1-\lambda) * P_i}{\max_i \lambda * S_i + \max_i (1-\lambda) * P_i} =$$

$$= \frac{\left\langle \begin{matrix} \lambda * (a_i{}^{(1)} + (1-\lambda) * a_i{}^{(2)}, \lambda * b_i{}^{(1)} + (1-\lambda) * b_i{}^{(2)}, \lambda * c_i{}^{(1)} + (1-\lambda) * c_i{}^{(2)}); \\ \alpha_{\tilde{A}_i}{}^{(1)}, \beta_{\tilde{A}_i}{}^{(1)}, \gamma_{\tilde{A}_i}{}^{(1)}; \alpha_{\tilde{A}_i}{}^{(1)}\wedge\alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(1)}\vee\beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(1)}\vee\gamma_{\tilde{A}_i}{}^{(2)} \end{matrix} \right\rangle}{\left[\begin{matrix} \max_i \langle (\lambda * a_i{}^{(1)}, \lambda * b_i{}^{(1)}, \lambda * c_i{}^{(1)}); \alpha_{\tilde{A}_i}{}^{(1)}, \beta_{\tilde{A}_i}{}^{(1)}, \gamma_{\tilde{A}_i}{}^{(1)} \rangle + \\ \max_i \langle ((1-\lambda) * a_i{}^{(2)}, (1-\lambda) * b_i{}^{(2)}, c(1-\lambda) *_i{}^{(2)}); \alpha_{\tilde{A}_i}{}^{(2)}, \beta_{\tilde{A}_i}{}^{(2)}, \gamma_{\tilde{A}_i}{}^{(2)} \rangle \end{matrix} \right]}$$

(3.7.18c)

here, min and max SVTNs are determined using score and accuracy degrees and comparison among SVTNs, and τ ∈ [0, 1] is the coefficient of compromise decision mechanism.

Step 10. Compute the final aggregating compromise index of the alternatives and rank them

The final ranking index U_i to specify the importance of the options is given by (as more important as better)

$$U_i = \frac{\left(U_i^{(1)} + U_i^{(2)} + U_i^{(3)}\right)}{3} + \left(U_i^{(1)} * U_i^{(2)} * U_i^{(3)}\right)^{1/3} \qquad (3.7.19)$$

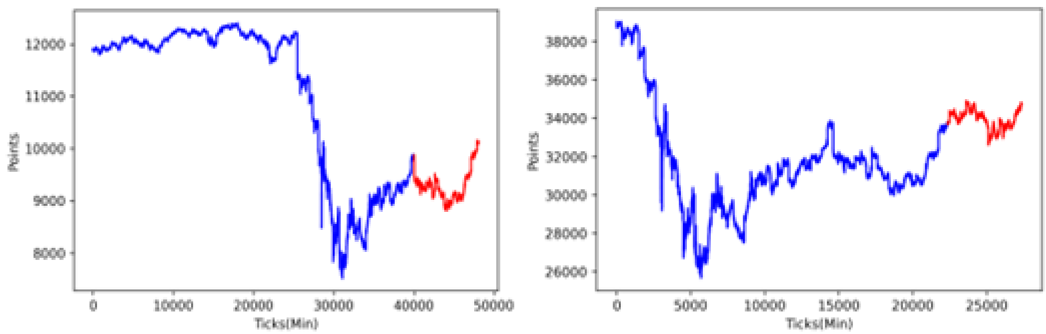
Results and Discussion

4.1 Model [1]: Deep reinforcement learning based multi-objective systems for financial trading

The proposed multi-objective systems are tested on real-world financial data. Stock index futures contracts from two stock exchange centers of India; NSE and BSE are used for illustration. First is, Nifty 50 which is calculated as average of top 50 stocks traded in NSE and second is Sensex, which represents average of top 30 stocks traded in BSE. Both indices having high liquidity are heavily traded in India. Trader can execute trading actions as long and short positions in almost real time. When price of stock goes higher (respectively, lower) it is ideal to take long position (respectively, short) to make profit.

4.1.1 Experimental setup

We have used minute-level close prices of both indices. There is 1 minute difference between consecutive closing prices z_t and z_{t+1}. The historic charts in minute resolution of both indices are shown in Figure 4.1.1. The Nifty 50, ranges from 2019/11 to 2020/06 and Sensex, data ranges from 2020/03 to 2020/06. If Nifty 50 index rise (down) by 1 point it lead to the reward of 1000 Indian rupees for long (short) position and if Sensex stock increase (decrease) by 1 point make reward of 100 Indian rupees for long (short) position.



(a) Nifty (b) Sensex

Figure 4.1.1: Closing prices (minute resolution) of Nifty and Sensex future contracts. Blue part: Training data. Red part: Testing data

In Figure 4.1.1(a), we observe that volatility in Nifty training data. Initially, there is less variation upto 25000 ticks, then prices goes very down because of the fallout in global market due to COVID-19 pandemic. After 30000 ticks again market starts to lift up and goes to little upward direction. In testing period also we can see some upward and downward movements.

In Figure 4.1.1(b), initially, Sensex prices showing downward movement as these are the prices of March month, starting of the lockdown in India. After 5000 ticks prices slightly goes up and continue with less volatility. Testing data also have upward and downward movements.

We set up the deep neural network with training data and initialized with 50 previous return values as raw features. Raw features input into LSTM autoencoder having two LSTM cell with 30 and 20 hidden nodes, respectively. Then 20 nodes of encoded features input into DRL network having LSTM cell with 10 hidden nodes. The output from the LSTM node input into last layer of DRL network having one node for decision (policy) making. The decision made at each tick collectively calculates the value of multi-objective functions in (3.1.5) and (3.1.6), i.e., profit under risk at the end of the trading period. Lastly, the whole neural network is trained to maximize the functions.

4.1.2 Evaluation and performance analysis

In this section, we evaluate both trading systems (named as MODRL1 and MODRL2) on the data. The performances of both systems are compared with DRL system and buy and hold strategy. In DRL system, window of 5 time steps is set to calculate moving averages, 50 recent returns values are used as input features and trained it for 100 epochs. MODRL1 and MODRL2 are also trained for 100 epochs.

The testing data of both indices and profit and loss (P&L) curves of all trading systems are shown in Figure (4.1.2) and Figure (4.1.3), respectively. The quantitative evaluation of the performance of all trading systems is summarized in Table 4.1.1, here, TP represents the total profit and TT represents the total no. of transactions.

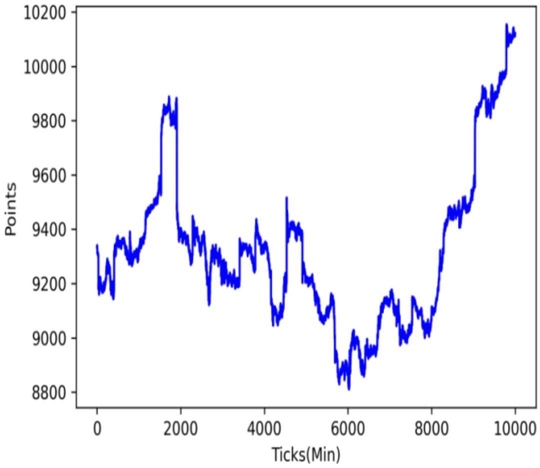
Figure 4.1.2(a) represents the testing data from 2020/04 to 2020/06 of Nifty and Figure 4.1.2(b) depicts the profit and loss values of the trading systems at every time step. It clearly shows that the MODRL1 and MODRL2 exceed the total profit gained by DRL and buy and hold. There is no such big difference in the performance of MODRL1 and MODRL2. Buy and hold strategy gave better result than DRL. DRL gained good profit at start but afterward perform worse among all systems because of high deviation in the data after 8000 ticks.

Figure 4.1.3(a) represents the testing data from 2020/05 to 2020/06 of sensex. Figure 4.1.3 (b) depicts the profit and loss curves of the trading systems. Here also MODRL1 and MODRL2 outperform DRL and buy and hold on total profit. MODRL2 total profit is higher than MODRL1 at 1200 tick then MODRL2 not gained more profit and try to sustain its performance. This behavior of MODRL2 is because of the downward trend in the middle of testing period. Again buy and hold overall performed better than DRL.

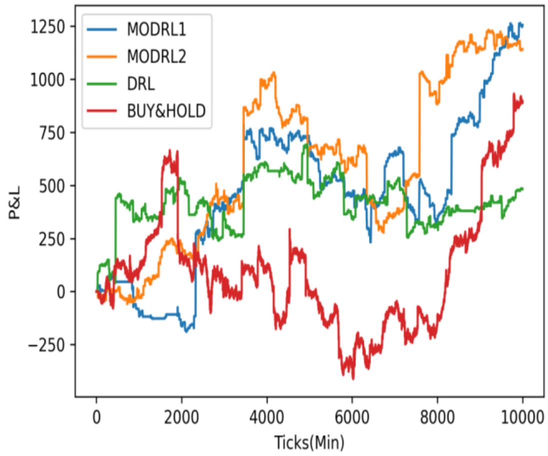
From Table 4.1.1, we can note that the both multi objective models acquire more total profit than the conventional DRL model and simply buying and holding stocks. Total no. of transactions is also more than other systems. There is no significant difference observed in total profit of MODRL1 and MODRL2. Even MODRL2 make profit in downward trends too which means MODRL2 can be helpful if trader is concern only for downward movement of the market. Therefore, both multi objective functions can be recommended to use practically over another trading systems, these systems are not only making profit but also restraining risk of the trader.

Table 4.1.1: Performances of different trading systems on both contracts

Models	Nifty		Sensex	
	TT	TP	TT	TP
MODRL1	1901	1254.80	741	2232.63
MODRL2	1896	1141.75	754	2309.22
DRL	1201	428.80	501	65.61
Buy &Hold	-	894.12	-	989.62

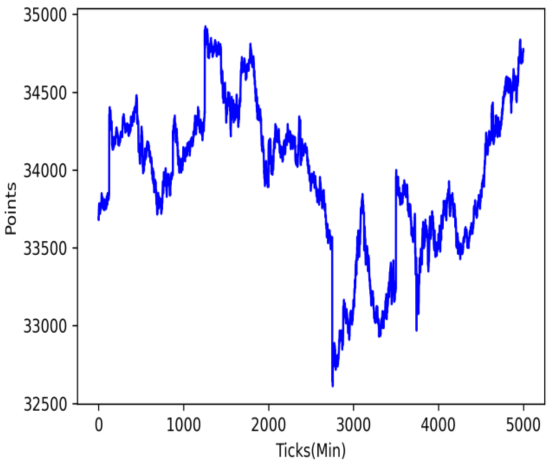


(a) Nifty (Testing data)

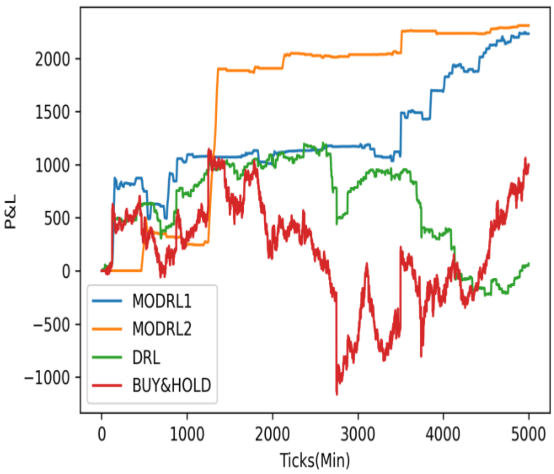


(b) Performances

Figure 4.1.2: Nifty testing data and performances (P&L curves) of different trading systems



(a) Sensex (Testing data)



(b) Performances

Figure 4.1.3: Sensex testing data and performances (P&L curves) of different trading systems

4.2 Model [2]: Stock portfolio selection based on fuzzy time series forecasting using fuzzy c-means clustering and deep learning techniques

In this study, the suggested model is examined on equities listed on the NSE to verify its applicability. The stocks of following 10 companies are taken:

1. Apollo Hospitals Enterprise Limited (APOLLOHOSP)
2. Ashok Leyland Limited (ASHOKLEY)
3. Bajaj Finance Limited (BAJFINANCE)
4. Divi's Laboratories Limited (DIVISLAB)
5. Jubilant Foodworks Limited (JUBLFOOD)
6. Pidilite Industries Limited (PIDILITIND)
7. Reliance Industries Limited (RELIANCE)
8. State Bank of India (SBIN)
9. Tata Steel Limited (TATASTEEL)
10. Tata Consultancy Services Limited (TCS)

We aimed to build a portfolio by selecting equities from the above list by estimating their closing prices for the financial year 2021. The suggested model is implemented using daily closing prices from the year 2017 to 2020. The data is obtained from https://finance.yahoo.com/. The model's detailed implementation is discussed below:

4.2.1 Forecasting closing prices of stocks by proposed FTSF method

The summary of the collected data is given in Table 4.2.1. The simulation of the FTSF method is done on jupyter notebook; first universe of discourse is defined by taking $d = 5$ for all the time series. Then find optimal number of clusters and get cluster centers by FCM algorithm using pre-defined scikit-fuzzy package in jupyter which is given in Table 4.2.2. By taking midpoints of cluster centers unequal lengths of intervals are calculated. Partition of UOD of all time series into unequal intervals is given in Table 4.2.3. After that, the interval index number in which each datum of the time series falls is identified and membership value of the datum in that interval is calculated by the expression given by equation 3.2.9. The crisp time series is normalized by min-max scalar of scikit learn package. Then input-output pattern is defined. For instance, the input-output pattern for APOLLOHOSP is presented in Table 4.2.4 for better understanding. Then crisp time series is divided into training and testing data respectively. Here, we take closing prices from 2017-2020 (988 observations) as training data and closing prices of the year 2021 (246 observations) as testing data.

The in-sample data (training data) is processed by SVM module of scikit-learn and the input part of the out-sample data (testing data) is passed to trained SVM and predicted values are obtained. The obtained prediction is denormalized by inverse min-max scalar of scikit-learn package. For multilayer perceptron (MLP) model TensorFlowkeras API is used to setup the neural network. We considered one hidden layer having the equal number of neurons as in input layer for every time series to make a fair comparison and used adam optimizer to minimize mean squared error. All the graphs presented in the study are plotted by matplotlib library of python programming package.

The processor is upgraded to GPU to implement the machine learning libraries which take less time in computation. The time complexity of FCM algorithm is (nc^2di), here, n is the number of observations, c is the number of clusters, d is the dimension and i is the number of iteration. In the proposed method, the data is one dimensional and FCM runs to 100 epochs. The scikit learn library for implementing SVM has time complexity $O(n^2m)$, here, n is the number of training sample and m is the number of features. Complexity in training MLP depends upon the number of sample data and number of epochs, here MLP is trained for 1000 epochs for each dataset. After training, MLP predict with $O(mn_{l1} + n_{l1}n_{l2})$ complexity here, m is the number of features and n_{li} is the number of nodes in the i^{th} layer.

Table 4.2.1: Summary of closing price series datasets

Stocks	Minimum Observation	Maximum observation	Round-off range	Step size
APOLLOHOSP	927.1	5733.95	[900-5800]	817
ASHOKLEY	34.45	165.55	[30-170]	28
BAJFINANCE	870.05	7929.3	[800-8000]	1200
DIVISLAB	545.3	5372.15	[500-5400]	980
JUBLFOOD	412.5	4525.7	[400, 4600]	700
PIDILITIND	604.5	2501.8	[600, 2600]	500
RELIANCE	503.18	2731.85	[500, 2800]	460
SBIN	150.85	530.45	[150, 550]	100
TATASTEEL	253.75	1519.4	[200, 1600]	350
TCS	1083.95	3954.55	[1000,4000]	1000

Table 4.2.2: Cluster centers of the datasets

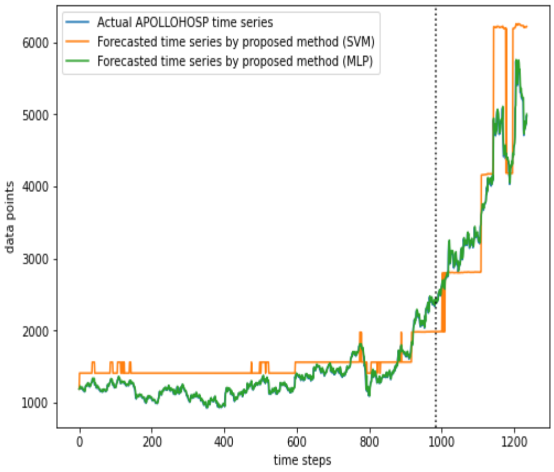
Stocks	No. of clusters	Cluster centers
APOLLOHOSP	6	1168.4,1389.4,2180,3055.8,3975.9,4797.6
ASHOKLEY	5	59.99, 82.84, 93.97, 119.15, 131.89
BAJFINANCE	6	1690.56, 2385.26, 3257.22, 4108.45, 5358.05, 7034.62
DIVISLAB	5	959.91, 1587.56, 2192.97, 3517.8, 4693.05
JUBLFOOD	6	648.1, 1239.4, 1463.42, 1784.48, 2830.36, 3737.7
PIDILITIND	4	853.58, 1195.5, 1481.69, 2207.13
RELIANCE	5	840.18, 1101.96, 1331.91, 1979.18, 2209.42
SBIN	4	209.13, 278.22, 318.79, 428.78
TATASTEEL	4	415.8, 531.53, 660.38, 1229.15
TCS	3	1381.79, 2098.42, 3216.243

Table 4.2.3: Partitioning of UOD in unequal intervals of all datasets

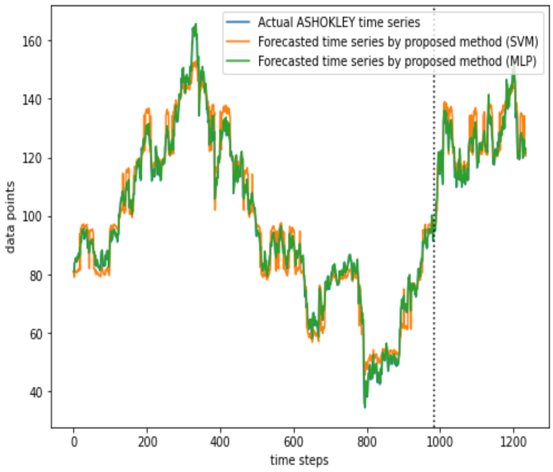
Stocks	UOD	Partition UOD in unequal intervals
APOLLOHOSP	[922.1,5738.95]	[922.1, 1278.87]; [1278.87,1784.69]; [1784.69, 2617.89]; [2617.89, 3515.80]; [3515.80, 4386.72]; [4386.72, 5738.95]
ASHOKLEY	[29.45,170.55]	[29.45, 71.41]; [71.41, 88.40]; [88.40, 106.56]; [106.56, 125.52]; [125.52, 170.55]
BAJFINANCE	[865.05,7934.3]	[865.05, 2037.91]; [2037.91, 2821.24]; [2821.24, 3682.83]; [3682.83, 4733.25]; [4733.25, 6196.33]; [6196.33, 7934.3]
DIVISLAB	[540.3,5377.15]	[540.3, 1273.73]; [1273.73, 1890.26]; [1890.26, 2855.38]; [2855.38, 4105.42]; [4105.42, 5377.15]
HINDUNILVR	[816.6,2817.45]	[816.6, 1211.84]; [1211.84, 1546.05]; [1546.05, 1933.56]; [1933.56, 2241.65]; [2241.65, 2817.45]
JUBLFOOD	[407.5,4530.7]	[407.5, 943.75]; [943.75, 1351.41]; [1351.41, 1623.95]; [1623.95, 2307.42]; [2307.42, 3274.03]; [3274.03, 4530.7]
PIDILITIND	[599.5,2506.8]	[599.5, 1024.54]; [1024.54, 1338.59]; [1338.59, 1844.41]; [1844.41, 2506.8]
RELIANCE	[498.18,2736.85]	[498.18, 971.07]; [971.07, 1216.94]; [1216.94, 1655.55]; [1655.55, 2094.3]; [2094.3, 2736.85]
SBIN	[145.85,535.45]	[145.85, 243.67]; [243.67, 298.5]; [298.5, 373.78], [373.78, 535.45]
TATASTEEL	[248.75,1524.4]	[248.75, 473.66]; [473.66, 595.95]; [595.95, 944.76]; [944.76, 1524.4]
TCS	[1078.95,3959.55]	[1078.95, 1740.10], [1740.10, 2657.33], [2657.33, 3959.55]

Table 4.2.4: Input features and target values for APOLLOHOSP

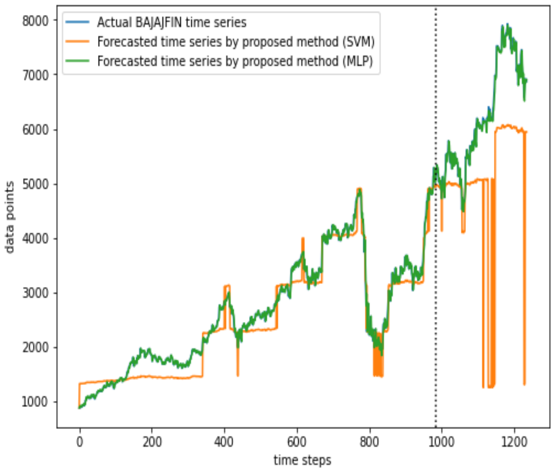
Date	Actual closing price	Normalised time series	Forecasted Date	Input Feature		Target values	
				Interval index	Membership value in the interval		
2-Jan-2017	1189.2	0.0545	3-Jan-2017	1	0.2513	0.0579	T
3-Jan-2017	1205.45	0.0579	4-Jan-2017	1	0.2058	0.0611	R
4-Jan-2017	1220.95	0.0611	5-Jan-2017	1	0.1623	0.0583	A
5-Jan-2017	1207.25	0.0583	6-Jan-2017	1	0.2007	0.0560	I
6-Jan-2017	1196.5	0.0560	9-Jan-2017	1	0.2309	0.0548	N
9-Jan-2017	1190.7	0.0548	10-Jan-2017	1	0.2471	0.0561	I
10-Jan-2017	1196.9	0.0561	11-Jan-2017	1	0.2298	0.0566	N
11-Jan-2017	1199.3	0.0566	12-Jan-2017	1	0.2230	0.0580	G
12-Jan-2017	1206	0.0580	13-Jan-2017	1	0.2042	0.0542	
13-Jan-2017	1187.4	0.0542	16-Jan-2017	1	0.2564	0.0519	D
.	.	.	.	.	.	.	A
.	.	.	.	.	.	.	T
.	.	.	.	.	.	.	A
29-Dec-2020	2376.5	0.3015	30-Dec-2020	3	0.2897	0.3056	S
30-Dec-2020	2396.35	0.3056	31-Dec-2020	3	0.2658	0.3090	E
31-Dec-2020	2412.8	0.3090	1-Jan-2021	3	0.2461	0.3095	T
1-Jan-2021	2414.85	0.3095	4-Jan-2021	3	0.2436	0.3083	TE
4-Jan-2021	2409.25	0.3083	5-Jan-2021	3	0.2504	0.3292	ST
5-Jan-2021	2509.45	0.3292	6-Jan-2021	3	0.1301	0.3276	IN
.	.	.	.	.	.	.	G
.	.	.	.	.	.	.	
.	.	.	.	.	.	.	DA
28-Dec-2021	4844.2	0.8149	29-Dec 2021	6	0.6617	0.8424	TA
29-Dec 2021	4976.6	0.8424	30-Dec 2021	6	0.5637	0.8412	SE
30-Dec 2021	4970.75	0.8412	-	-	-	-	T


Actual APOLLOHOSP time series
Forecasted time series by proposed method (SVM)
Forecasted time series by proposed method (MLP)
data points
time steps

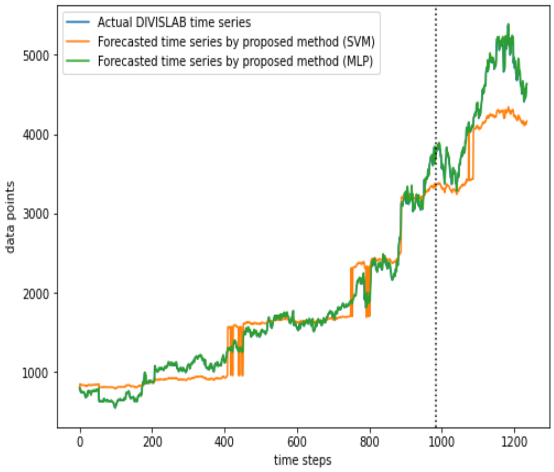
(a)


Actual ASHOKLEY time series
Forecasted time series by proposed method (SVM)
Forecasted time series by proposed method (MLP)
data points
time steps

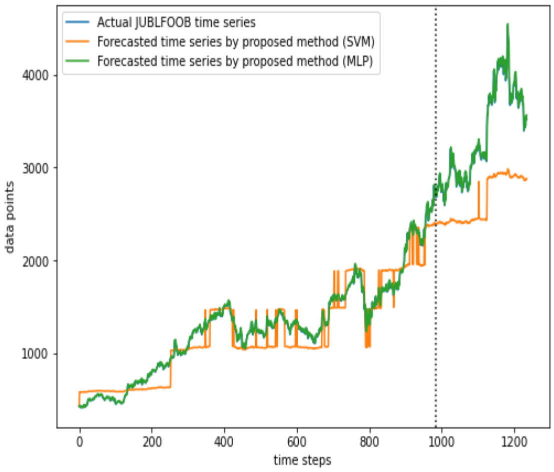
(b)


Actual BAJAJFIN time series
Forecasted time series by proposed method (SVM)
Forecasted time series by proposed method (MLP)
data points
time steps

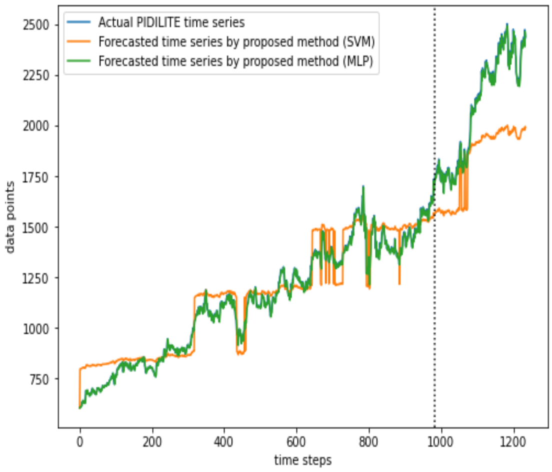
(c)


Actual DIVISLAB time series
Forecasted time series by proposed method (SVM)
Forecasted time series by proposed method (MLP)
data points
time steps

(d)


Actual JUBLFOOB time series
Forecasted time series by proposed method (SVM)
Forecasted time series by proposed method (MLP)
data points
time steps

(e)


Actual PIDILITE time series
Forecasted time series by proposed method (SVM)
Forecasted time series by proposed method (MLP)
data points
time steps

(f)

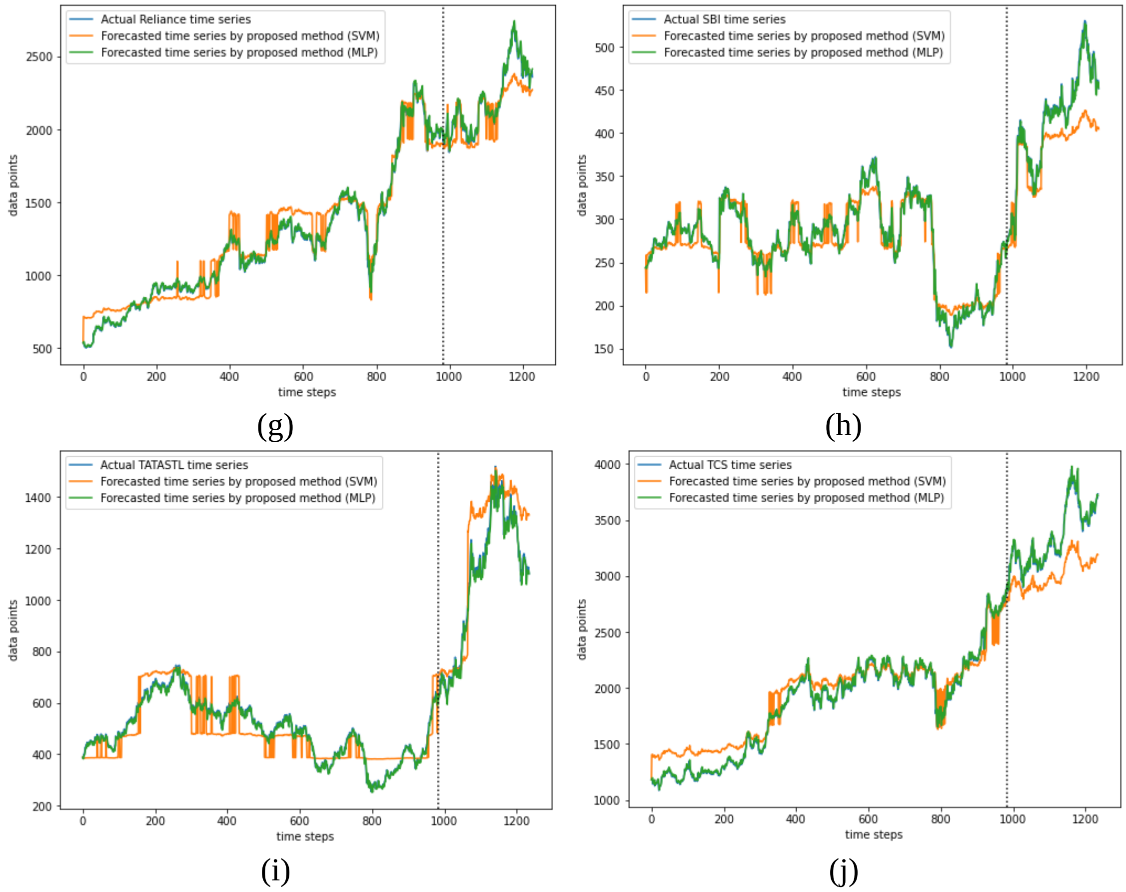


Figure 4.2.1: Actual and forecasted closing prices of all companies.

The actual and forecasted closing prices for all companies are depicted in Figure 4.2.1(a)-(j). Actual closing prices are given in blue, whereas forecasted closing prices using MLP and SVM are shown in green and orange, respectively. In the plots, a dotted line separates the training and testing data. Training data is plotted before the dotted line, and testing data is plotted after. The plotted graphs show that the proposed method utilizing MLP fit the time series quite well and provided a better forecast than SVM. Thus, we further proceed with the forecasted closing prices by MLP.

4.2.2 Portfolio construction

In this section, the annual return and risk of all the stocks for the year 2021 are estimated using the forecasted prices. The companies are ranked based on their return-risk ratio. Ranking of companies is shown in Table 4.2.5. A portfolio of top five companies is built, which includes APOLLOHOSP, SBIN, PIDILITIND, TATASTEEL, and TCS.

Table 4.2.5: Ranking of the companies

Stocks	Return	Risk	Return-Risk ratio	Ranking
APOLLOHOSP	0.7525	0.0232	32.44	1
ASHOKLEY	0.2523	0.0253	9.97	9
BAJFINANCE	0.3325	0.0235	14.15	7
DIVISLAB	0.1823	0.0185	9.82	10
JUBLFOOD	0.2923	0.0205	14.26	6
PIDILITIND	0.3250	0.0152	21.38	5
RELIANCE	0.2150	0.0162	13.27	8
SBIN	0.5178	0.0241	21.49	4
TATASTEEL	0.6025	0.0250	24.10	2
TCS	0.2705	0.0120	22.54	3

4.2.3 Portfolio optimization

The ACO portfolio optimization algorithm is executed in jupyter notebook. The parameters are set the as $r_f = 0.02$, $\alpha = 0.2$, $\beta = 0.3$, $m = 0$ and $M = 0.8$, and iterated up to 500 epochs. The maximum return is found as 0.5175. The ratio allocation obtained for this return is given in Table 4.2.6. Figure 4.2.2 depicts the accumulation of ants to the optimum objective values in each iteration and Figure 4.2.3 shows convergence of the objective value as per the propose model. It is clear from these figures that proposed ACO algorithm can effectively solve the proposed portfolio model.

Table 4.2.6: Ratio allocation for the proposed portfolio

APOLLOHOSP	SBIN	PIDILITIND	TATASTEEL	TCS
0.2081	0.2081	0.2081	0.1877	0.1877

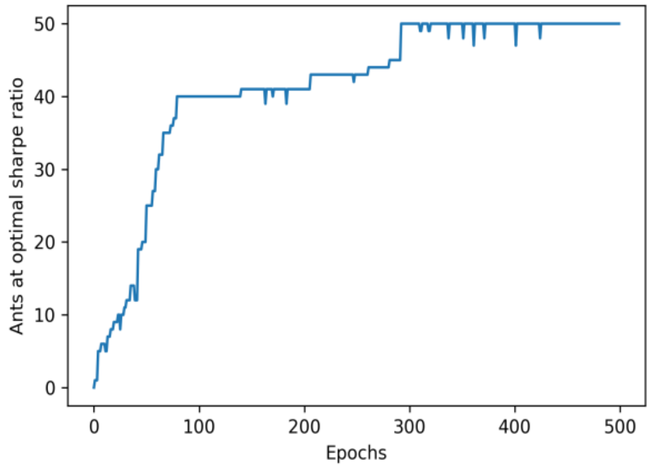


Figure 4.2.2: Ant accumulation at optimum solutions

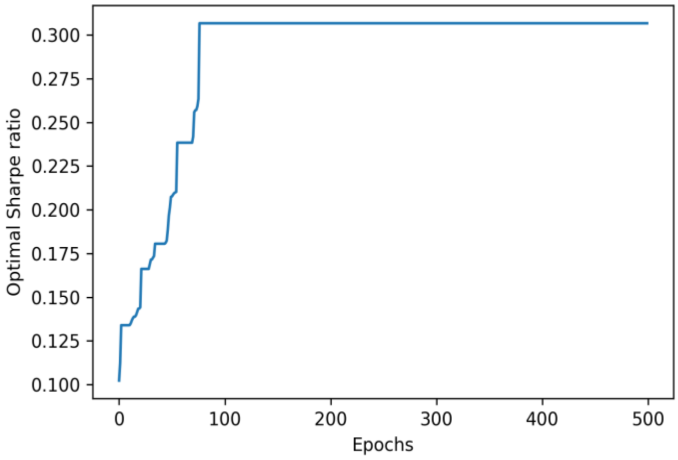


Figure 4.2.3: Convergence of Sharpe ratio

4.2.4 Effectiveness of the proposed model

To examine the effectiveness of the proposed forecasting model, the root mean square error (RMSE) and symmetric mean absolute percentage error (SMAPE) between forecasted and actual data are calculated for all companies (shown in Table 4.2.7). It is clear by the table that MLP has a lower RMSE and SMAPE than SVM. As a conclusion, it demonstrates that the proposed forecasting model through MLP training is extremely effective.

Table 4.2.7: RSME and SMAPE between the forecasted and actual closing price

Stocks	RSME by MLP	SMAPE by MLP	RSME by SVM	SMAPE by SVM
APOLLOHOSP	102.513	1.6157	774.382	14.0813
ASHOKLEY	3.003	1.8357	5.040	3.1936
BAJFINANCE	126.39	1.504	1754.370	23.957
DIVISLAB	71.038	1.2039	554.42	11.827
JUBLFOOD	73.914	1.6148	761.326	23.104
PIDILITIND	31.261	1.1379	297.543	13.639
RELIANCE	37.394	1.2821	136.817	5.0326
SBIN	8.452	1.549	43.713	8.652
TATASTEEL	31.494	2.323	137.274	9.890
TCS	47.509	1.065	358.142	10.195

Table 4.2.8 displays the actual annual return and volatility with the annual return and volatility estimated by forecasted prices. It can be seen that the actual and predicted results are extremely close. Estimated return of the proposed portfolio is also found close to actual portfolio return as presented in the Table 4.2.9. This validates the model's applicability for portfolio stock selection.

Table 4.2.8: Comparison of return and volatility between the forecasted and actual closing prices

Stocks	Actual annual return	Forecasted annual return	Actual volatility	Forecasted volatility
APOLLOHOSP	0.8016	0.7525	0.0251	0.0232
ASHOKLEY	0.2696	0.2523	0.0243	0.0253
BAJFINANCE	0.3179	0.3325	0.0208	0.0235
DIVISLAB	0.2160	0.1823	0.0159	0.1857
JUBLFOOD	0.2890	0.2923	0.0214	0.0205
PIDILITIND	0.3440	0.3250	0.0141	0.0152
RELIANCE	0.2050	0.2150	0.0163	0.0162
SBIN	0.5378	0.5178	0.0212	0.0241
TATASTEEL	0.6255	0.6025	0.0263	0.0250
TCS	0.2657	0.2705	0.0132	0.0120

Table 4.2.9: Comparison between the portfolio returns

Portfolio annual return by forecasted stock prices	0.4958
Portfolio annual return by actual stock prices	0.5175

4.3 Model [3]: An effective hybrid MCDM approach of portfolio construction using modern portfolio theory

The considered model has been illustrated with the help of NSE stocks aiming to construct a portfolio which fulfills the appeal of MPT. There are plenty of factors (such as growth, safety, performance, demand, and risk) that decide the functionality of a stock in future. In this study, 5 fundamental criteria are considered that investors or decision makers seek before investing in some stock. The criteria are as follows:

1. **Revenue(C_1):** Revenue depicts the growth of the company. The company is considered to be profitable if it earns revenue more than its expenses. Investors ponder the revenue as the health of the company before investing their money in the company.

2. **Debt Equity Ratio (D/E)(C_2):** Debt equity Ratio of a company is calculated by dividing the debt of the company to the share holder’s equity (share holders fund). It reflects the ability of a company to outstand of all the debts if the market downturn eventually. The company having D/E less than 1 is considered good for investment.

3. **Return on Equity (ROE) (C_3):** Return on equity is one of the important ratios for fundamental analysis of a stock. It measures the return or profit earned per share holder’s equity. High ROE of a company is preferred by investors to get more return on their investment.

4. **Price to Earnings ratio (P/E) (C_4):** Price to Earnings ratio measure that price of a share corresponding to the earning per share. It interprets that the stock of a company is overvalued or not. Investors try to look for a company in which they have to invest less for earning more.

5. **Price to Book Value ratio (P/B) (C_5):** Price to Book Value ratio is calculated by dividing the market price to book price of a share. It depicts the market’s value to its book value. Investors use this ratio for indentifying the potential of the stock for future investment. Stocks having P/B less than 1 are preferred as good investment.

So, the two criteria revenue (C_1) and ROE (C_3) are considered as beneficial criteria. D/R (C_2), P/E (C_4) and P/B (C_5) are taken as non-beneficial or cost criteria.

Stocks of following 20 NSE companies are taken as alternatives to investigate them over five mentioned criteria:

1. Apollo Hospitals Enterprise Ltd. (APOLLOHOSP) (A_1)
2. Reliance Industries Limited (RELIANCE) (A_2)
3. Oil & Natural Gas Corporation Limited (ONGC) (A_3)
4. Jubilant Foodworks Limited (JUBLFOOD) (A_4)
5. HDFC Bank Limited (HDFCBANK) (A_5)
6. State Bank of India (SBIN) (A_6)
7. Bajaj Finance Limited (BAJFINANCE) (A_7)
8. Dr. Reddy's Laboratories Limited (DRREDDY) (A_8)
9. Aurobindo Pharma Limited (AUROPHARMA) (A_9)
10. Tata Steel Limited (TATASTEEL) (A_{10})
11. Divi's Laboratories Limited (DIVISLAB) (A_{11})
12. ICICI Bank Limited (ICICIBANK) (A_{12})
13. Pidilite Industries Limited (PIDILITIND) (A_{13})
14. Tata Consultancy Services Limited (TCS) (A_{14})
15. TITAN Company Limited (TITAN) (A_{15})
16. Ultra Tech Cement Limited (ULTRACEMCO) (A_{16})
17. Asian Paints Limited (ASIANPAINT) (A_{17})
18. Kotak Mahindra Bank Limited (KOTAKBANK) (A_{18})
19. JSW Steel Limited (JSWSTEEL) (A_{19})
20. Hindustan Uniliver Limited (HINDUNILVR) (A_{20})

4.3.1 Determination of criteria weights by NBCM

Out of the five criteria return on equity (ROE) (C_3) is set as base criteria. The relative importance of ROE (C_3) to other criteria given by the expert is shown in Table 4.3.1.

Table 4.3.1: Base-comparison matrix

	Revenue (C_1)	DER (C_2)	P/E (C_4)	P/BV (C_5)
ROE(C_3)(crisp)	5	7	4	9
TFNs	$\langle (4,5,6); 0.8,0.15,0.2 \rangle$	$\langle (6,7,8); 0.9,0.1,0.1 \rangle$	$\langle (3,4,5); 0.6,0.35,0.4 \rangle$	$\langle (9,9,9); 1,0,0 \rangle$
Deterministic	4.594	7.088	2.775	10.125

The base- comparison vector is obtained as follows:

$\tilde{A}^*_{Base} = \{4.594, 7.088, 2.775, 10.125\}$

The optimal weights (w_j) of each criterion can be derived on solving the following nonlinearly constrained optimization problem:

$Min\ \xi$

satisfying the following conditions:

$$\left|\frac{w_3}{w_1} - 4.595\right| \le \xi, \left|\frac{w_3}{w_2} - 7.088\right| \le \xi, \left|\frac{w_3}{w_4} - 2.775\right| \le \xi, \left|\frac{w_3}{w_5} - 10.125\right| \le \xi$$

$$w_1 + w_2 + w_3 + w_4 + w_5 = 1$$

$$w_1, w_2, w_3, w_4, w_5 \ge 0$$

After solving above problem, the criteria weights are $w_1 = 0.1197, w_2 = 0.0776, w_3 = 0.55, w_4 = 0.1982, w_5 = 0.0543$ and $\xi = 0$. Hence, the consistency ratio is optimum and the solutions are fully consistent.

4.3.2 Pairwise comparison of stock by PROMETHEE I

In this section, PROMETHEE I method is executed for comparing the stocks. For this, quarterly data of 11 years (from Jan 2010 to Dec 2020) of the stocks is collected from the websites http://www.investello.com and http://www.ratestar.in. The collected data is multidimensional, therefore, we applied exponential moving average (EMA) given in equation (4.3.1) to convert the data into a single deterministic value.

$$Exponential\ moving\ average\ (EMA)$$
$$= current\ value * k + EMA(previous\ time\ stamp) * (1 - k) \quad (4.3.1)$$

here, k is the smoothing constant and $k = 2/(time\ period + 1)$.

The single numeric value is taken as the evaluation value of each alternative by taking $k = 4$. The evaluation table $\tilde{E}_{ij}$ is given in the Table 4.3.2.

Table 4.3.2: Evaluation table $(\tilde{E}_{ij})$

Stocks	Revenue (C_1)	DER (C_2)	ROE (C_3)	P/E (C_4)	P/B (C_5)
APOLLOHOSP(A_1)	9300.04	1.014	9.339	93.705	7.890
RELIANCE(A_2)	511663.60	0.737	10.258	22.35	2.253
ONGC(A_3)	390249.70	0.481	9.567	9.729	0.816
JUBLFOOD(A_4)	3376.76	0.0004	22.099	83.684	20.079
HDFCBANK(A_5)	118167.20	0.357	15.783	26.453	4.148
SBIN(A_6)	365657.50	3.122	4.347	55.626	1.262
BAJFINANCE(A_7)	18341.78	4.520	17.556	43.238	7.372
DRREDDY(A_8)	16556.73	0.264	13.050	30.281	3.897
AUROPHARMA(A_9)	19118.55	0.453	19.323	15.863	2.941
TATASTEEL(A_{10})	145832.20	1.674	6.418	9.472	0.924
DIVISLAB(A_{11})	4783.90	0.008	19.310	40.616	7.329
ICICIBANK(A_{12})	129738.40	2.667	7.448	35.384	2.064
PIDILITIND(A_{13})	6699.10	0.039	24.872	64.413	14.983
TCS(A_{14})	141259.00	0.0006	35.543	25.287	8.387
TITAN(A_{15})	18139.04	0.245	14.647	77.338	15.032
ULTRACEMCO(A_{16})	36986.65	0.648	11.513	36.098	3.729
ASIANPAINT(A_{17})	18434.92	0.056	17.550	18.996	16.306
KOTAKBANK(A_{18})	42500.38	0.839	12.646	34.391	4.480
JSWSTEEL(A_{19})	70748.37	1.601	14.703	14.525	1.845
HINDUNILVR(A_{20})	37694.80	0.007	78.643	70.781	53.329

For each criterion, the preference function and their corresponding threshold parameters are set up according to the expert suggestion presented in Table 4.3.3. The execution of the PROMETHEE I is done on visual PROMETHEE software. Table 4.3.4 represents flow value of each alternative regarding individual criteria on the basis of preference function. The computed values of positive outranking flow φ^+, negative outranking flow φ^- and net flow ($\varphi = \varphi^+ - \varphi^-$) of each company are summarized in Table 4.3.5. Using the flow values, the companies are compared and partial ranking is generated. Figure 4.3.1 displays the obtained partial ranking network of companies.

Table 4.3.3: Preference functions and threshold parameters

Criteria	**Revenue**	**DER**	**ROE**	**P/E**	**P/B**
Weights	0.1197	0.0776	0.55	0.1982	0.0543
Preference Function	U-Shape	U-shape	Usual	U-shape	Linear
Indifference threshold	10000.00	2.0	-	20.761	13.213
Preference threshold	-	-	-	-	23.641

Table 4.3.4: Flow value of alternatives (stocks) corresponding different criteria

	C_1	C_2	C_3	C_4	C_5		C_1	C_2	C_3	C_4	C_5
A_1	-0.579	0.105	-0.684	-0.895	0.053	A_{11}	-0.842	0.158	0.368	0.105	0.053
A_2	1.000	0.105	-0.474	0.421	0.080	A_{12}	0.474	-0.632	-0.790	0.158	0.082
A_3	0.895	0.158	-0.579	0.632	0.105	A_{13}	-0.790	0.158	0.790	-0.632	0.041
A_4	-0.842	0.158	0.579	-0.737	-0.185	A_{14}	0.632	0.158	0.895	0.368	0.053
A_5	0.368	0.158	0.053	0.368	0.066	A_{15}	-0.421	0.158	0.684	-0.737	0.040
A_6	0.790	-0.790	-1.000	-0.316	0.096	A_{16}	0.053	0.158	-0.368	0.158	0.069
A_7	-0.421	-0.895	0.263	0.053	0.053	A_{17}	-0.421	0.158	0.158	-0.632	0.005
A_8	-0.474	0.158	-0.158	0.316	0.068	A_{18}	0.053	0.105	-0.263	0.263	0.065
A_9	-0.421	0.158	0.474	0.474	0.073	A_{19}	0.263	0.053	-0.053	0.579	0.084
A_{10}	0.632	0.053	-0.895	0.684	0.102	A_{20}	0.053	0.158	1.000	-0.632	-1.000

Table 4.3.5: Positive flow, negative flow and net flow of alternatives (stocks)

Alternatives	φ^+	φ^-	φ	Alternatives	φ^+	φ^-	φ
A_1	0.0979	0.7099	-0.612	A_{11}	0.4541	0.3163	0.1378
A_2	0.3605	0.4053	-0.0449	A_{12}	0.2132	0.604	-0.3908
A_3	0.3724	0.4406	-0.0682	A_{13}	0.5178	0.2887	0.2291
A_4	0.4494	0.3756	0.0738	A_{14}	0.7038	0.0479	0.656
A_5	0.4603	0.2984	0.162	A_{15}	0.4973	0.3029	0.1944
A_6	0.1436	0.7179	-0.5743	A_{16}	0.309	0.4581	-0.1491
A_7	0.4318	0.3936	0.0382	A_{17}	0.3629	0.4392	-0.0763
A_8	0.3332	0.3982	-0.065	A_{18}	0.3441	0.4187	-0.0746
A_9	0.5344	0.2141	0.3203	A_{19}	0.4596	0.3336	0.126
A_{10}	0.2688	0.5401	-0.2713	A_{20}	0.6295	0.2404	0.3892

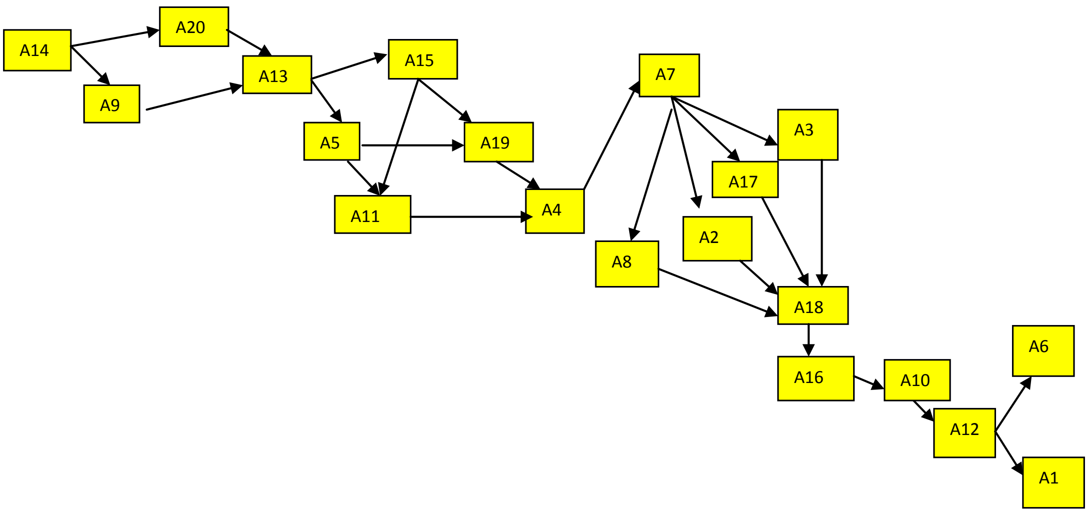


Figure 4.3.1: Partial ranking network of stocks by PROMETHEE I

4.3.3 Selection of stocks

The selection procedure of stocks is conducted in this section. For this, the adjusted closing price data of each company from 01-Jan-2020 to 31-Dec-2020 are taken and correlation between their per day returns is calculated. Figure 4.3.2 shows the correlation between each pair of stocks of 20 companies.

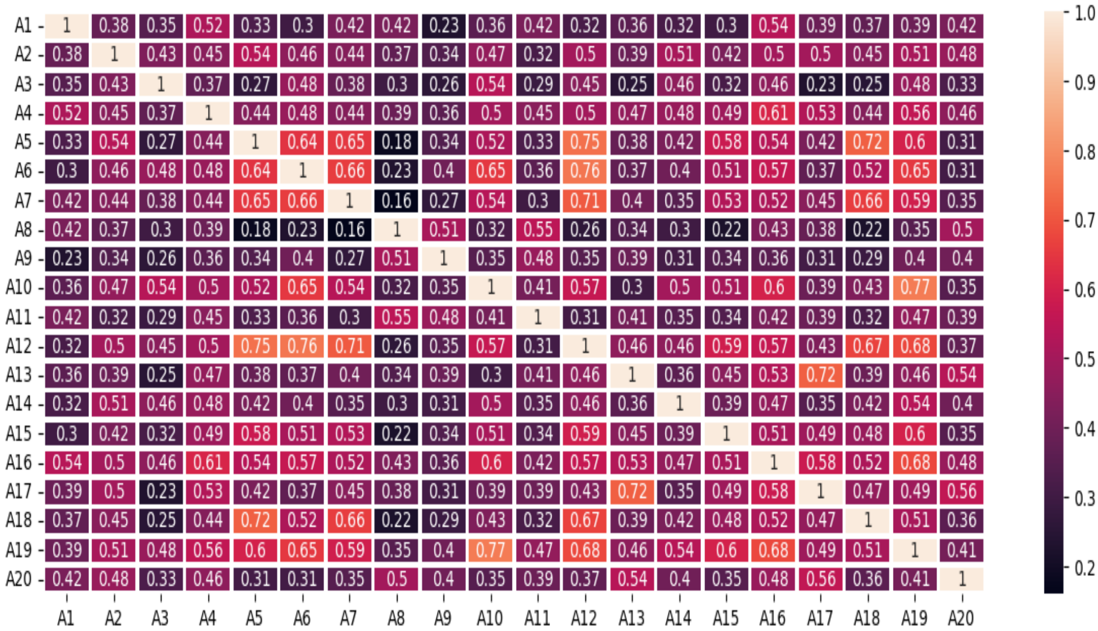


Figure 4.3.2: Correlation between the stocks.

The selection of stock is done according to the depicted four cases in section 3.3.2.3. From Figure 4.3.2, it can observe that all the stocks are positively correlated to each other. The pair of stocks correlating above 0.6 consider highly dependent. Stocks are considered moderately dependent having correlation coefficients between 0.4 to 0.6 and below 0.4 stocks are very less dependent on each other. From the comparison relation network in Figure 4.3.1, we can see A_{14} is preferred over all the stocks. It implies that A_{14} outperforms than all alternatives under the specified criteria and should consider as most valuable asset for the portfolio. Therefore, we select A_{14} as first asset of the portfolio. After A_{14} , A_9 and A_{20} are preferred over remaining stocks but A_9 and A_{20} are incomparable. Here, the selection of next stock is done as described in case 1 of proposed method. The correlation coefficient for the pair (A_{14}, A_{20}) is 0.40 while for the pair (A_{14}, A_9) its value is 0.31. Hence, A_{14} and A_9 are less related to each other. Therefore, A_9 is picked as second asset. Then the network depicts to prioritize A_{13} over other remaining stocks and the correlation coefficients of A_{13} with selected stocks (A_{14}, A_9) are (0.36, 0.39). A_{13} is very less dependent on first two stocks, so we consider A_{13} as third stock of the portfolio. A_{13} is directing to A_5 and A_{15} however A_5 is better choice for fourth asset as the correlation coefficients of a_5 with the already picked securities is {0.42, 0.34, 0.38} which is less with the securities as compared to A_{15}. Then A_5 indicates to two incomparable stocks A_{11} and A_{19}. Comparing the correlations of A_{11} and A_{19} with selected four stocks observed as {0.5, 0.48, 0.41, 0.33} and {0.54, 0.4, 0.46, 0.6} respectively, as the result A_{11} found less dependent of all stocks which conclude to choose A_{11}. After A_{11}, A_4 and then A_7 are preferred. A_7 is having high correlation with A_5 but correlation with other selected asset are less as compare to A_4. Here, we use strategy of case 3 to promote diversification. The difference between $\varphi(A_4)$ and $\mu = mean(\varphi(A_{11}), \varphi(A_4), \varphi(A_7))$ is found nearest to $\frac{\mu}{2}$. As a result, A_4 has been selected. For the seventh asset there are four choices among A_8, A_2, $A_{17,}$ A_3. The correlation coefficients of A_8, A_2, $A_{17,}$ A_3 with 6 selected stocks are noted as {0.3, 0.51, 0.34, 0.18, 0.55, 0.39}; {0.51, 0.34, 0.39, 0.54, 0.32, 0.45}; {0.35, 0.31, 0.72, 0.42, 0.39, 0.53}; {0.46, 0.26, 0.25, 0.27, 0.29, 0.37}, respectively. A_3 is observed as ideal choice among all. From the preference network A_{16}, A_{10} and A_1 are chosen as eighth, ninth and tenth assets of the portfolio. All three assets have adequate difference between the net flows. Thus, a portfolio of 10 stocks is built { A_{14}, A_9, A_{13}, A_5, A_{11}, $A_{4,}$ A_3, A_{16}, A_{10}, A_1} by choosing the path of the preference network which have diversified stocks with significant priority character.

4.3.4 Portfolio Optimization

The PSO code for portfolio optimization is executed in Google Colaboratory with the securities' data given in the Table 4.3.6 and set the parameters $r_f = 0.02$, $\alpha = 0.2$, $\beta = 0.3$, $m = 0$ and $M = 0.8$. We have taken 20 particles, $\omega = 0.5$, $c_1 = 1$, $c_2 = 2$ and optimize up to 500 iteration. The optimal solutions i.e. ratios obtained after execution are given in Table 4.3.7. The maximum value of the objective function converged to 1.67 as shown in Figure 4.3.3. For this, the maximum return is found as 0.501 with minimum volatility as 0.259. This clarifies the effectiveness of the implemented PSO algorithm for the optimizing proposed portfolio.

Table 4.3.6: Data of stocks for formulation of objective function

Securities	Returns	Stdev	Covariance										
				A_{14}	A_9	A_{13}	A_5	A_{11}	A_4	A_3	A_{16}	A_{10}	A_1
A_{14}	0.3799	0.0221	A_{14}	0.123	0.068	0.043	0.061	0.046	0.080	0.087	0.065	0.087	0.052
A_9	0.8908	0.0390	A_9	0.068	0.384	0.081	0.088	0.111	0.107	0.087	0.087	0.106	0.067
A_{13}	0.2864	0.0214	A_{13}	0.043	0.081	0.116	0.054	0.052	0.076	0.046	0.070	0.051	0.057
A_5	0.2015	0.0260	A_5	0.061	0.088	0.054	0.171	0.051	0.086	0.062	0.088	0.107	0.063
A_{11}	0.8566	0.0235	A_{11}	0.046	0.111	0.052	0.051	0.140	0.081	0.059	0.061	0.076	0.072
A_4	0.6001	0.0298	A_4	0.080	0.107	0.076	0.086	0.081	0.224	0.094	0.114	0.118	0.114
A_3	-0.098	0.0341	A_3	0.087	0.087	0.046	0.062	0.059	0.094	0.295	0.098	0.145	0.088
A_{16}	0.3602	0.0246	A_{16}	0.065	0.087	0.070	0.088	0.061	0.114	0.098	0.153	0.116	0.098
A_{10}	0.4675	0.0311	A_{10}	0.087	0.106	0.051	0.107	0.076	0.118	0.145	0.116	0.245	0.082
A_1	0.6392	0.0293	A_1	0.052	0.067	0.057	0.063	0.072	0.114	0.088	0.098	0.082	0.216

Table 4.3.7: Weight allocation of proposed portfolio

Securities	A_{14}	A_9	A_{13}	A_5	A_{11}	A_4	A_3	A_{16}	A_{10}	A_1
Rank based weights	0.369	0.124	0.124	0.124	0.124	0.027	0.027	0.027	0.027	0.027

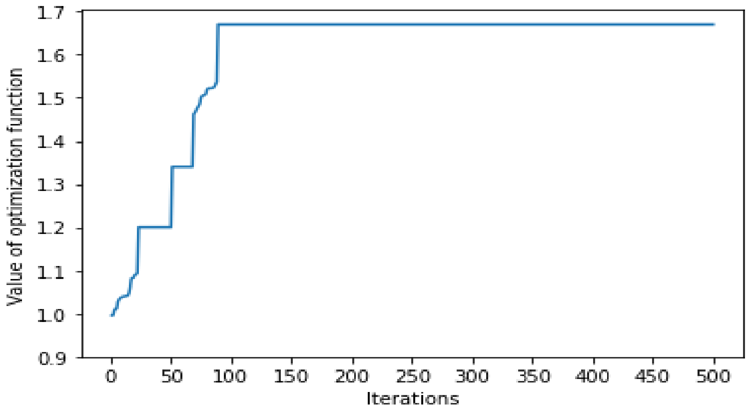


Figure 4.3.3: Convergence of the objective function

4.3.5 Comparative analysis

4.3.5.1 Comparison with other portfolios

In this section, we constructed different portfolios obtained by other ranking method such as TOPSIS, VIKOR, SAW, EDAS, COPRAS and PROMETHEE II and compared with constructed portfolio in the previous section. Table 4.3.8 shows the ranking of stocks evaluated by aforementioned methods. Table 4.3.9 displays the portfolios constructed by different methods.

To compare the performance of the different portfolios, the optimization problem (mentioned in subsection 3.2.2.3) is solved for all the portfolios. Obtained rank wise optimal ratios to the securities for different portfolios are presented Table 4.3.10. Table 4.3.11 displays total return, volatility or risk and Sharpe ratio obtained for different portfolios. From Table 4.3.11, it can notice that portfolios constructed by TOPSIS, VIKOR, SAW, EDAS and COPRAS gave almost same return with almost same tolerable risk. Portfolio constructed by PROMETHEE II gave 42.6% return of with 27.1% risk, its Sharpe ratio is 1.50. But the proposed portfolio outperforms overall portfolios with maximum Sharpe ratio 1.67 and gave maximum return of 50.1% with 25.9% minimum risk. Figure 4.3.4 and Figure 4.3.5 display the comparison graphically. Thus, the proposed portfolio is best for investment. The comparison analysis proves that the proposed method of portfolio construction provides diversification and gives more return over risk.

Table 4.3.8: Ranking of alternatives (stocks) by different methods

Alternatives	TOPSIS	VIKOR	SAW	EDAS	COPRAS	PROMETHEE II
A_1	20	20	20	20	20	20
A_2	8	6	5	5	4	11
A_3	10	9	3	7	3	13
A_4	6	11	6	15	12	9
A_5	9	7	10	4	6	6
A_6	18	19	13	19	17	19
A_7	13	15	14	17	18	10
A_8	14	13	15	11	11	12
A_9	5	3	7	3	5	3
A_{10}	17	17	4	16	13	17
A_{11}	7	5	12	6	8	7
A_{12}	19	18	18	18	19	18
A_{13}	3	4	9	8	7	4
A_{14}	2	2	2	1	2	1
A_{15}	4	8	11	10	10	5
A_{16}	16	16	19	14	16	16
A_{17}	11	12	16	13	14	15
A_{18}	15	14	17	12	15	14
A_{19}	12	10	8	9	9	8
A_{20}	1	1	1	2	1	2

Table 4.3.9: Portfolio constructed by different methods

Methods	Securities									
TOPSIS	A_{20}	A_{14}	A_{13}	A_{15}	A_{9}	A_{4}	A_{11}	A_{2}	A_{5}	A_{3}
VIKOR	A_{20}	A_{14}	A_{9}	A_{13}	A_{12}	A_{2}	A_{5}	A_{15}	A_{3}	A_{19}
SAW	A_{20}	A_{14}	A_{3}	A_{10}	A_{2}	A_{4}	A_{9}	A_{19}	A_{13}	A_{5}
EDAS	A_{14}	A_{20}	A_{9}	A_{5}	A_{2}	A_{11}	A_{3}	A_{13}	A_{19}	A_{15}
COPRAS	A_{20}	A_{14}	A_{3}	A_{2}	A_{10}	A_{5}	A_{13}	A_{11}	A_{19}	A_{15}
PROMETHEE II	A_{14}	A_{20}	A_{9}	A_{13}	A_{15}	A_{5}	A_{11}	A_{19}	A_{4}	A_{7}
PROPOSED METHOD	A_{14}	A_{9}	A_{13}	A_{5}	A_{11}	A_{4}	A_{3}	A_{16}	A_{10}	A_{1}

Table 4.3.10: Rank wise weights allocation to the stocks of different portfolios

Methods	Weights									
TOPSIS	0.271	0.271	0.261	0.261	0.117	0.013	0.013	0.013	0.013	0.013
VIKOR	0.364	0.353	0.059	0.059	0.0376	0.0376	0.0376	0.0376	0.0376	0.0
SAW	0.406	0.366	0.028	0.028	0.028	0.028	0.028	0.028	0.028	0.028
EDAS	0.315	0.315	0.061	0.061	0.061	0.061	0.061	0.061	0.0	0.0
COPRAS	0.329	0.274	0.066	0.066	0.066	0.066	0.066	0.066	0.0	0.0
PROMETHEE II	0.304	0.291	0.081	0.081	0.081	0.081	0.081	0.0	0.0	0.0
PROPOSED METHOD	0.369	0.124	0.124	0.124	0.124	0.027	0.027	0.027	0.027	0.027

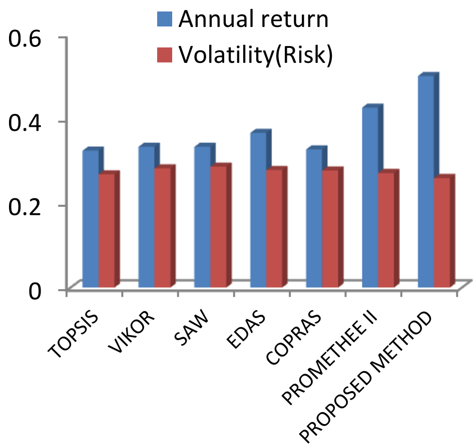


Figure 4.3.4: Annual return and volatility of portfolios obtained by different methods

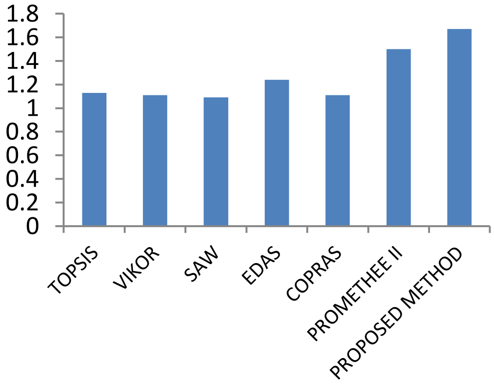


Figure 4.3.5: Sharpe ratio of portfolios obtained by different methods

Table 4.3.11: Comparison between different portfolios

Methods	Return	Volatility	Sharpe Ratio
TOPSIS	0.324	0.268	1.13
VIKOR	0.333	0.282	1.11
SAW	0.333	0.286	1.09
EDAS	0.366	0.278	1.24
COPRAS	0.327	0.277	1.11
PROMETHEE II	0.426	0.271	1.50
PROPOSED METHOD	**0.501**	**0.259**	**1.67**

4.3.5.2 Comparing rank based and rank irrelevant portfolio

In the proposed model investment ratio are assigned to the assets based on the ranking order i.e. higher ratio is assigned to higher rank asset. We solve the proposed optimization problem without rank constraint and obtained rank irrelevant weights of the securities (alike the procedure of (**Bhattacharyya *et al.*, 2014, Markowitz, 1952**). Table 4.3.12 presents the rank irrelevant ratio allocation of the proposed portfolio. Table 4.3.13 shows comparison between performances of rank based and rank irrelevant portfolio and Figure 4.3.6 graphically presents the comparison.

Table 4.3.12: Rank irrelevant weight allocation of proposed portfolio

	A_{14}	A_9	A_{13}	A_5	A_{11}	A_4	A_3	A_{16}	A_{10}	A_1
Rank irrelevant	0.095	0.075	0.095	0.179	0.169	0.123	0.055	0.082	0.094	0.034

The comparison clears that the rank based portfolio gives more return with less risk as compared to rank irrelevant portfolio. Its Sharpe ratio is also higher than rank irrelevant portfolio. This certifies the rootedness and effectiveness of the proposed portfolio model and ranking system.

Table 4.3.13: Rank relevant and rank irrelevant performance of proposed portfolio

	Return	Risk	Sharpe Ratio
Rank based	**0.501**	**0.259**	**1.67**
Rank irrelevant	0.427	0.295	1.42

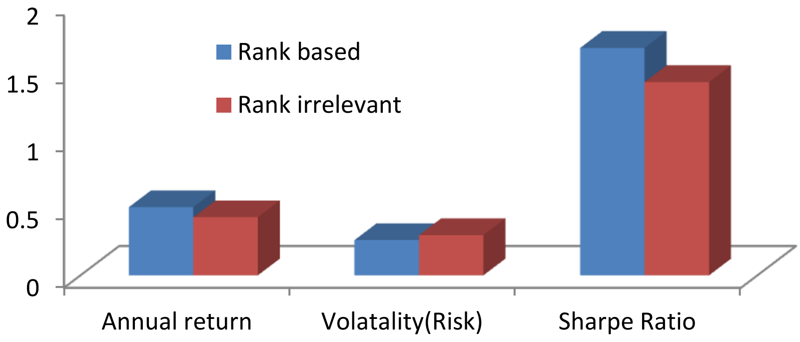


Figure 4.3.6: Rank based vs rank irrelevant portfolio

4.3.5.3 Comparison with existing models

In this section, we compared the portfolios obtained by proposed method with some recent models developed by **(Thakur *et al.*, 2018, Naveenan, 2019, Narang *et al.*, 2021 and Narang *et al.*, 2022)**. Table 4.3.14 displays the comparison between the annual returns obtained by the other models and proposed model. The comparison shows that proposed portfolio gain maximum profit.

Table 4.3.14: Comparison of proposed model with other models

Models	Thakur *et al.* (2018)	Naveenan (2019)	Narang *et al.* (2021)	Narang *et al.* (2022)	Proposed method
Year	2016	2019	2020	2020	2020
Annual return	0.1301	0.17	0.2361	0.1672	0.501

4.4 Model [4]: A portfolio construction model based on sector analysis using Dempster-Shafer evidence theory and Granger causal network

The proposed model has been illustrated with the help of a case study on NSE sectors. NSE comprises various sectors, for this study, following 12 major sectors are taken: Auto, Bank, Consumer durables, Financial services, Fmcg, Healthcare, IT, Media, Metal, Oil & Gas, Pharma, and Reality.

4.4.1 D-S theory for ranking sectors of NSE

Four breadth indicators (ADX, RSI, SMA and SRS) are considered as evidences for sector analysis. The first three evidences for each sector are collected every six months for the financial years 2016 to 2020 from the website https://www.definedge.com/tradepoint-web/. The SRS indicator is determined based on its performances in the last six months of the year 2020. To ensure independency between the evidences as required to apply the Dempster-Shafer theory, the evidences are examined for different time frequencies (for example, ADX is calculated using a moving window of 30 days, RSI for 14 days, SMA for 200 days, and SRS for 6 months). For more clarification, calculations for Auto sector are demonstrated. Table 4.4.1 shows the value of breadth indicators for every six months from the year 2016 to 2020 for Auto sector. Note the observation of Jan-Jun 2018 in Table 4.4.1, it shows 80 % of stocks in the Auto sector were in a strong upward trend, 53.33 % of stocks were in strong momentum (above average RSI line), and 73.33 % of stocks were trading over 200 days simple moving average. Table 4.4.2 shows a matrix of the Auto sector's relative strength in comparison to the other sectors. In the matrix, a value of 1 indicates that the Auto sector performed better than another sector, while a value of 0 indicates that the Auto sector performed poor. In the last six months of 2020, the Auto sector outperforms three other sectors.

Table 4.4.1: First three evidences for Auto sector

Evidences	Jan-Jun 2016	Jul-Dec 2016	Jan-Jun 2017	Jul-Dec 2017	Jan-Jun 2018	Jul-Dec 2018	Jan-Jun 2019	Jul-Dec 2019	Jan-Jun 2020	Jul-Dec 2020
ADX	73.33	86.66	93.33	86.66	80.00	60.00	33.33	13.33	40.00	20.00
RSI	0.00	60.00	53.33	66.66	53.33	40.00	0.00	6.66	86.66	86.66
SMA	66.66	73.33	73.33	73.33	73.33	80	73.33	73.33	73.33	80.00

Table 4.4.2: Relative strength of Auto sector

	Con. dura.	Healthcare	Oil& Gas	Bank	Fmcg	Fin. Serv.	It	Media	Metal	Pharma	Realty
Auto	0	1	0	0	1	0	0	0	0	1	0

The data collected by ADX and RSI breadth indicators were found to be randomly distributed. Experts suggested giving more weight and significance to the recent behavior of the sector. Therefore, we computed exponential moving average of data collected by ADX and RSI breadth indicators. Thus, value (in percent) such obtained manifest an average portion of the sector which is inclined to the recent trend as well as accounts the previous years' performance. This value has been set as the basic probability as an evidence favoring future potential in the sector.

On noticing the observations of the SMA breadth indicator, we found a high frequency of a particular value in the collected data. The higher frequency value denotes the numbers of stocks of the sector that are traded above than 200 days moving average value usually. Hence, the mode of the historical observations is considered for BPA.

For the last evidence SSR, BPA value is obtained by dividing "the number of sectors by which the given sector performed better" by "the total number of sectors on which the comparison performed."

Again explaining for the Auto sector, the exponential moving average of observations of ADX and RSI breadth indicators are computed as 24.86% and 78.08%, respectively. According to the ADX breadth indicator, BPA assigned as 0.2486 in the support of evidence towards the hypothesis "*sector is weak*". In the same manner, according to the RSI breadth indicator, the value of BPA is 0.7808 which advocates the hypothesis "*sector is strong*". The mode of the collected data by the SMA breadth indicator is found 73.33, so 0.7333 has been assigned as BPA which strongly recommends the hypothesis "*sector is strong*". Auto sector outperforms 3 sectors out of 11 sectors. So, the BPA equals 3/11 = 0.2727 has been assigned which favors the evidence for the hypothesis "*sector is weak*". Table 4.4.3 presents the basic probability assignment (BPA) for all sectors.

Table 4.4.3: Basic probability assignment (BPA) for all sectors

Sectors	ADX	RSI	SMA	SRS
AUTO	0.2486	0.7808	0.7333	0.27
BANK	0.2753	0.2489	0.50	0.45
CONSUMER DURABLES	0.6245	0.3700	0	0.81
FMCG	0.7031	0.4969	0.7031	0.18
IT	0.7991	0.8238	0.7991	0.72
PHARMA	0.7079	0.7502	0.7079	0
MEDIA	0.3664	0.5218	0.10	0.90
METAL	0.7880	0.7666	0.3333	0.36
REALITY	0.5330	0.3474	0	0.90
HEALTHCARE	0.7456	0.8050	0.55	0.09
FINANCE	0.3750	0.4198	0.40	0.63
OIL & GAS	0.4768	0.3551	0.5333	0.54

Proceeding with the Auto sector again, Table 4.4.4 shows the BPA for all four evidences favoring different hypotheses. Suppose, ADX is the first evidence, the mass $m_1(WS) = 0.24$ represents the belief towards hypothesis "*sector is weak*" and $m_1(\Theta) = 1 - 0.24 = 0.76$ is the belief on the rest of the hypotheses of the frame of discernment. Consider, RSI as second evidence with $m_2(SS) = 0.78$ and $m_2(\Theta) = 1 - 0.78 = 0.22$. Now, the masses of the first two evidences are combined which generates a new mass m_3 as shown in Table 4.4.5.

Table 4.4.4: BPA for Auto sector

Evidences	Hypotheses		
	WS	MS	SS
ADX	0.24		
RSI			0.78
SMA			0.73
SSR	0.27		

Table 4.4.5: Combining masses of first two evidences

	$m_1(WS) = 0.24$	$m_1(\Theta) = 0.76$
$m_2(SS) = 0.78$	$\varphi = 0.1872$	$SS = 0.5928$
$m_2(\Theta) = 0.22$	$WS = 0.0528$	Θ =0.1672

The new mass m_3 has been derived as:

$m_3(SS) = 0.5928/(1 - 0.1872)$= 0.729331

$m_3(WS) = 0.0528/(1 - 0.1872)$= 0.064961

$m_3(\Theta) = 0.1672/(1 - 0.1872)$= 0.205709

Now, consider the SMA as third evidence with $m_4(SS) = 0.73$ and $m_4(\Theta) = 0.27$ as given in Table 4.4.6. Again, the masses m_3 and m_4 are combined and generates a new mass m_5.

Table 4.4.6: Combining m_3 (new mass) and m_4 (mass of third evidence)

	$m_4(SS) = 0.73$	$m_4(\Theta) = 0.27$
$m_3(SS)$ =0.729331	$SS = 0.532411$	$SS = 0.196919$
$m_3(WS)$ =0.064961	$\varphi = 0.047421$	$WS = 0.017539$
$m_3(\Theta)$= 0.205709	$SS = 0.150167$	$\Theta = 0.055541$

Using Table 4.4.6 the new mass m_5 has been calculated as:

$m_5(SS) = \frac{0.532411+0.196919+0.150167}{1-0.047421}$ =0.923281

$m_5(WS) = \frac{0.017539}{1-0.047421}$ =0.018413

$m_5(\Theta) = \frac{0.055541}{1-0.047421}$ =0.058306

Now, consider the last evidence SSR having belief value $m_6(WS) = 0.27$ and $m_6(\Theta) = 0.73$, which is combined with the mass m_5 as given in Table 4.4.7.

Table 4.4.7: Combining m_5 (new mass) and m_6 (mass of four evidence)

	$m_6(WS) = 0.27$	$m_6(\Theta) = 0.73$
$m_5(SS)$ =0.923281	$\varphi = 0.249286$	$SS = 0.673995$
$m_5(WS)$ =0.018413	$WS = 0.004971$	$WS = 0.013441$
$m_5(\Theta)$ =0.058306	$WS = 0.015743$	$\Theta = 0.042564$

On combining m_5 and m_6, the new mass m_7 is generated as:

$$m_7(SS) = \frac{0.673995}{1-0.249286} = 0.897806$$

$$m_7(WS) = \frac{0.004971+0.013441+0.015743}{1-0.249286} = 0.045497$$

$$m_7(\Theta) = \frac{0.042564}{1-0.249286} = 0.056697$$

In the same manner, evidences of each sector are combined and the final masses of the hypotheses are calculated. Table 4.4.8 displays the final masses of all sectors. The sectors are ranked based on the mass value of the hypothesis "*sector is strong*". From the Table 4.4.8 we can notice that the It sector is ranked at the top and believed as strongest among all sectors. Six sectors: Auto, Consumer Durables, IT, Pharma, Metal, and Healthcare are concluded as strong sectors of NSE because of having more than 80% belief in the hypothesis "*sector is strong*".

Table 4.4.8: Ranking of the sectors

Sectors	**WS**	**MS**	**SS**	**Θ**	**Ranking**
Auto	0.045	0.000	0.897	0.056	2
Bank	0.180	0.593	0.000	0.225	12
Consumer durables	0.037	0.072	0.825	0.064	6
Fmcg	0.035	0.425	0.377	0.161	9
IT	0.007	0.000	0.982	0.010	1
Pharma	0.000	0.083	0.847	0.068	5
Media	0.062	0.091	0.761	0.084	8
Metal	0.065	0.000	0.884	0.049	4
Reality	0.044	0.096	0.773	0.085	7
Healthcare	0.004	0.059	0.887	0.048	3
Finance	0.326	0.137	0.337	0.198	10
Oil & Gas	0.058	0.833	0.000	0.107	11

4.4.2 Diversification in the strong sectors of NSE

This section proceeds with the top six strong sectors and discovers relations among them to achieve diversification. The relationship between the price movements of the strong sectors is determined by using 10 years daily closing prices i.e. real-time series data (from the year 2011-2020). The data is taken from https://in.finance.yahoo.com/ and denoised it before doing further analysis.

Kalman filter is simulated for every time series providing initial state same as in the real data. For instance, the real and filtered data of Auto sector is displayed in Figure 4.4.1.

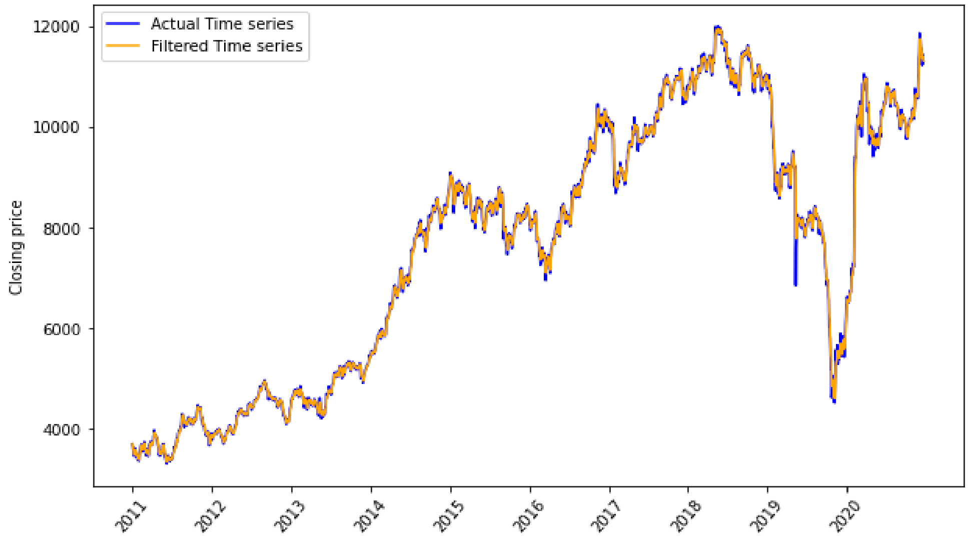


Figure 4.4.1: Actual and filtered closing prices of Auto Sector from in 10 years

The results of the VAR model for lag length selection are displayed in the Table 4.4.9. The appropriate lag length suggested by above mentioned criteria is 8 for all time series. Then, the pairwise co-integration test is performed for each sector-pair. The results of the statistical test determined at 5% level of significance. Table 4.4.10 presents the results of co-integration test displaying p-values for each sector-pair. P-value less than 0.05 indicates the co-integration between the two time series. After the pairwise co-integration test, VECM Granger causality test is performed for co-integrated pairs and VAR Granger causality test for other pairs. The test is done for 5% level of significance. P-value less than 0.05 rejects the null hypothesis and advocates the dependency of time series Y_t on the lagged values of X_t and Y_t i.e. Y_t is granger caused by X_t. Table 4.4.11 depicts the result of Granger causality test. Values referring no causal relations are highlighted by the bold character.

Figure 4.4.2 represents the Granger causality network among the sectors. In the network, each vertex represents a sector and a directed edge from sector A to another sector B interpreted as A Granger causes B. We target to find a sub-network of Figure 4.4.2 having the least causal relations or no causal relations while considering the ranking order of sectors too. Again from Figure 4.4.2, bi-directional causality exists among Pharma-Healthcare, Auto- Consumer durables, and Metal-Consumer durables is observed. Investors must avoid investing in both sectors of bi-directional pairs at the same time. As It sector has been preferred as the best sector to invest and It doesn't have causal relationship with Healthcare and Metal. Healthcare and Metal also don't cause each other. Therefore, we ended up with three sectors It, Healthcare, and Metal that are having no dependency among each other as represented in the sub-network in Figure 4.4.3.

Table 4.4.9: VAR length lag selection criteria

Lag	LL	LR	FPE	AIC	SC
0	-193836.0	NA	2.16e+64	182.1879	182.2199
1	-138490.5	110014.7	6.34e+41	130.3069	130.7220
2	-138041.8	886.9039	4.76e+41	130.0205	130.8188
3	-137863.8	379.8937	4.61e+41	129.9885	131.1700
4	-137714.9	290.8983	4.59e+41	129.9839	131.5487
5	-137564.0	293.0363	4.56e+41	129.9775	131.9254
6	-137412.0	293.6248	4.53e+41	129.9699	132.3011
7	-137287.7	238.5845	4.61e+41	129.9885	132.7029
8	-137117.9	324.1826	4.50e+41*	129.9642*	133.0618
9	-136999.0	225.7315	4.61e+41	129.9877	133.4685
10	-136842.6	295.0250	4.56e+41	129.9761	133.8401
11	-136718.8	232.0527	4.65e+41	129.9951	134.2423
12	-136602.6	216.4558	4.77e+41	130.0213	134.6517

Table 4.4.10: Pairwise co-integration between selected sectors

	Auto	**Consumer Durables**	**IT**	**Pharma**	**Metal**	**Healthcare**
Auto	-	**0.048***	0.686	0.375	0.896	0.738
Consumer Durables		-	0.721	0.921	**0.020***	0.951
IT			-	0.673	0.173	0.812
Pharma				-	0.743	0.801
Metal					-	0.247
Healthcare						-

Table 4.4.11 Granger causality between selected sectors

	Auto	**Consumer Durables**	**IT**	**Pharma**	**Metal**	**Healthcare**
Auto	-	0.0003	0.000	**0.1979***	**0.3058***	0.0243
Consumer Durables	0.000	-	**0.5682***	**0.0614***	0.0020	0.0001
IT	**0.4334***	0.000	-	**0.2814***	**0.3746***	**0.1846***
Pharma	**0.3003***	**0.5706***	0.003	-	**0.7681***	0.0476
Metal	**0.3705***	0.0184	**0.1912***	**0.1498***	-	**0.0705***
Healthcare	**0.2121***	**0.1060***	**0.4947***	0.0342	**0.0832***	-

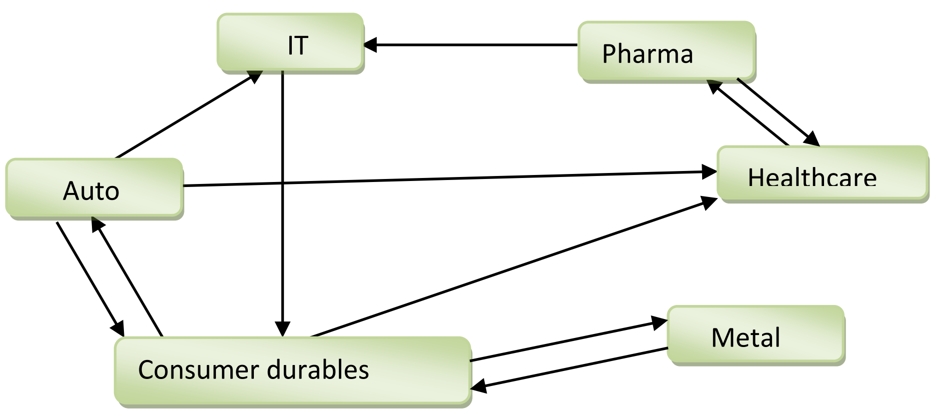


Figure 4.4.2: Causal relations among strong sectors

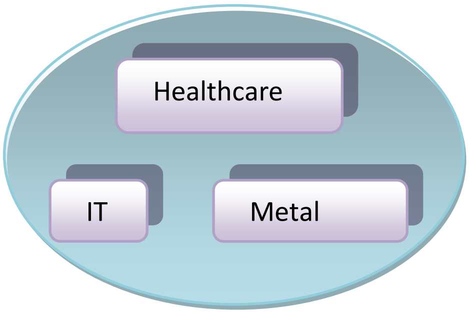


Figure 4.4.3: Strong sectors having no causal relationships

4.4.3 Portfolio construction

The share of each sector in the portfolio, i.e. the total number of stocks to be taken from that sector, is calculated as follows:

$$\text{No. of stocks from IT sector} = \frac{0.982}{2.753} = 0.3567 \approx 4$$

$$\text{No. of stocks from Healthcare sector} = \frac{0.887}{2.753} = 0.3221 \approx 3$$

$$\text{No. of stocks from Metal sector} = \frac{0.884}{2.753} = 0.3211 \approx 3$$

After determining investment portion, the leading stocks of a particular sector are selected. For this, the top listed stocks of the sector are analyzed based on their performances in the last 5 financial years (from 2016 to 2020). For example, there are 10 big IT companies listed in the Nifty IT; Table 4.4.12 shows the percentage change in the closing price of the companies every six months from the year 2016 to 2020. It can be noticed from the Table 4.4.12 that the IT sector is in profit since 2017. The stock prices of HCL and TCS companies had been grown in 8 observations out of 10 which are the highest time compared with the rest of the companies. After that INFOSYS and MPHASIS performed well and remain 7 times in profit out of 10. Hence, we selected 4 companies HCL, TCS, MPHASIS, and INFOSYS from the IT sector. The rating of each company in the IT sector is graphically presented in Figure 4.4.4. In the similar manner, companies from Healthcare sector are selected. Healthcare sector of NSE comprises 20 big companies. The bar graph in Figure 4.4.5 depicts the rating of 20 companies of healthcare sector. Figure 4.4.5 shows highest rating for LALPATH and then same rating to BIOCON, DIVISLAB, IPCALAB and TORNTPHARM. We choose LALPATH as it is rated at top and two companies DIVISLAB and TORNTPHARM having less volatility than other two companies.

Again, applying the same procedure for the Metal sector we choose three companies; ADANIENT, JSWSTEEL and RATNAMANI. The bar graph in Figure 4.4.6 depicts the rating of 15 companies of the Metal sector.

Thus, the composition of the portfolio is as follows:

IT Sector

- HCL Technologies Limited (HCLTECH)
- Tata Consultancy services (TCS)
- Infosys Limited (INFY)
- Mphasis Limited (MPHASIS)

Healthcare Sector

- Dr. Lal PathLabs Limited (LALPATH)
- Divi's Laboratories Limited (DIVISLAB)
- Torrent Pharmaceuticals Limited (TORNTPHARM)

Metal Sector

- Adani Enterprises Limited (ADANIENT)
- JSW Steel Limited (JSWSTEEL)
- Ratnamani metals and tubes (RATNAMANI)

Table 4.4.12: Change in closing price of the companies in IT sector

Stocks	Jan-Jun 2016	Jul-Dec 2016	Jan-Jun 2017	Jul-Dec 2017	Jan-Jun 2018	Jul-Dec 2018	Jan-Jun 2019	Jul-Dec 2019	Jan-Jun 2020	Jul-Dec 2020
NIFTY IT	-2.87	-9.76	9.21	20.74	12.33	6.25	0.78	3.35	11.94	34.19
COFORGE	-18.42	-8.88	25.7	66.08	43.3	6.83	-8.18	61.39	-0.84	40.46
HCLTECH	-13.03	7.73	10.08	10.49	-2.17	4.16	2.92	14.33	19.24	34.17
INFY	-7.82	-13.53	8.9	13.75	18.68	9.82	5.88	-2.23	24.49	30.00
LTTS	-	-	-9.14	58.89	21.28	8.08	-9.10	16.53	-10.00	54.8
LTI	-	-	14.28	66.52	47.15	-5.98	-14.04	28.18	24.56	0.51
MINDTREE	-21.38	-22.15	5.98	61.27	21.08	-4.1	-19.88	23.96	22.02	53.25
MPHASIS	18.58	3.39	8.11	48.01	28.71	-12.95	-6.36	-0.96	24.53	32.99
TCS	9.53	-14.87	11.75	24.9	24.68	3.81	9.51	-5.74	9.73	25.48
TECHM	-2.68	-7.33	-14.67	58.78	11.12	7.51	-13.07	25.2	-14.43	42.77
WIPRO	-3.15	-15.97	25.96	5.63	-9.29	33.57	-4.19	-10.74	18.64	37.48

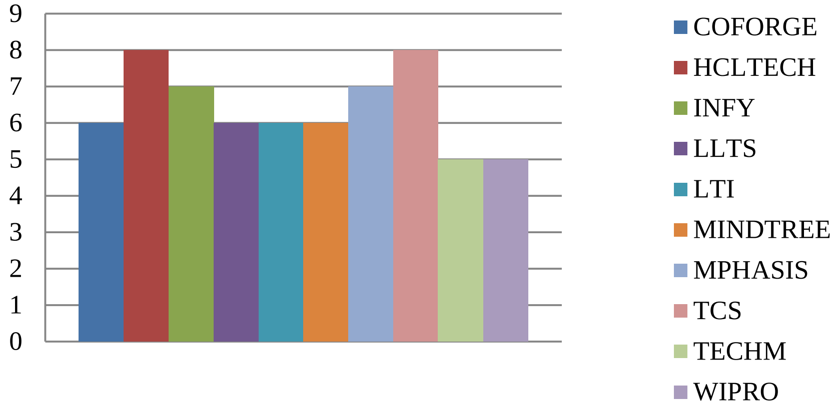


Figure 4.4.4: Rating of IT companies

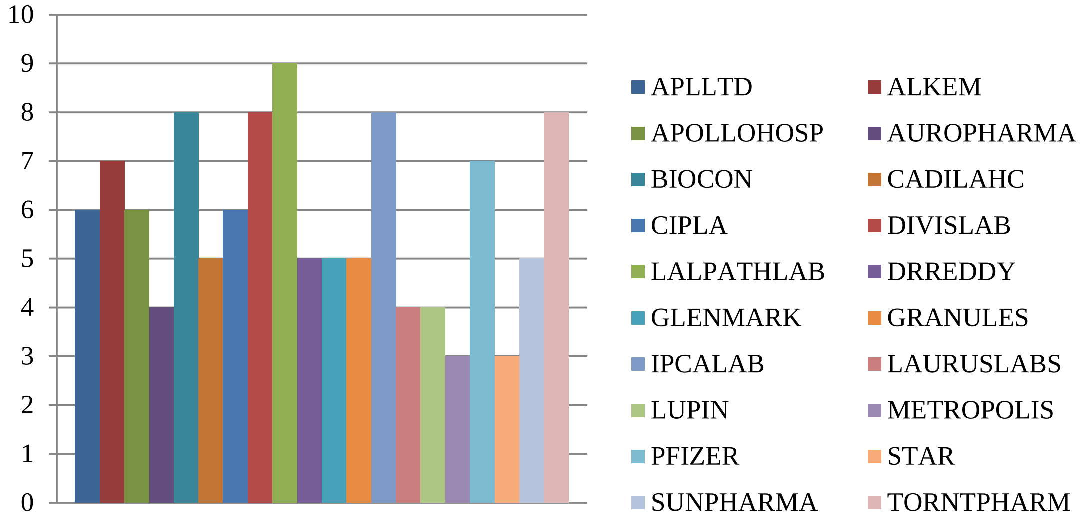


Figure 4.4.5: Rating of Healthcare companies

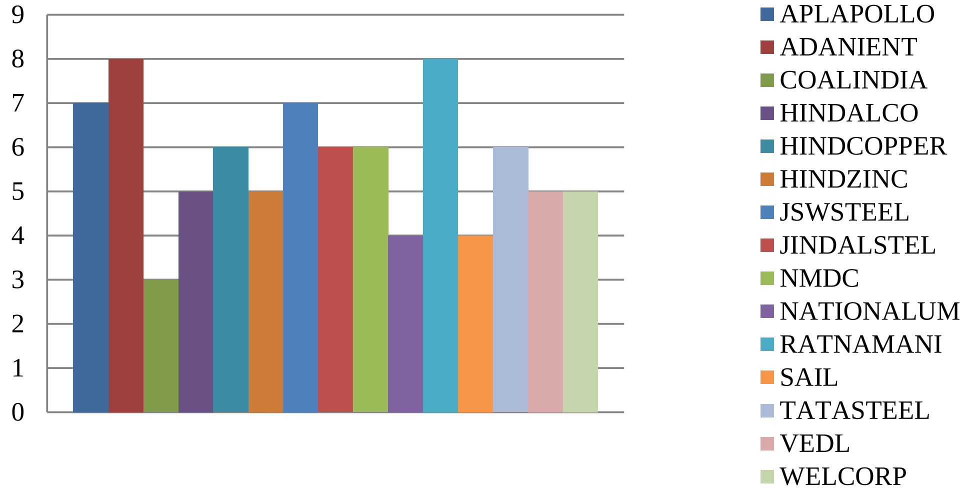


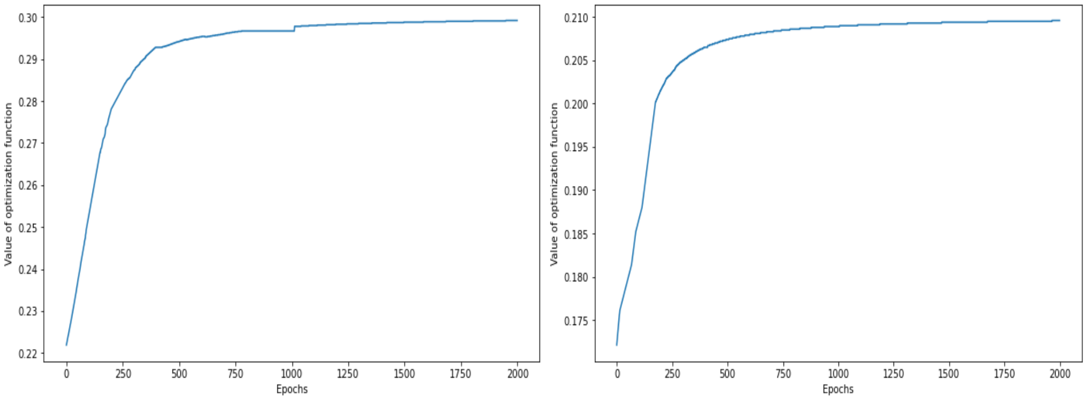
Figure 4.4.6: Rating of Metal companies

4.4.4 Portfolio optimization

The optimization of portfolio by deep recurrent neural network is executed in Google Colaboratory for half-year and annual investment in the year 2021. For half-year investment, daily closing prices and daily returns of securities from (01-Jan-2021to 30-Jun-2021) and for annual investment, daily closing prices and daily returns from (01-Jan-2021to 31-Dec-2021) are taken. The past 20 days observations are fed as single input in the network. Then three hidden layers of simple recurrent neural network having 20 nodes each are set. The same configuration of the neural network is adopted for both investments. The adam optimizer is used for training the network. In mini-batch size 32 it iterated up to 2000 epochs. The optimal solution i.e. allocated ratios obtained after 2000 iteration for both (half-year and annual) investment is given in Table 4.4.13. The maximum value of the objective function converged to 0.29 and 0.21 for 6 months and annual data, respectively. The maximum return is found as 0.6731(in six months) and 0.8902 (annually) with minimum volatility as 0.1970 and 0.2503, respectively. Figure 4.4.7 clarifies the effectiveness of the implemented deep recurrent neural network for the optimizing proposed portfolio.

Table 4.4.13: Ratio allocation to the components of portfolio

Stocks	ADANIENT	DIVISLAB	HCLTECH	INFY	JSWSTEEL	LALPATH	MPHASIS	RATNAM	TCS	TORNTPHARM
Half-year Return	1.219	0.138	0.051	0.229	0.609	0.323	0.340	0.265	0.145	0.061
Ratio allocation	0.313	0.006	0.016	0.017	0.288	0.11	0.201	0.009	0.017	0.022
Annual return	1.383	0.220	0.368	0.439	0.585	0.520	0.863	0.240	0.271	0.168
Ratio allocation	0.248	0.002	0.004	0.036	0.055	0.192	0.460	0.0001	0.001	0.001



(a) For half-year investment **(b)** For annual investment

Figure 4.4.7: Convergence of the objective function

4.4.5 Effectiveness of the proposed model

This model is focused on sector-based portfolio construction by ranking sectors. To verify the reliability of the ranking system we checked the performance of the strong sectors recommended by the ranking for next the year 2021. Table 4.4.14 presents half-year and annual expected returns of the strong sectors. All sectors are found to be in an uptrend in the year 2021. In six months all sectors show more than 10% growth and Metal outperforming others with 60% expected return. Annually, all sectors show more than 15% growth except Pharma. Metal and IT are found to be highly rewarding showing more than 50% growth. Metal and IT remained leading sectors throughout the year. Hence, this proves the applicability of proposed model for selection of sectors.

All securities of the proposed portfolio also gave positive returns. Massive growth in the prices of the stocks from Metal sector is noticed. Stocks of the IT sector and Healthcare sector also performed very well. If we assign equal weightage to each assets of the portfolio, it gives 39.29% return in half year and 50.75% annually. This proves the effectiveness of the proposed sector-based portfolio selection model.

Table 4.4.14: Expected returns of strong sectors

Sectors	Half-year	Annual
Auto	0.143	0.179
Consumer Durables	0.154	0.463
IT	0.193	0.582
Pharma	0.100	0.094
Metal	0.601	0.694
Healthcare	0.152	0.175

4.4.6 Comparative analysis

4.4.6.1 Comparison of proposed portfolio optimization with other optimizations algorithms

In this section, we compared the results obtained by the proposed optimization technique with other classical methods of portfolio optimization. The comparison is done with three popular methods; the Markowitz model, critical line algorithm (CLA), and hierarchical risk parity (HRP). Markowitz model is a mean-variance model using efficient frontier to get three types of optimal portfolio allocations (for minimum volatility, maximum Sharpe ratio, or maximum return under tolerable risk). Critical line algorithm (CLA) has been specially designed for optimizing portfolios having inequality constraints; this algorithm guarantees to provide an exact solution after a given number of iteration (For more details on CLA see **(Niedermayer and Niedermayer, 2010).** Hierarchical risk parity (HRP) is a novel method of portfolio optimization introduced by **Lopez de Prado (2016)**. HRP uses graph theory and machine learning techniques to construct a diversified portfolio (For more details on HRP details see **(Lopez de Prado, 2016).** We implemented all three optimization methods for half-year as well as annual investment in the year 2021. Markowitz model for minimum volatility is implemented. The results obtained by all three optimizers are given in Table 4.4.15. It is clear from the Table 4.4.15 that proposed model gain maximum return and the highest Sharpe ratio as compared to other optimizers for both investments. Figure 4.4.8 and Figure 4.4.9 are graphical illustrations of the comparison. Hence, it proves the effectiveness of proposed optimizer.

Table 4.4.15: Comparison of the proposed portfolio optimization model with other models

For Six month			
	Return	Volatility	Sharpe Ratio
Proposed model	67.31%	19.10%	3.29
Markowitz	43.5%	15.7%	2.64
HRP	49.8%	16.6%	2.88
CLA	18.4%	10.9%	1.51
Annual			
	Return	Volatility	Sharpe Ratio
Proposed model	89.02%	25.03%	3.26
Markowitz	30.4%	14.3%	1.99
HRP	41.3%	15.4%	2.55
CLA	14.0%	14.3%	0.84

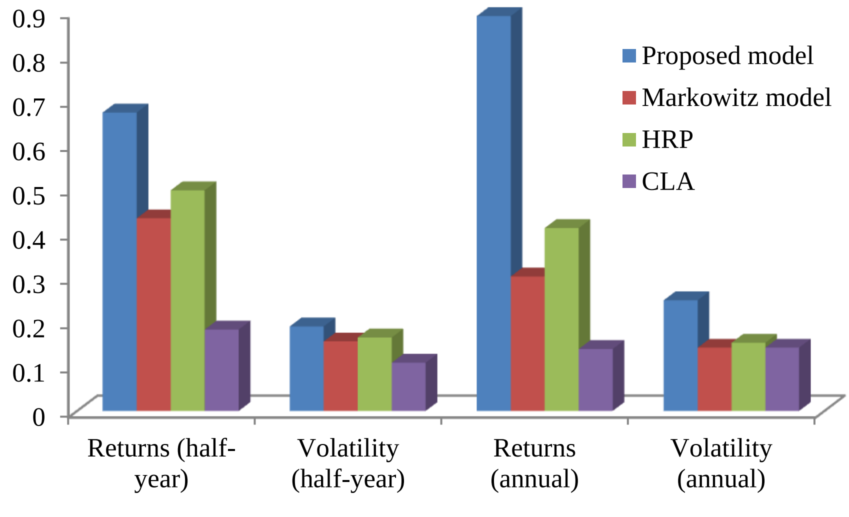


Figure 4.4.8: Comparison between returns and volatility among different optimization techniques

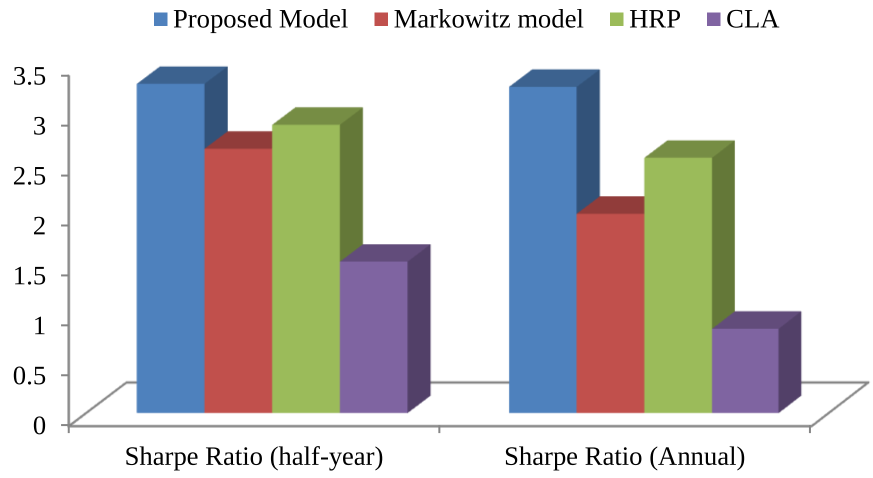


Figure 4.4.9: Comparison between Sharpe ratios obtained by different optimization techniques

4.4.6.2 Comparison with existing models

In this section, the portfolio obtained by proposed model is compared with some recent models. All mention models are based on ranking of stocks. Table 4.4.16 displays the comparison between the annual returns obtained by the other models and proposed model. Even without optimization, the proposed portfolio generates a higher return than other models. The comparison verifies that the proposed sector-based portfolio construction approach is more effective than other models.

Table 4.4.16: Comparison of proposed model with other models

Models	**Thakur *et al.* (2018)**	**Naveenan (2019)**	**Narang *et al.* (2021)**	**Narang *et al.* (2022)**	**Proposed portfolio (without optimization)**	**Proposed portfolio (after optimization)**
Year	2016	2019	2020	2020	2020	2020
Annual return	0.1301	0.17	0.2361	0.1672	**0.5075**	**0.8902**

4.5 Model [5]: Stock portfolio selection hybridizing fuzzy base-criterion method and evidence theory in triangular fuzzy environment

To illustrate this model for common Indian investor we employed the proposed model on following 30 companies of NSE:

1. Adani Enterprises Ltd.
2. Apollo Hos. Ent. Ltd.
3. Ashok Leyland Ltd.
4. Aurobindo Pharma Ltd.
5. UltraTech Cement Ltd.
6. Dabur India Ltd.
7. Divi’s Laboratories Ltd.
8. Dr. Reddy’s Lab. Ltd.
9. GAIL (India) Ltd.
10. HCL Technologies Ltd.
11. Hindutan Unilever Ltd.
12. Indiabulls Real Est. Ltd.
13. Infosys Ltd.
14. Jindal Steel & power Ltd.
15. JSW Steel Ltd.
16. Jubilant FoodWorks Ltd.
17. Dr. Lal PathLabs Ltd.
18. Maruti Suzuki India Ltd.
19. Mphasis Ltd.
20. Nestle India Ltd.
21. ONGC Ltd.
22. Pidilite Industries Ltd.
23. Ratnamani M. & T. Ltd.
24. Reliance Industries Ltd.
25. Tata Motors Ltd.
26. Tata Steel Ltd.
27. TCS Ltd.
28. Titan Company Ltd.
29. Torrent Phar. Ltd.
30. Zee Entertain. Ent. Ltd.

4.5.1 Selection of criteria

For novice investor: As previously stated, return and risk are the two prior factors for neophyte investor. Different approaches are used to calculate a stock's return. In this work, we employ the basic formula for calculating return as follows:

$$Return\ of\ a\ stock = \frac{Final\ stock\ price - Initial\ stock\ price}{Initial\ stock\ price} * 100$$

There are various methods as well for calculating risk of a stock. Here, we used standard deviation, which is the most popular way of measuring risk. It determines the spread of stock prices from its average price. High standard deviation refers to wide movement in stock prices and the stock is considered risky. The annualized standard deviation for daily returns is calculated as follows:

$$annualized\ standard\ deviation = Standard\ deviation\ (r_1, r_2, r_3 \ldots r_n) * \sqrt{n}$$

here, n is the number of trading days in a year and r_t are the daily returns.

For domain expert: Due to the prevalence of multi-dimensional uncertainties, domain experts believe that there is no one-size-fits-all strategy for stock analysis. However, some successful investors, such as George Soros and Warren Buffet, have suggested to research about some fundamentals of a company financials and operational statistics while making investment decisions. Fundamental analysis for stocks considers revenues, risk, debt level, future growth, returns, profit margins, and other factors to identify a company's underlying value and future development potential. The stocks' data used in this study is collected from the website https://web.stockedge.com , which provides fundamental indicators in six different categories, including return ratios, growth ratios, efficiency ratios, cash flow ratios, solvency ratios, and valuation ratios.

To select the critical measures among six measures we applied the FDM. We asked ten domain experts to value the importance of six measures for stock analysis. The geometric mean calculated for six measures are given in Table 4.5.1. The four measures with a geometric mean greater than 0.6 are considered as critical measures of a company for stock analysis. These four measures are return ratios, efficiency ratios, solvency ratios and valuation ratios. These measures further comprise different indicators of each category. For example, different return ratios for different perspective like Return on Equity, Return on Capital Employment and Returns on Assets etc. are used. To choose one most effective indicators among all indicators of each selected measure, again FDM is employed. The indicators of different measures and their geometric mean are presented in Table 4.5.2.

Table 4.5.1: Fundamental measures with geometric mean

Measures	Geometric mean
Return Ratios	0.86
Growth Ratios	0.54
Efficiency Ratios	0.78
Cash flow Ratios	0.51
Solvency Ratios	0.65
Valuation Ratios	0.71

Table 4.5.2: Indicators of fundamental measures with geometric mean

Return Ratios	G.M.	Efficiency Ratios	G.M.	Solvency Ratios	G.M.	Valuation Ratios	G.M.
Return on Equity	0.81	EBIT Margin	0.77	Interest Coverage Ratio	0.64	Price to Earnings Ratio	0.82
Return on Capital Employed	0.62	EBITDA Margin	0.85				
		PAT Margin	0.64	Total Debt to Assets	0.75	Price to Book Ratio	0.87
Return on Assets	0.78	Earnings per Share	0.80				
		Book Value Per Share	0.74	Total Debt to Equity Ratio	0.86	Price to Cash Flow Ratio	0.78
		Asset Turnover Ratio	0.60				
		Inventory Turnover Ratio	0.54	Current Ratio	0.70	EV to EBIT Ratio	0.56
		Debtors Turnover Ratio	0.56	Quick Ratio	0.54		
		Days of Receivables	0.45			EV to EBITDA Ratio	0.54
		Days of Payables	0.20				
		Days of Inventory	0.24			Market Cap to	0.45
		Working Capital Days	0.21			Sales Ratio	0.46
		Dividend Payout Ratio	0.48				
		Effective Tax Rate	0.51				

Hence the following ratios are chosen for working further: ROE from returns ratios, EBITDA Margin from efficiency ratios, Total Debt to Equity Ratio from solvency ratios, and Price to Book Ratio from valuation ratios. The significance of final four factors is discussed below:

1. Return on Equity (ROE): Return on equity is one of the important return ratios for fundamental analysis of a stock. It calculates the profit or return per share holder's equity. The return on equity (ROE) is a measure of a company's profitability and efficiency in generating profits. Investors prefer a company with a high ROE because it means investors will obtain a higher return on their investment. ROE of a company is defined as:

$$Return\ on\ Equity\ (ROE) = \frac{Company's\ net\ income}{Average\ shareholders'\ equity}$$

2. EBITDA Margin (EBITDA): It is an efficiency metric that assesses a company's operational profitability. EBITDA is an abbreviation for earnings before interest, taxes, depreciation, and amortization. EBITDA eliminates all of those variables to focus on the most important ones: operating profitability and cash flow. The higher EBITDA margin of a company indicates the lower its operating expenses are in relation to total revenue. EBITDA Margin of a company is calculated as follows:

$$EBITDA\ Margin = \frac{(Earnings\ before\ interest\ and\ tax\ +\ Depreciation\ +\ Amortization)}{Total\ revenue}$$

3. Total Debt to Equity Ratio (D/E): Total Debt to Equity ratio is a solvency ratio of a company is calculated by dividing the debt of the company to the share holders' equity (share holders fund). It is most commonly used to determine how much debt a company is taking on in order to leverage its resources. It also reflects the ability of a company to outstand of all the debts if the market downturn eventually. The formula of calculating Total Debt to Equity ratio is:

$$Total\ Debt\ to\ Equity\ Ratio\ (D/E) = \frac{Total\ liabilities}{Total\ shareholders'\ equity}$$

4. Price to Book Value ratio (P/B): The P/B ratio measures how much market players value a company's stock in comparison to its book value. The book

value of an asset is the same as its carrying value on the balance sheet, and companies determine it by subtracting the asset's cumulative amortization. In other words, if a company liquidated all of its assets and paid off all of its debt, the amount left would be the company's book value, which reflects the company's valuation. The P/B is calculated as:

$$Price\ to\ Book\ Value\ ratio\ (P/B) = \frac{Market\ value\ per\ share}{Book\ value\ per\ share}$$

4.5.2 Weight determination of attributes

The novice investors attributes (risk and return) are considered equally important, therefore both are given equal i.e. 0.5 weightage in this study. The experts compared the attributes on triangular fuzzy scale given in Table 3.5.1 and the comparison is presented in Table 4.5.3. Majority of experts suggested ROE as the most important factor among all and compared importance of ROE with other factors. By the consensus of all experts, ROE is considered moderate important to EBITDA Margin, very strong important as compared to D/E and strong important to price to book value ratio. Using the comparison results, the fuzzy weights of all attributes are obtained by fuzzy BCM method. The final weights of the attributes are obtained by calculating GMIR which are presented in Table 4.5.4.

Table 4.5.3: Comparison of experts attributes on triangular fuzzy scale

	EBITDA Margin	D/E	P/B
	Moderate Important	Very Strong Important	Strong Important
Return on Equity (Base criterion)	(1,1,3/2)	(3/2,2,2/5)	(1,3/2,2)

Table 4.5.4: Weights of attributes

Decision makers	Criteria	l	m	u	w
Novice Investor	Return	0.50	0.50	0.50	0.50
	Risk	0.50	0.50	0.50	0.50
Domain Expert	ROE	0.3183	0.3183	0.3304	0.32
	EBIDTA Margin	0.2202	0.3183	0.3183	0.30
	D/E	0.1321	0.1591	0.2122	0.16
	P/B	0.1652	0.2122	0.3183	0.22

4.5.3 Collecting evidences and basic probability assignment (BPA)

The experts suggested the description of the attributes' frame of discernment which is given in Table 4.5.5. For example, that the returns of a company in a particular year below or equal to 0 is considered as an evidence supporting the preposition (BI) i.e. *"bad for investment"*, return between the interval (0,10] advocates the preposition (AI) i.e. *"average for investment"* and return above 10 is taken as the evidence in support of the preposition *"good for investment"*. Similarly the descriptions of all attributes are given. We collected data of thirty companies of NSE for 9 the financial years 2012-2020. For calculating returns and risk as per investors concern, stocks' daily closing price data is obtained for 9 the financial years 2012-2020 from https://finance.yahoo.com/ . Hence, in this manner we get 9 observations for each attribute for each company. The number of observations supporting each proposition {BI, AI, GI} is noted then divided by 9. The obtained value is taken as the BPA for the preposition presented in frame of discernment. For instance, the data collected for Tata Steel is presented in Table 4.5.6. Tata Steel gave negative returns in 5 years (2013, 2014, 2015, 2018, and 2019), so 5/9= 0.555 is taken as the BPA for the preposition (BI), i.e., *"bad for investment"*, none of the observation is noticed as the evidence supporting the preposition (AI), i.e., *"average for investment"* and 4 observations for the years (2012, 2016, 2017, 2020) are noticed referring more than 10% returns so 4/9= 0.444 is taken as the BPA for the preposition (GI), i.e., *"good for investment"*. In the same manner all BPA for frame of discernment for Tata Steel is calculated which are shown in Table 4.5.7.

Table 4.5.5: Description of attributes frame of discernment

<table>
<tr><th>Decision makers</th><th>Factors</th><th>BI</th><th>AI</th><th>GI</th></tr>
<tr><td rowspan="2">Novice Investor</td><td>Return</td><td>Below and equal to 0</td><td>Between (0, 10]</td><td>Above 10</td></tr>
<tr><td>Risk</td><td>Above 50</td><td>Between [30, 50]</td><td>Below 30</td></tr>
<tr><td rowspan="4">Domain Expert</td><td>ROE</td><td>Below 10</td><td>Between [10, 15]</td><td>Above 15</td></tr>
<tr><td>EBIDTA Margin</td><td>Below 8</td><td>Between [8, 10]</td><td>Above 10</td></tr>
<tr><td>DOE</td><td>Below 0.1 or Above 2</td><td>Between [0.1, 0.5) or Between (1.5, 2]</td><td>Between [0.5, 1.5]</td></tr>
<tr><td>P/B</td><td>Below 0.5 or Above 3.5</td><td>Between [0.5, 1) or Between (3, 3.5]</td><td>Between [1, 3]</td></tr>
</table>

Table 4.5.6: Historical data of Tata Steel Ltd.

Decision makers	Factors	2012	2013	2014	2015	2016	2017	2018	2019	2020
Novice Investor	Return	0.26	-0.03	-0.06	-0.36	0.52	0.80	-0.24	-0.08	0.37
	Risk	0.34	0.39	0.35	0.40	0.41	0.26	0.33	0.38	0.49
Domain Expert	ROE	12.85	-19.61	9.97	-11.29	-1.13	-11.87	38.91	14.20	1.43
	EBIDTA Margin	10.29	9.22	11.05	9.24	5.01	11.39	17.13	18.55	13.19
	DOE	1.36	1.92	1.93	2.49	2.01	2.44	1.52	1.46	1.58
	P/B	1.10	0.91	0.96	1.02	0.80	1.47	1.12	0.90	0.43

Table 4.5.7: Basic probability assignment for Tata Steel Ltd.

Decision makers	Factors	BI	AI	GI
Novice Investor	Return	0.555	0.000	0.444
	Risk	0.000	0.889	0.111
Domain Expert	ROE	0.667	0.222	0.111
	EBIDTA Margin	0.111	0.222	0.667
	DOE	0.333	0.444	0.222
	P/B	0.111	0.444	0.444

4.5.4 Combining evidences and ranking stocks

The data collected for this study is based on the crisp observations, therefore $\left(\beta_i^{1,j} + \beta_i^{2,j} + \beta_i^{3,j}\right) = 1$ which means there is no uncertainty in the belief. For further explanation, the example of Tata Steel is extended. The D-S theory for combining the expert's attributes is discussed below:

The evaluations of the attributes are:

$$S(a_2^1) = \{(f_1, 0.667), (f_2, 0.222), (f_3, 0.111)\};$$

$$S(a_2^2) = \{(f_1, 0.111), (f_2, 0.222), (f_3, 0.667)\};$$

$$S(a_2^3) = \{(f_1, 0.333), (f_2, 0.444), (f_3, 0.222)\};$$

$$S(a_2^4) = \{(f_1, 0.111), (f_2, 0.444), (f_3, 0.444)\}$$

The weights of the expert's attributes are: $W_2 = \{w_2^1, w_2^2, w_2^2, w_2^2\} = \{0.32, 0.30, 0.16, 0.22\}$. BPA of the attributes is obtained by calculated by multiplying with the respective weights and $(1 - \sum BPA)$ is considered as the uncertainty or ignorance (λ).

The BPA of the attributes are represented as follows:

$$m(a_2^1) = \{m(a_2^{f_n,1}), \lambda(a_2^1)\} = \{0.21344, 0.07104, 0.03552, 0.68\}$$

$$m(a_2^2) = \{m(a_2^{f_n,2}), \lambda(a_2^2)\} = \{0.03333, 0.06666, 0.20010, 0.70\}$$

$$m(a_2^3) = \{m(a_2^{f_n,3}), \lambda(a_2^3)\} = \{0.05328, 0.07104, 0.03552, 0.84\}$$

$$m(a_2^4) = \{m(a_2^{f_n,4}), \lambda(a_2^4)\} = \{0.02442, 0.09768, 0.09768, 0.78\}$$

For combining evidences of first two attributes $\{a_2^1, a_2^2\}$, the aggregation $\left(P = \frac{1}{K}\right)$ is calculated as follows:

$$P(1 \oplus 2) = \left[1 - \sum_{n=1}^{3} \left\{\sum_{l=1}^{3} m(a_2^{f_n,1}) * m(a_2^{f_l,2})\right\}\right]^{-1}$$

$$= [1 - (0.21344 * 0.03333 + 0.21344 * 0.06666 + 0.21344 * 0.20010 + 0.07104 * 0.03333 + 0.07104 * 0.06666 + 0.07104 * 0.20010 + 0.03552 * 0.03333 + 0.03552 * 0.06666 + 0.03552 * 0.20010)]^{-1}$$

=1.0975

The new BPAs by combining $m(a_2^1)$ and $m(a_2^2)$ are as follows:

$$m\left(a_2^{f_1,(1\oplus 2)}\right) = P(1 \oplus 2) * \{m(a_2^{f_1,1}) * m(a_2^{f_1,2}) + m(a_2^{f_1,1}) * \lambda(a_2^2) + m(a_2^{f_1,2}) * \lambda(a_2^1)\}$$

$=1.0975 * (0.21344 * 0.03333 + 0.21344 * 0.70 + 0.03333 * 0.68)$

=0.19658

$$m\left(a_2^{f_2,(1\oplus 2)}\right) = P(1 \oplus 2) * \{m(a_2^{f_1,1}) * \lambda(a_2^2) + m(a_2^{f_1,2}) * \lambda(a_2^1) + \lambda(a_2^1) * \lambda(a_2^2)\}$$

$=1.0975 * (0.07104 * 0.06666 + 0.07104 * 0.70 + 0.06666 * 0.68)$

=0.10959

$m\left(a_2^{f_3,(1\oplus 2)}\right) = P(1 \oplus 2) * \left\{m\left(a_2^{f_1,1}\right) * \lambda(a_2^2) + m\left(a_2^{f_1,2}\right) * \lambda(a_2^1) + \lambda(a_2^1) * \lambda(a_2^2)\right\}$

$=1.0975 * (0.03552 * 0.20010 + 0.03552 * 0.70 + 0.20010 * 0.68)$

=0.18439

$\lambda\left(a_2^{(1\oplus 2)}\right) = P(1 \oplus 2) * \left\{m\left(a_2^{f_1,1}\right) * \lambda(a_2^2) + m\left(a_2^{f_1,2}\right) * \lambda(a_2^1) + \lambda(a_2^1) * \lambda(a_2^2)\right\}$

$= 1.0975 * (0.70 * 0.68)$

=0.52244

The obtained BPAs are combined with 3rd attribute, and in the similar manner new generated BPAs are combined with 4th attribute. After combining all attributes the final mass values for decision maker i.e. expert are obtained by the formulae given in equation 3.5.7.

Similarly, the final mass values for both decision makers for all companies are calculated, which are presented in Table 4.5.8. Further, the obtained final masses of the preposition *"good for investment (GI)"* are combined to achieve the ultimate goal of selecting the best stocks for investment. The final assessment (U) of the companies is determined by equation (3.5.8). We calculated U for 9 different pair the weights, i.e., $(\alpha, \beta) = \{(0.1, 0.9), (0.2, 0.8), (0.3, 0.7), (0.4, 0.6), (0.5, 0.5), (0.6, 0.4), (0.7, 0.3), (0.8, 0.2), (0.9, 0.1)\}$, here, α denotes the weightage of novice investor and β denotes the weightage of the domain experts in the final decision. Based on the final evaluation subjected to each pairs of weights the companies are ranked. The final ranking obtained by unifying the all rankings is presented in Table 4.5.9.

Table 4.5.8: Final masses of decision makers evaluation

Stocks	Novice Investor			Domain expert		
	BI	AI	GI	BI	AI	GI
Adani Enterprises Ltd.	0.555	0.198	0.247	0.338	0.261	0.401
Apollo Hos. Ent. Ltd.	0.046	0.303	0.651	0.382	0.215	0.403
Ashok Leyland Ltd.	0.287	0.368	0.345	0.339	0.028	0.633
Aurobindo Pharma Ltd.	0.161	0.589	0.250	0.103	0.099	0.798
UltraTech Cement Ltd.	0.000	0.038	0.962	0.175	0.291	0.534
Dabur India Ltd.	0.000	0.132	0.868	0.190	0.078	0.732
Divi’s Laboratories Ltd.	0.045	0.178	0.777	0.302	0.019	0.679
Dr. Reddy’s Lab. Ltd.	0.138	0.200	0.662	0.155	0.239	0.606
GAIL (India) Ltd.	0.190	0.206	0.603	0.115	0.156	0.728
HCL Technologies Ltd.	0.044	0.251	0.704	0.228	0.059	0.712
Hindutan Unilever Ltd.	0.090	0.199	0.711	0.302	0.019	0.679
Indiabulls Real Est. Ltd.	0.826	0.043	0.130	0.300	0.032	0.668
Infosys Ltd.	0.043	0.192	0.764	0.323	0.000	0.677
Jindal Stl. & power Ltd.	0.658	0.250	0.092	0.296	0.164	0.540
JSW Steel Ltd.	0.159	0.379	0.462	0.173	0.149	0.678
Jubilant FoodWorks Ltd.	0.100	0.650	0.250	0.335	0.100	0.565
Dr. Lal PathLabs Ltd.	0.333	0.206	0.460	0.242	0.000	0.758
Maruti Suzuki India Ltd.	0.092	0.250	0.658	0.235	0.166	0.599
Mphasis Ltd.	0.095	0.237	0.668	0.084	0.225	0.690
Nestle India Ltd.	0.042	0.137	0.821	0.172	0.077	0.751
ONGC Ltd.	0.265	0.341	0.395	0.084	0.286	0.630
Pidilite Industries Ltd.	0.000	0.132	0.868	0.308	0.013	0.679
Ratnamani M. & T. Ltd.	0.213	0.246	0.541	0.085	0.128	0.788
Reliance Industries Ltd.	0.000	0.183	0.817	0.238	0.285	0.477
Tata Motors Ltd.	0.249	0.431	0.320	0.139	0.100	0.761
Tata Steel Ltd.	0.217	0.482	0.301	0.324	0.303	0.373
TCS Ltd.	0.085	0.090	0.825	0.323	0.000	0.677
Titan Company Ltd.	0.144	0.316	0.540	0.250	0.135	0.615
Torrent Phar. Ltd.	0.045	0.310	0.645	0.203	0.025	0.772
Zee Entertain. Ent. Ltd.	0.267	0.204	0.529	0.257	0.039	0.705

Table 4.5.9 Ranking of securities

Companies	(0.1,0.9)	R_1	(0.2,0.8)	R_1	(0.3,0.7)	R_2	(0.4,0.6)	R_4	(0.5,0.5)	R_5	(0.6,0.4)	R_6	(0.7,0.3)	R_7	(0.8,0.2)	R_8	(0.9,0.1)	R_9	Final
Adani Enterprises Ltd.	0.3856	29	0.3702	29	0.3548	29	0.3394	30	0.324	29	0.3086	29	0.2932	28	0.2778	28	0.2624	28	29
Apollo Hos. Ent. Ltd.	0.4278	28	0.4526	27	0.4774	26	0.5022	25	0.527	22	0.5518	20	0.5766	18	0.6014	16	0.6262	15	23
Ashok Leyland Ltd.	0.6042	23	0.5754	23	0.5466	24	0.5178	24	0.489	25	0.4602	25	0.4314	24	0.4026	24	0.3738	23	25
Aurobindo Pharma Ltd.	0.7432	5	0.6884	12	0.6336	16	0.5788	22	0.524	23	0.4692	24	0.4144	25	0.3596	25	0.3048	26	22
UltraTech Cement Ltd.	0.5768	24	0.6196	18	0.6624	14	0.7052	9	0.748	5	0.7908	4	0.8336	1	0.8764	1	0.9192	1	**8**
Dabur India Ltd.	0.7456	4	0.7592	2	0.7728	1	0.7864	1	0.8	1	0.8136	1	0.8272	2	0.8408	2	0.8544	2	**1**
Divi's Laboratories Ltd.	0.6888	12	0.6986	9	0.7084	8	0.7182	6	0.728	6	0.7378	6	0.7476	6	0.7574	6	0.7672	7	**5**
Dr. Reddy's Lab. Ltd.	0.6116	19	0.6172	19	0.6228	18	0.6284	16	0.634	15	0.6396	15	0.6452	13	0.6508	13	0.6564	13	16
GAIL (India) Ltd.	0.7155	8	0.703	8	0.6905	10	0.678	13	0.6655	12	0.653	13	0.6405	14	0.628	15	0.6155	16	13
HCL Technologies Ltd.	0.7112	9	0.7104	6	0.7096	7	0.7088	8	0.708	9	0.7072	8	0.7064	9	0.7056	9	0.7048	10	**7**
Hindutan Unilever Ltd.	0.6822	16	0.6854	14	0.6886	11	0.6918	10	0.695	10	0.6982	9	0.7014	10	0.7046	10	0.7078	9	10
Indiabulls Real Est. Ltd.	0.6142	18	0.5604	24	0.5066	25	0.4528	26	0.399	27	0.3452	27	0.2914	29	0.2376	29	0.1838	29	26
Infosys Ltd.	0.6857	15	0.6944	11	0.7031	9	0.7118	7	0.7205	7	0.7292	7	0.7379	7	0.7466	8	0.7553	8	**9**
Jindal Stl. & power Ltd.	0.4952	27	0.4504	28	0.4056	28	0.3608	28	0.316	30	0.2712	30	0.2264	30	0.1816	30	0.1368	30	30
JSW Steel Ltd.	0.6564	17	0.6348	17	0.6132	20	0.5916	19	0.57	20	0.5484	21	0.5268	21	0.5052	21	0.4836	21	21
Jubilant FoodWorks Ltd.	0.5335	25	0.502	26	0.4705	27	0.439	27	0.4075	26	0.376	26	0.3445	26	0.313	27	0.2815	27	27
Dr. Lal PathLabs Ltd.	0.7282	6	0.6984	10	0.6686	13	0.6388	14	0.609	18	0.5792	18	0.5494	20	0.5196	20	0.4898	20	15
Maruti Suzuki India Ltd.	0.6049	22	0.6108	20	0.6167	19	0.6226	17	0.6285	16	0.6344	16	0.6403	15	0.6462	14	0.6521	14	18
Mphasis Ltd.	0.6878	13	0.6856	13	0.6834	12	0.6812	12	0.679	11	0.6768	12	0.6746	12	0.6724	11	0.6702	11	12
Nestle India Ltd.	0.758	3	0.765	1	0.772	2	0.779	2	0.786	2	0.793	2	0.8	4	0.807	4	0.814	4	**2**
ONGC Ltd.	0.6065	21	0.583	22	0.5595	23	0.536	23	0.5125	24	0.489	23	0.4655	22	0.442	22	0.4185	22	24
Pidilite Industries Ltd.	0.6979	10	0.7168	5	0.7357	3	0.7546	3	0.7735	3	0.7924	3	0.8113	3	0.8302	3	0.8491	3	**3**
Ratnamani M. & T. Ltd.	0.7633	1	0.7386	4	0.7139	6	0.6892	11	0.6645	13	0.6398	14	0.6151	16	0.5904	17	0.5657	17	**10**
Reliance Industries Ltd.	0.511	26	0.545	25	0.579	22	0.613	18	0.647	14	0.681	11	0.715	8	0.749	7	0.783	6	14
Tata Motors Ltd.	0.7169	7	0.6728	15	0.6287	17	0.5846	21	0.5405	21	0.4964	22	0.4523	23	0.4082	23	0.3641	24	19
Tata Steel Ltd.	0.3658	30	0.3586	30	0.3514	30	0.3442	29	0.337	28	0.3298	28	0.3226	27	0.3154	26	0.3082	25	28
TCS Ltd.	0.6918	11	0.7066	7	0.7214	5	0.7362	4	0.751	4	0.7658	5	0.7806	5	0.7954	5	0.8102	5	**4**
Titan Company Ltd.	0.6075	20	0.6	21	0.5925	21	0.585	20	0.5775	19	0.57	19	0.5625	19	0.555	19	0.5475	18	20
Torrent Phar. Ltd.	0.7593	2	0.7466	3	0.7339	4	0.7212	5	0.7085	8	0.6958	10	0.6831	11	0.6704	12	0.6577	12	**6**
Zee Entertain. Ent. Ltd.	0.6874	14	0.6698	16	0.6522	15	0.6346	15	0.617	17	0.5994	17	0.5818	17	0.5642	18	0.5466	19	17

4.5.5 Portfolio optimization

The optimization is executed in Google Colaboratory for annual investment in the year 2021. The daily closing price and returns of each security in the year 2021 has input in the neural network. We set up window size of 20 days to form a single input and used a single layer of LSTM connectivity, with 32 units. The neural network is iterated up to 2000 epochs in 32 min batch-size. The optimal solution i.e. allocated ratios obtained after 2000 iteration is given in Table 4.5.10. The maximum value of the Sharpe ratio converged to 2.25. The maximum return is found as 0.3670 with minimum volatility as 0.1451. Figure 4.5.1 clarifies the effectiveness of the implemented LSTM neural network for the optimizing proposed portfolio.

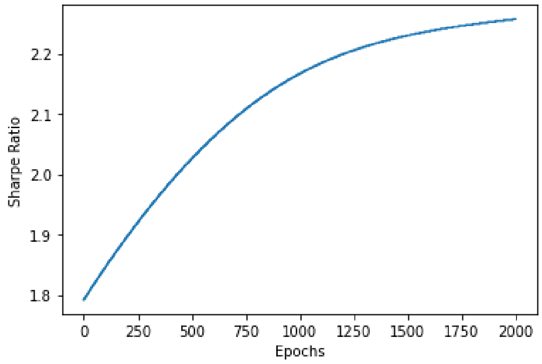


Figure 4.5.1: Convergence of the objective function by LSTM optimization

Table 4.5.10: Ratio allocation to the components of portfolio

Stocks	Dabur India Ltd.	Nestle India Ltd.	Pidilite Industries Ltd.	TCS Ltd.	Divi's Laboratories Ltd.	Torrent Phar. Ltd.	HCL Technologies Ltd.	UltraTech Cement Ltd.	Infosys Ltd.	Ratnamani M. & T. Ltd.
Annual return	0.0857	0.0674	0.3510	0.2711	0.2204	0.1684	0.3676	0.3762	0.4386	0.2486
Ratio allocation	0.0237	0.0223	0.2188	0.0169	0.0282	0.0319	0.0161	0.1567	0.4479	0.0371

4.5.6 Empirical results and discussion

4.5.6.1 Sensitivity analysis

In this section, sensitivity analysis is conducted for examining the significance of both decision makers on the ranking of securities. For this, we developed 11 scenarios by changing the weightage of decision makers in final ranking. The ranking obtained by 11 scenarios and original proposed ranking are presented graphically in Figure 4.5.2. It clears that the proposed model is sensitive to the weightage of the decision makers. Hence, the ranking of stocks could affect the quality of decision. The Spearman's correlation coefficient (SCC) between the proposed final ranking and the rankings obtained by all 12 scenarios is calculated which are shown in Table 4.5.11. For scenarios where the weight of the novice investor is greater than that of the expert, the correlation coefficients between the original ranking and the scenario rankings are found greater than 88%. It signifies that novice investor could make good decision with the little knowledge of a company's fundamentals. The correlations between the original ranking and scenario ranking are found low in two scenarios where the novice investor's weightage is very less (i.e. for (0, 1), (0.1, 0.9) and (0.2, 0.8). It means that novice investors' concerns about a company's past returns and risk are crucial for analysis.

For a deeper insight, we compare different portfolios of the top 10 companies in each scenario. Table 4.5.12 shows the results of the comparison. We discovered that the best case scenario is the recommended ranking, in which no security loses money, eight out of ten companies gain more than 10%, and the portfolio return is highest with the highest Sharpe ratio. The expected return of the expert's portfolio (for the (0, 1) scenario) is shown to be the second highest, but with the largest risk among the others. Other scenarios' portfolios have yielded similar returns and Sharpe ratios. As a result of the study, experts can build a successful portfolio with their in-depth analysis, and inexperienced investors can make a good decision as well. However, combining elements from both would be more profitable.

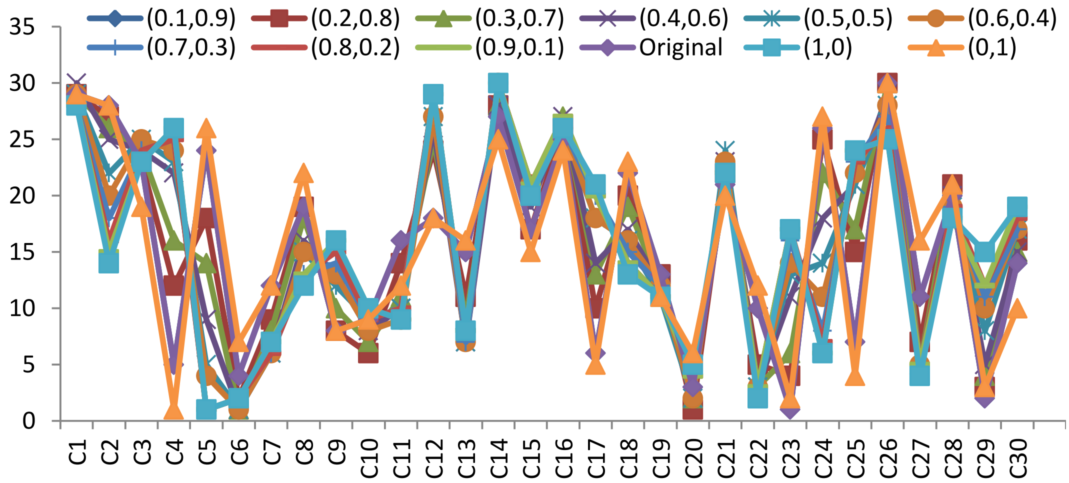


Figure 4.5.2: Effects of changing decision makers' weights on the ranking of stocks

Table 4.5.11: Spearman's correlation coefficient between original ranking and different scenarios ranking

Scenarios	(0,1)	(0.1,0.9)	(0.2,0.8)	(0.3,0.7)	(0.4,0.6)	(0.5,0.5)	(0.6,0.4)	(0.7,0.3)	(0.8,0.2)	(0.9,0.1)	(1,0)
Correlation coefficients	0.542	0.680	0.877	0.952	0.988	0.987	0.976	0.9394	0.923	0.907	0.885

Table 4.5.12: Comparison of the portfolio obtained by different scenarios

Scenarios	No. of securities with negative returns	No. of securities with less than 10% returns	No. of securities with more than 10% returns	Portfolio return (without optimization)	Portfolio return (after optimization)	Portfolio volatility (after optimization)	Sharpe Ratio
(0,1)	1	3	6	0.28	0.327	0.171	1.80
(0.1,0.9)	1	3	6	0.28	0.323	0.145	2.08
(0.2,0.8)	0	3	7	0.24	0.236	0.129	1.67
(0.3,0.7)	0	3	7	0.23	0.231	0.129	1.64
(0.4,0.6)	1	2	7	0.23	0.244	0.130	1.72
(0.5,0.5)	1	2	7	0.23	0.244	0.130	1.72
(0.6,0.4)	1	2	7	0.23	0.244	0.130	1.72
(0.7,0.3)	1	2	7	0.24	0.249	0.132	1.73
(0.8,0.2)	1	2	7	0.24	0.249	0.132	1.73
(0.9,0.1)	1	2	7	0.24	0.249	0.132	1.73
(1,0)	1	2	7	0.24	0.249	0.132	1.73
Proposed Ranking	0	2	8	0.26	0.367	0.145	2.25

4.5.6.2 Effectiveness of the proposed model

This model focuses on discovering stocks that are good for investment by analyzing past performance of companies in various perspectives. The ranking does not promise the highest return from the top ranked firms, but it does ensure that they are safe to invest in and there is merely any chance of any disastrous event in near future which can cause huge loss of investors. To verify the reliability of the ranking system, we checked the performances of the top 10 stocks which are recommended to compose a portfolio for investment in the year 2021. The stocks returns and standard deviation (volatility) are shown in Table 4.5.13. Every stock had good returns and volatility is also relatively low. Figure 4.5.3 depicts the daily closing price chart of top 10 companies in the year 2021. It is also clear from the chart that all companies of the portfolio are in uptrend in the year 2021 which verifies the application of the proposed model.

Table 4.5.13: Performance of top 10 ranked companies in the year 2021

Companies	Annual return	Standard deviation (Volatility)
Dabur India Ltd.	0.0857	0.0114
Nestle India Ltd.	0.0674	0.0112
Pidilite Industries Ltd.	0.351	0.0141
TCS Ltd.	0.2711	0.0132
Divi’s Laboratories Ltd.	0.2204	0.0159
Torrent Phar. Ltd.	0.1684	0.0161
HCL Technologies Ltd.	0.3676	0.0162
UltraTech Cement Ltd.	0.3762	0.0162
Infosys Ltd.	0.4386	0.0132
Ratnamani M. & T. Ltd.	0.2486	0.0180

Figure 4.5.3: Normalized closing price time series of companies

4.5.6.3 Comparison with existing models

The proposed portfolio annual return is discovered to be superior to portfolios derived by other models. The comparison results are summarized in Table 4.5.14. Even without optimization, the proposed portfolio outperforms all other models.

Table 4.5.14: Comparison of proposed model with other models

Models	**Thakur *et al.* (2018)**	**Naveenan (2019)**	**Narang *et al.* (2021)**	**Narang *et al.* (2022)**	**Proposed portfolio (without optimization)**	**Proposed portfolio (after optimization)**
Year	2016	2019	2020	2020	2021	2021
Annual return	0.1301	0.17	0.2361	0.1672	**0.2595**	**0.3670**

4.6 Model [6]: A method of intraday stock selection integrating fuzzy TOPSIS and belief divergence measure in evidence theory

In this study, the proposed methodology is applied on real stocks data of NSE. We intended to recommend some stocks to trade on June 18, 2021. It has been observed that traders generally consider stocks that undergo a large price change in the previous day. Such stocks are already in the news and on the list of big market players which makes trading these stocks much easier. Hence, we picked 10 stocks as alternatives that were the top 5 gainers and top 5 losers on the previous day, June 17, 2021. The stocks are listed below:

1. UltraTech Cement Limited (ULTRACEMCO) (A_1)
2. Tata Consultancy Services Limited (TCS) (A_2)
3. Infosys Limited (INFY) (A_3)
4. Asian Paints Limited (ASIANPAINTS) (A_4)
5. Tech Mahindra Limited (TECHM) (A_5)
6. Adani Ports and Special Economic Zone Limited (ADANIPORTS) (A_6)
7. Hindalco Industries Limited (HINDALCO) (A_7)
8. Indusind Bank Limilted (INDUSINDBK) (A_8)
9. Eicher Motors Limited (EICHERMOT) (A_9)
10. NTPC Limited (NTPC) (A_{10})

Following 6 important criteria are chosen for intraday stocks selection.

1. **% Change $(\boldsymbol{C_1})$:** It represents the degree of change in the price of a stock in over time. It is calculated as $\frac{new\ price - old\ price}{old\ price} * 100$. It gives an idea to estimate the price margin for next day. It can be negative or positive. For this study, we have taken the % change of stocks in previous 5 days. Extreme high % change would be riskier, therefore medium TFN is considered as its desired value.

2. **Volume** (C_2)**:** Volume refers to the quantity of shares traded over a given period of time. Intraday requires fast and accurate timing of taking buy or sell positions. High volume indicates the liquidity of the market which makes easier to get in and out of trades. Here, we take volume of previous day indicating high possibility of same quantity of trade in upcoming day. Hence, it is considered as beneficial criteria.

3. **Volatility** (C_3)**:** Volatility reflects the up and down fluctuations in the price of a stock. Stocks with high volatility are considered for intraday trading. As day trader has to trade within the day, therefore the stock with high volatility gives a good profit margin. It may lead to loss if prices move in downward trend. So, very high TFN value is set as desired value for volatility.

4. **Trend** (C_4)**:** Trend of a stock depicts the direction of the movement of stock. Trend analysis help in intraday trading to take the correct positions according to the intraday trend of a stock. A stock with upward trend (bullish) indicates to spot buying opportunity and downward trend (bearish) indicates to look for selling opportunity during intraday. In this study, the prices of last one week are noted to identify the trend of stock. Here, we consider very bullish trend as best value for trend criteria.

5. **Technical Rating** (C_5)**:** Technical rating of pervious day tells about the position of stock based on technical analysis. The technical rating combined the response of moving averages in different time frames, moving averages crossovers for short, medium and long term and some technical indicators such as relative strength index (RSI), moving average convergence divergence (MACD), stochastic oscillator, rate of change (ROC), commodity channel index (CCI), bollinger bands etc. Here also very bullish position is taken as desired value.

6. **Overbought/ Oversold** (C_6)**:** The overbought position of stock means that the stock is reached at higher price than it's worth and expected to be drop soon. Similarly, oversold position means the stock is trading below its fair value and it has the chance to go upward. Trader avoids these positions for starting

intraday. Here we note number of indicators indicating the overbought and oversold position of stocks at the end of previous day and take it as cost criteria.

4.6.1 Step-wise application of proposed model

Step 1. Define initial decision matrix $\{D_{ij}\}$

The initial decision matrix having crisp evaluation value of each stock under the above mentioned criteria at the end of the trading day 17/06/2021 is given in the Table 4.6.1. The data is collected from the websites https://www.moneycontrol.com/ and https://www.ratestar.in/.

Table 4.6.1: Initial decision matrix $\{D_{ij}\}$

Alternatives	%Change	Volume	Volatility	Trend	Technical Rating	Overbought/ oversold
ULTRACEMCO(A_1)	1.69	6,37,296	2.07	Very bullish	Bullish	0
TCS (A_2)	1.64	22,73,413	1.73	Bearish	Very bullish	2
INFY (A_3)	1.39	76,77,951	1.88	Neutral	Very bullish	3
ASIANPAINT (A_4)	1.36	15,33,864	1.95	Very bullish	Very bullish	3
TECHM (A_5)	1.3	31,61,931	2.05	Neutral	Very bullish	2
ADANIPORTS (A_6)	-8.99	7,08,12,142	2.79	Neutral	Bearish	2
HINDALCO (A_7)	-2.96	1,43,21,673	3	Bullish	Neutral	3
INDUSINDBK (A_8)	-2.96	61,26,591	3.82	Neutral	Neutral	0
EICHERMOT (A_9)	-2.63	7,01,970	2.29	Bullish	Bullish	0
NTPC (A_{10})	-2.34	1,44,66,025	1.93	Bullish	Bullish	1

Step 2. Transforming the initial decision matrix to fuzzy decision matrix

The evaluation values are compared and converted into linguistic terms, represented in the Table 4.6.2. Then the linguistic terms converted into TFNs, presented in Table 4.6.3.

Table 4.6.2: Decision matrix in linguistic terms $\{L_{ij}\}$

Alternatives	%Change	Volume	Volatility	Trend	Technical Rating	Overbought/oversold
ULTRACEMCO(A_1)	M	L	MH	EH	MH	EL
TCS (A_2)	M	M	ML	L	VH	H
INFY (A_3)	ML	MH	ML	M	H	EH
ASIANPAINT (A_4)	ML	ML	M	EH	H	EH
TECHM (A_5)	ML	M	MH	M	VH	H
ADANIPORTS (A_6)	EH	EH	VH	M	VL	H
HINDALCO (A_7)	H	H	VH	H	L	EH
INDUSINDBK (A_8)	H	MH	EH	M	ML	EL
EICHERMOT (A_9)	MH	L	H	H	EH	EL
NTPC (A_{10})	MH	H	M	H	M	L

Table 4.6.3: TFNs representation of decision matrix $\{E_{ij}\}$

Alternatives	%Change	Volume	Volatility	Trend	Technical Rating	Overbought/oversold
ULTRACEMCO(A_1)	(0.4,0.5,0.6)	(0.2,0.3,0.4)	(0.5,0.6,0.7)	(0.8,0.9,0.9)	(0.5,0.6,0.7)	(0.1,0.1,0.1)
TCS (A_2)	(0.4,0.5,0.6)	(0.4,0.5,0.6)	(0.3,0.4,0.5)	(0.2,0.3,0.4)	(0.7,0.8,0.9)	(0.6,0.7,0.8)
INFY (A_3)	(0.3,0.4,0.5)	(0.5,0.6,0.7)	(0.3,0.4,0.5)	(0.4,0.5,0.6)	(0.6,0.7,0.8)	(0.8,0.9,0.9)
ASIANPAINT (A_4)	(0.3,0.4,0.5)	(0.3,0.4,0.5)	(0.4,0.5,0.6)	(0.8,0.9,0.9)	(0.6,0.7,0.8)	(0.8,0.9,0.9)
TECHM (A_5)	(0.3,0.4,0.5)	(0.4,0.5,0.6)	(0.5,0.6,0.7)	(0.4,0.5,0.6)	(0.7,0.8,0.9)	(0.6,0.7,0.8)
ADANIPORTS (A_6)	(0.8,0.9,0.9)	(0.8,0.9,0.9)	(0.7,0.8,0.9)	(0.4,0.5,0.6)	(0.1,0.2,0.3)	(0.6,0.7,0.8)
HINDALCO (A_7)	(0.6,0.7,0.8)	(0.6,0.7,0.8)	(0.7,0.8,0.9)	(0.6,0.7,0.8)	(0.2,0.3,0.4)	(0.8,0.9,0.9)
INDUSINDBK (A_8)	(0.6,0.7,0.8)	(0.5,0.6,0.7)	(0.8,0.9,0.9)	(0.4,0.5,0.6)	(0.3,0.4,0.5)	(0.1,0.1,0.1)
EICHERMOT (A_9)	(0.5,0.6,0.7)	(0.2,0.3,0.4)	(0.6,0.7,0.8)	(0.6,0.7,0.8)	(0.8,0.9,0.9)	(0.1,0.1,0.1)
NTPC (A_{10})	(0.5,0.6,0.7)	(0.6,0.7,0.8)	(0.4,0.5,0.6)	(0.6,0.7,0.8)	(0.4,0.5,0.6)	(0.2,0.3,0.4)

Step 3. Identify the best and worst TFNs values of each criterion.

The best and worst values of each criterion are consulted by the domain experts which are presented in Table 4.6.4.

Table 4.6.4: Best and worst TFNs values of each criterion

	%Change	Volume	Volatility	Trend	Technical Rating	Overbought/ oversold
Best	(0.4,0.5,0.6)	(0.8,0.9,0.9)	(0.7,0.8,0.9)	(0.8,0.9,0.9)	(0.8,0.9,0.9)	(0.1,0.1,0.1)
Worst	(0.8,0.9,0.9)	(0.2,0.3,0.4)	(0.3,0.4,0.5)	(0.2,0.3,0.4)	(0.2,0.3,0.4)	(0.8,0.9,0.9)

Step 4. Calculate distance between best (worst) criterion value and evaluated value

The distance between the best (worst) criterion value and evaluated TFN value is calculated by the help of Euclidian distance between two TFNs. The distance matrices from best value and worst value are present in Table 4.6.5 and Table 4.6.6, respectively.

Table 4.6.5: Distance from best value

Alternatives	%Change	Volume	Volatility	Trend	Technical Rating	Overbought/ oversold
A_1	0.000	0.569	0.200	0.000	0.271	0.000
A_2	0.000	0.370	0.400	0.569	0.082	0.606
A_3	0.100	0.271	0.400	0.370	0.173	0.768
A_4	0.100	0.469	0.300	0.000	0.173	0.768
A_5	0.100	0.370	0.200	0.370	0.082	0.606
A_6	0.370	0.000	0.000	0.370	0.668	0.606
A_7	0.200	0.173	0.000	0.173	0.569	0.768
A_8	0.200	0.271	0.082	0.370	0.469	0.000
A_9	0.100	0.569	0.216	0.173	0.000	0.000
A_{10}	0.100	0.173	0.300	0.173	0.370	0.216

Table 4.6.6: Distance from worst value

Alternatives	%Change	Volume	Volatility	Trend	Technical Rating	Overbought/ oversold
A_1	0.370	0.000	0.518	0.569	0.300	0.768
A_2	0.370	0.200	0.000	0.000	0.500	0.173
A_3	0.469	0.300	0.000	0.200	0.400	0.000
A_4	0.469	0.100	0.466	0.569	0.400	0.000
A_5	0.469	0.200	0.518	0.200	0.500	0.173
A_6	0.000	0.569	0.620	0.200	0.100	0.173
A_7	0.173	0.400	0.620	0.400	0.000	0.000
A_8	0.173	0.300	0.625	0.200	0.100	0.768
A_9	0.271	0.000	0.524	0.400	0.569	0.768
A_{10}	0.271	0.400	0.466	0.400	0.200	0.569

Step 5. Define frame of discernment (Θ) and basic probability assignment (BPA)

The BPAs obtained for supporting the evidences for two events Low (L) and High (H) for each criteria and each alternative are given in Table 4.6.7.

Table 4.6.7: BPAs corresponding each evidence (criteria) to all alternatives

Alternatives	%Change		Volume		Volatility		Trend		Technical Rating		Overbought/ oversold	
	L\`	H	L	H	L	H	L	H	L	H	L	H
A_1	0.000	1.000	1.000	0.000	0.279	0.721	0.000	1.000	0.474	0.526	0.000	1.000
A_2	0.000	1.000	0.649	0.351	1.000	0.000	1.000	0.000	0.140	0.860	0.778	0.222
A_3	0.176	0.824	0.474	0.526	1.000	0.000	0.649	0.351	0.302	0.698	1.000	0.000
A_4	0.176	0.824	0.824	0.176	0.392	0.608	0.000	1.000	0.302	0.698	1.000	0.000
A_5	0.176	0.824	0.649	0.351	0.279	0.721	0.649	0.351	0.140	0.860	0.778	0.222
A_6	1.000	0.000	0.000	1.000	0.000	1.000	0.649	0.351	0.870	0.130	0.778	0.222
A_7	0.536	0.464	0.302	0.698	0.000	1.000	0.302	0.698	1.000	0.000	1.000	0.000
A_8	0.536	0.464	0.474	0.526	0.116	0.884	0.649	0.351	0.824	0.176	0.000	1.000
A_9	0.270	0.730	1.000	0.000	0.292	0.708	0.302	0.698	0.000	1.000	0.000	1.000
A_{10}	0.270	0.730	0.302	0.698	0.392	0.608	0.302	0.698	0.649	0.351	0.275	0.725

From step 6 to step 10, the same procedure is followed for all alternatives. Here, we have elaborated for alternative A_1.

Step 6. Obtain symmetric belief divergence matrix (SBDM) for each alternative

The SBDM matrix for A_1 is given in Table 4.6.8.

Table 4.6.8: Symmetric belief divergence matrix for A_1

0.000	26.766	2.113	0.000	3.799	0.000
26.766	0.000	13.839	27.918	9.859	27.918
2.113	13.839	0.000	2.433	0.083	2.433
0.000	27.918	2.433	0.000	4.345	0.000
3.799	9.859	0.083	4.345	0.000	4.345
0.000	27.918	2.433	0.000	4.345	0.000

Step 7. Calculate the supporting degree of each evidence

For A_1, the supporting degree corresponding each evidence are $S_1 = 0.0306, S_2 = 0.0094, S_3 = 0.0478, S_4 = 0.0288, S_5 = 0.0445$ and $S_6 = 0.0288$.

Step 8. Calculate weight of each evidence

For A_1, the weight of each evidence are $W_1 = 0.1609$, $W_2 = 0.0494, W_3 = 0.2517, W_4 = 0.1516, W_5 = 0.2345$ and $W_6 = 0.1516$.

Step 9. Calculated weighted BPA of evidences

For A_1, the weighted BPA of each evidence are given in Table 4.6.9.

Table 4.6.9: Weighted basic probabilities

$\boldsymbol{M_i(L)}$	$\boldsymbol{M_i(H)}$
0.000	0.161
0.049	0.000
0.070	0.182
0.000	0.152
0.111	0.123
0.000	0.152

Step 10. Combining evidences

The weighted BPAs of first two evidences of A_1 are represented as:

$$M_1 = \{M_1(L), M_1(H), \lambda_1\} = \{0.000, 0.161, 0.839\}$$

$$M_2 = \{M_2(L), M_2(H), \lambda_2\} = \{0.049, 0.000, 0.950\}$$

here, λ_i represents the degree of ignorance. The aggregation $\left(P = \frac{1}{K}\right)$ is calculated as follows:

$$P(1 \oplus 2) = [1 - \{M_1(L) * M_2(L) + M_1(L) * M_2(H) + M_1(H) * M_2(L) + M_1(H) * M_2(H)\}]^{-1} = 1.008$$

New BPAs generated by combining M_1 and M_2 are as follows:

$$M_{1\oplus 2}(L) = P(1 \oplus 2) * \{M_1(L) * M_2(L) + M_1(L) * \lambda_2 + M_2(L) * \lambda_1\} = 0.0418$$

$$M_{1\oplus 2}(H) = P(1 \oplus 2) * \{M_1(H) * M_2(H) + M_1(H) * \lambda_2 + M_2(H) * \lambda_1\} = 0.1542$$

$$\lambda_{1\oplus 2} = P(1 \oplus 2) * \lambda_1 * \lambda_2 = 0.8038$$

The obtained BPAs are combined with 3rd evidence, and in the similar manner upto 6th evidence. After combining all evidences the final mass values are obtained as:

$$\{L\} = \frac{M_{1\oplus 2\oplus 3\oplus 4\oplus 5\oplus 6}(L)}{M_{1\oplus 2\oplus 3\oplus 4\oplus 5\oplus 6}(L) + M_{1\oplus 2\oplus 3\oplus 4\oplus 5\oplus 6}(H)} = 0.1862$$

$$\{H\} = \frac{M_{1\oplus 2\oplus 3\oplus 4\oplus 5\oplus 6}(H)}{M_{1\oplus 2\oplus 3\oplus 4\oplus 5\oplus 6}(L) + M_{1\oplus 2\oplus 3\oplus 4\oplus 5\oplus 6}(H)} = 0.8137$$

Similarly, final belief values for all alternatives are calculated.

Step 11. Ranking alternatives

At last, the alternatives are ranked based on the final belief values as given in Table 4.6.10.

Table 4.6.10: Final belief values and ranking of alternatives

Alternatives	**L**	**U**	**Ranking**
A_1 (ULTRACEMCO)	0.18622	0.81378	1
A_2 (TCS)	0.671832	0.328168	10
A_3 (INFY)	0.589748	0.410252	8
A_4 (ASIANPAINT)	0.396066	0.603934	4
A_5 (TECHM)	0.431878	0.568122	6
A_6 (ADANIPORTS)	0.291391	0.708609	3
A_7 (HINDALCO)	0.481964	0.518036	7
A_8 (INDUSINDBK)	0.398183	0.601817	5
A_9 (EICHERMOT)	0.205188	0.794812	2
A_{10} (NTPC)	0.670479	0.329521	9

4.6.2 Experimental verification using intraday trading bot

In this section, the performance of the stocks on next trading day is evaluated by a deep reinforcement learning based trading bot. The trading bot has been trained to give trading signals {buy, hold and sell} in the real-time market. Consider, the sequence of closing prices of a stock as $p_1, p_2, \dots, p_t, \dots$ with 1 minute difference and $r_t = p_t - p_{t-1}$ as the return at time t. Let, $F_t = [r_{t-n+1}, \dots, r_t] \in \mathbb{R}_n$ be the feature vector having previous n returns at time t. Observing the sequence of previous n returns, the trading bot is trained to make a decision (policy) $\phi_t = \{-1,0,1\}$ i.e. {sell, hold, buy} in the real time at the end of time interval $(t-1, t)$. The decision (policy) ϕ_t is calculated by the following equation:

$$\phi_t = tanh(\langle u, F_t \rangle + v + w\phi_{t-1}) \qquad (4.6.1)$$

here, u, v, w are the learning parameters of reinforcement learning. The term $w\,\phi_{t-1}$ incorporates the impact of trading decision made in pervious time step. The profit earned at time step t is calculated as:

$$P_t = \phi_{t-1} * r_t - c|\phi_t - \phi_{t-1}| \qquad (4.6.2)$$

The main objective of the trading bot is to maximize the total profit while taking minimum risk throughout the trading period T. The trading bot aim to maximize the risk-adjustable return function Y, which is defined as:

$$Y = max \sum_{t=1}^{T} [a\{mean(P_{t+k}, \dots, P_t)\} - b\{stdev(P_{t+k}, \dots, P_t)\}] \qquad (4.6.3)$$

here, a and b are the adjustable parameters, here we take $a = 1, b = 0.01$. The first term in the summation determines the average of recent k returns and second term determines the standard deviation of recent k returns. To achieve the maximum total profit the second term has to get minimize i.e. minimum risk is taken throughout the trading period T.

The LSTM recurrent neural network based deep reinforcement learning system is set to train the trading bot. First layer of the neural network is the input layer having 50 nodes containing closing price of previous 50 days. The data is forwarded to LSTM recurrent neural network which is designed to analyse the pattern in sequential data and memorizing the information for long period of time. An LSTM cell consists a memory unit c, a hidden state h and three types of gate- input gate i, output gate o and forget gate f. These gates regulate the flow of information through the cell. At each time step t input x_t received by input layer and previous hidden state h_{t-1} activates all three gates. Forget gate removes less important information from cell, input gate feed some additional information to the cell and output gate selects some important information and show it, then memory cell c_t and hidden state h_t get updated. h_t is the final output of the LSTM cell. The overview of trading model is displayed in Figure 4.6.1. The involved computations are given as follows:

$$\left.\begin{aligned} i_t &= \sigma(w_{xi}x_t + w_{hi}h_{t-1} + b_i) \\ f_t &= \sigma\left(w_{xf}x_t + w_{hf}h_{t-1} + b_f\right) \\ o_t &= \sigma(w_{xo}x_t + w_{ho}h_{t-1} + b_o) \\ c_t &= f_t * c_{t-1} + i_t * tanh(w_{xi}x_t + w_{hi}h_{t-1} + b_i) \\ h_t &= o_t * tanh(c_t) \end{aligned}\right\} \quad (4.6.4)$$

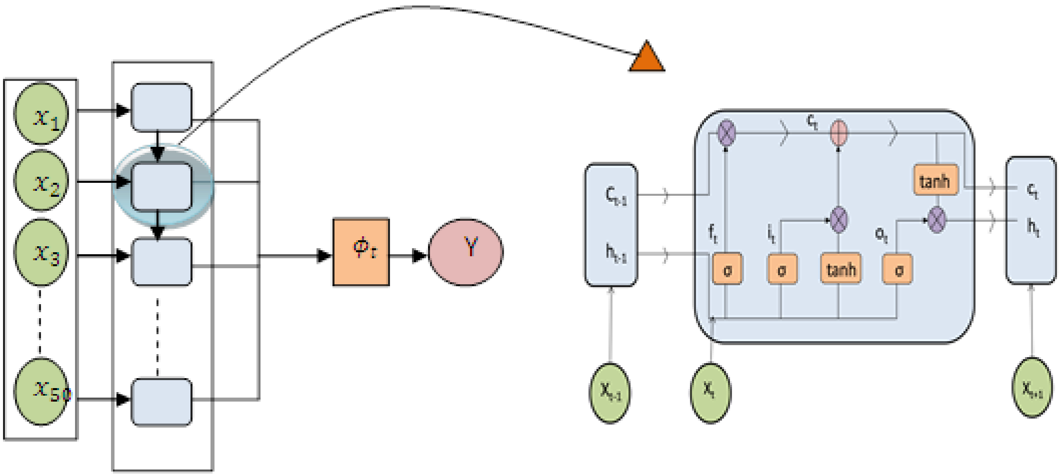
The output from the LSTM node input into last layer of DRL network having one node for decision (policy) making:

$$\phi_t = tanh(\langle u, h_t \rangle + v + w\phi_{t-1}) \quad (4.6.5)$$

At $t = 0$, we take $\phi_0 = 0$ i.e. initial trading decision is to be in holding position. The values of θ_t are ranged between -1 to 1, so here define a parameter λ to distinguish three decisions, here we take $\lambda = 0.33$ and determine $\phi_t$as follows:

$$\phi_t = \begin{cases} -1\,, -1 \le \phi_t \le \lambda \\ 0, -\lambda \le \phi_t \le \lambda \\ 1, \lambda \le \phi_t \le 1 \end{cases} \quad (4.6.6)$$

The decision made at each tick collectively calculates the value of multi-objective function Y i.e. profit under risk at the end of the trading period T. Lastly, the whole neural network trained to maximize the function Y in equation 4.6.3.



LSTM Cell

Figure 4.6.1: Overview of trading bot

4.6.3 Performance analysis

The minute-level closing price data of each stocks form 04/April/2021 to 17/June/2021 is used for training the trading bot. After training the bot predict the signals in the live market on 18/June/2021. For instance the chart of the closing price of Asian paints used as training and testing are depicted in the Figure 4.6.2(a) and Figure 4.6.2(b).

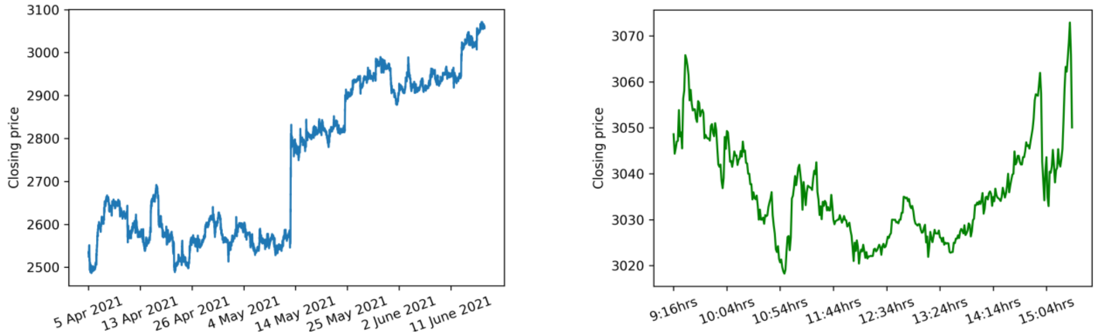


Figure 4.6.2: Closing prices (minute resolution) of Asian paints.

Table 4.6.11 presents the performance assessed by the trading bot for each stock according to the price fluctuations on 18 June 2021. Each stock's maximum total profit is also given in the Table 4.6.11. It can be seen that the top four rated stocks produce good intraday profits, and that these stocks also have a lot of buy and sell chances in the live market. Alternative A_1 ranked on first place earned maximum profit and also found most volatile. A_4 does not perform well as expected according to its rank. Except A_4, top five stocks did well, gaining more than 2000 points. Top ranked stocks also found highly volatile. The stocks ranked six and below attain less profit than 1000 points except A_2. Figure 4.6.3 graphically represents the obtained results. Hence, we can say that despite of highly uncertain behavior of stock market, the proposed method give a reliable ranking.

To verify the credibility of the proposed method, it is tested for another two trading days. Stocks are ranked to trade on December 31, 2021 and March 14, 2022. Tables 4.6.12 and Table 4.6.13 show the achieved results. The results are graphically represented in Figure 4.6.4 and Figure 4.6.5. Results of both days supported the applicability of the proposed methodology. Hence, the case study verifies that if a trader selects top-ranked stocks recommended by proposed model one day prior to trading, a good profit can be archived.

Table 4.6.11: Performance of the trading bot in live market on 18 June 2021

Rank	Alternatives	Volatility	High	Low	No. of buy signals	No. of sell signals	Total transactions	Total profit
1	ULTRACEMCO(A_1)	26.48	6723.82	6596.25	212	135	347	4077.19
2	EICHERMOT(A_9)	11.83	2712.95	2644.2	212	312	524	2597.8
3	ADANIPORTS (A_6)	17.29	704.05	647.2	168	168	336	3904.20
4	ASIANPAINT (A_4)	10.64	3072.9	3018.25	14	114	228	986.65
5	INDUSINDBK (A_8)	8.27	999.65	961.2	156	95	251	2193.09
6	TECHM (A_5)	5.95	1090.1	1055.45	161	151	312	496.15
7	HINDALCO(A_7)	2.98	373.1	360.45	133	103	236	434.25
8	INFY(A_3)	4.68	1514.5	1487.65	67	67	134	123.54
9	NTPC(A_{10})	1.06	117.7	112.9	31	45	79	12.96
10	TCS (A_2)	11.05	3355.75	3277.1	152	152	304	1759.65

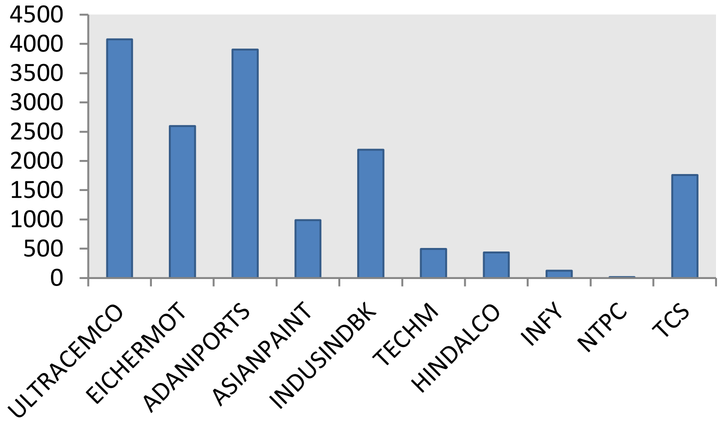


Figure 4.6.3: Total profit attained by stocks on 18 June 2021

Table 4.6.12: Performance of the trading bot in live market on 31 December 2021

Stocks	Rank	Volatility	Total Profit
NTPC	10	0.25	16.2
INDUSIND	9	2.13	156.2
HCL TECHNOLOGIES	6	4.15	959.6
TITAN	1	9.11	2015.5
CIPLA	3	3.44	616.56
BAJAJ AUTO	4	8.81	1985.3
RELIANCE	5	4.86	756.65
JSW STEEL	7	3.33	253.36
TATA STEEL	8	3.00	224.35
MARUTI	2	21.95	3785.95

Table 4.6.13: Performance of the trading bot in live market on 14 March 2022

Stocks	Rank	Volatility	Total Profit
CIPLA	4	3.75	1085.12
BPCL	8	1.84	440.32
SUNPHARMA	3	5.12	1202.62
JSWSTEEL	7	2.51	884.65
IOC	9	0.46	123.26
NESTLEIND	2	50.15	5402.23
MARUTI	1	87.18	6452.02
TATACONSUM	6	2.35	356.25
HINDALCO	5	4.07	1165.20
NTPC	10	0.41	156.25

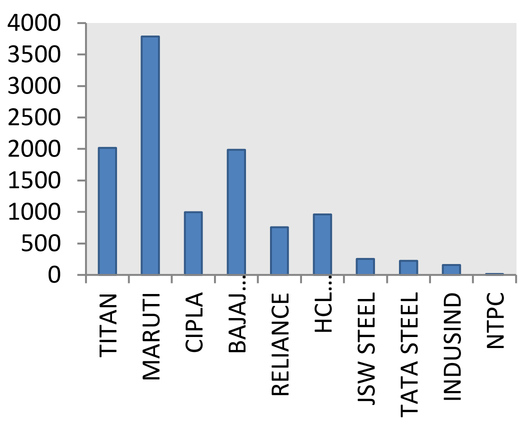


Figure 4.6.4: Total profit attained by stocks on 31 Dec 2021

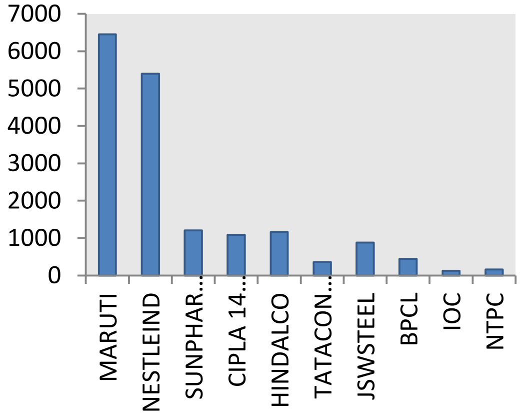


Figure 4.6.5: Total profit attained by stocks on 14 Mar 2022

4.7 Model [7]: Single valued triangular neutrosophic MEREC-CoCoSo method for multi-criteria decision making and its application in portfolio construction

In this study, we illustrated the model using NSE data. The ranking given by the proposed model is verified for portfolio construction.

4.7.1 Step-wise illustration of the proposed model

Step 1. Identification of criteria and alternatives

The following ten securities are included in the set of alternatives:

1. Apollo Hospitals Enterprise Ltd. (APOLLOHOSP) (A_1)
2. Reliance Industries Limited (RELIANCE) (A_2)
3. Oil & Natural Gas Corporation Limited (ONGC) (A_3)
4. Jubilant Foodworks Limited (JUBLFOOD) (A_4)
5. HDFC Bank Limited (HDFCBANK) (A_5)
6. State Bank of India (SBIN) (A_6)
7. Bajaj Finance Limited (BAJFINANCE) (A_7)
8. Titan Company Limited (TITAN)(A_8)
9. Asian Paints Limited (ASIANPAINT) (A_9)
10. Tata Steel Limited (TATASTEEL) (A_{10})

The set of criteria is formed considering following five indicators: (Revenue(C_1), Debt Equity Ratio (D/E) (C_2), Return on Equity (ROE) (C_3), Price to Earnings ratio (P/E) (C_4) and Price to Book Value ratio (P/B) (C_5)).

Step 2. Determination of the decision matrix

Real data of the ten stocks for the above mentioned criteria (from Jan-2010 to Dec-2020) is collected from the websites http://www.investello.com and http://www.ratestar.in. Since the collected data is multi-dimensional, a method of “exponential moving average (EMA)” is used to convert to convert multi-dimensional data into a deterministic value which is taken as evaluation value. The evaluation matrix is given in the Table 4.7.1. Based on the description of the linguistic term for each criterion given in Table 4.7.2, the observations of evaluation matrix are changed to linguistic term. Then each linguistic term is converted to STFNs according to the Table 3.7.1. The initial decision matrix thus obtained is presented in Table 4.7.4.

Table 4.7.1: Crisp evaluation matrix

Securities	Revenue (C_1)	D/E (C_2)	ROE (C_3)	P/E (C_4)	P/B (C_5)
APOLLOHOSP	9300.04	1.014	9.339	93.705	7.890
ASIANPAINTS	18434.92	0.056	17.550	18.996	16.306
BAJFINANCE	18341.78	4.520	17.556	43.238	7.372
HDFCBANK	118167.20	0.357	15.783	26.453	4.148
JUBLFOOD	3376.76	0.0004	22.099	83.684	20.079
ONGC	390249.70	0.481	9.567	9.729	0.816
RELIANCE	511663.60	0.737	10.258	22.35	2.253
SBIN	365657.50	3.122	4.347	55.626	1.262
TATASTEEL	145832.20	1.674	6.418	9.472	0.924
TITAN	18139.04	0.245	14.647	77.338	15.032

Table 4.7.2: Description of criteria into linguistic terms

Revenue	Below 10000 (ML)	10000-20000 (L)	100000-200000 (H)	Above 200000 (MH)	
D/E	0-0.5 (MH)	0.5-1 (H)	1-2 (M)	2-3 (L)	Above 3 (ML)
ROE	0-5 (L)	5-10 (M)	10-15 (H)	Above 15 (MH)	
P/E	0-10 (MH)	10-20 (M)	Above 20 (ML)		
P/B	0-1 (MH)	1-2 (H)	2-3 (L)	Above 3 (ML)	

Table 4.7.3: Initial decision matrix $(\tilde{D}_{ij})$

	C_1	C_2	C_3	C_4	C_5
A_1	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(3,5,7); 0.5,0.5,0.5⟩	⟨(3,5,7); 0.5,0.5,0.5⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(0,1,3); 0.17,0.85,0.83⟩
A_2	⟨(1,3,5); 0.33,0.75,0.67⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(3,5,7); 0.5,0.5,0.5⟩	⟨(1,3,5); 0.33,0.75,0.67⟩
A_3	⟨(1,3,5); 0.33,0.75,0.67⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(0,1,3); 0.17,0.85,0.83⟩
A_4	⟨(5,7,9); 0.67,0.25,0.33⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(0,1,3); 0.17,0.85,0.83⟩
A_5	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(0,1,3); 0.17,0.85,0.83⟩
A_6	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(3,5,7); 0.5,0.5,0.5⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(7,9,10); 0.83,0.15,0.17⟩
A_7	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(5,7,9); 0.67,0.25,0.33⟩	⟨(5,7,9); 0.67,0.25,0.33⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(1,3,5); 0.33,0.75,0.67⟩
A_8	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(1,3,5); 0.33,0.75,0.67⟩	⟨(0,1,3); 0.17,0.85,0.83⟩	⟨(5,7,9); 0.67,0.25,0.33⟩
A_9	⟨(5,7,9); 0.67,0.25,0.33⟩	⟨(3,5,7); 0.5,0.5,0.5⟩	⟨(3,5,7); 0.5,0.5,0.5⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(7,9,10); 0.83,0.15,0.17⟩
A_{10}	⟨(1,3,5); 0.33,0.75,0.67⟩	⟨(7,9,10); 0.83,0.15,0.17⟩	⟨(5,7,9); 0.67,0.25,0.33⟩	⟨(5,7,9); 0.67,0.25,0.33⟩	⟨(0,1,3); 0.17,0.85,0.83⟩

Step 3. Normalize fuzzy decision matrix:

The observations of the matrix $\tilde{D}_{ij}$ are normalized using equation 3.7.12. Table 4.7.4 represents the normalized decision matrix $(\tilde{N}_{ij})$.

Table 4.7.4: Normalized decision matrix $(\tilde{N}_{ij})$

	C_1	C_2	C_3	C_4	C_5
A_1	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0.3,0.56,1); 0.5,0.5,0.5⟩	⟨(0.3,0.56,1); 0.5,0.5,0.5⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩
A_2	⟨(0.1,0.33,0.71); 0.33,0.75,0.67⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.3,0.56,1); 0.5,0.5,0.5⟩	⟨0.1,0.33,0.71; 0.33,0.75,0.67⟩
A_3	⟨(0.1,0.33,0.71); 0.33,0.75,0.67⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩
A_4	⟨(0.5,0.78,1.28); 0.67,0.25,0.33⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩
A_5	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩
A_6	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.3,0.56,1); 0.5,0.5,0.5⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩
A_7	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.5,0.78,1.28); 0.67,0.25,0.33⟩	⟨(0.5,0.78,1.28); 0.67,0.25,0.33⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨(0.1,0.33,0.71); 0.33,0.75,0.67⟩
A_8	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨0.1,0.33,0.71; 0.33,0.75,0.67⟩	⟨(0,0.11,0.42); 0.17,0.85,0.83⟩	⟨0.5,0.78,1.28; 0.67,0.25,0.33⟩
A_9	⟨(0.5,0.78,1.28); 0.67,0.25,0.33⟩	⟨(0.3,0.56,1); 0.5,0.5,0.5⟩	⟨(0.3,0.56,1); 0.5,0.5,0.5⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩
A_{10}	⟨(0.1,0.33,0.71); 0.33,0.75,0.67⟩	⟨(0.7,1,1.42); 0.83,0.15,0.17⟩	⟨(0.5,0.78,1.28); 0.67,0.25,0.33⟩	⟨0.5,0.78,1.28; 0.67,0.25,0.33⟩	⟨0,0.11,0.42; 0.17,0.85,0.83⟩

Step 4. Compute overall performance of the alternatives:

Table 4.7.5 presents overall performance of all alternatives obtained by using parabolic measure as given equation 3.7.13.

Table 4.7.5: Overall performance of the alternatives

A_1	⟨(0.04,0.13,0.51); 0.17,0.85,0.83⟩
A_2	⟨(0.22,0.51,1.22); 0.33,0.75,0.67⟩
A_3	⟨(0.1,0.23,0.62); 0.17,0.85,0.83⟩
A_4	⟨(0.25,0.53,1.22); 0.17,0.85,0.83⟩
A_5	⟨(0.2,0.41,0.93); 0.17,0.85,0.83⟩
A_6	⟨(0.41,0.86,1.83); 0.5,0.5,0.5⟩
A_7	⟨(0.2,0.47,1.21); 0.17,0.85,0.83⟩
A_8	⟨(0.15,0.35,0.91); 0.17,0.85,0.83⟩
A_9	⟨(0.28,0.64,1.55); 0.5,0.5,0.5⟩
A_{10}	⟨(0.2,0.47,1.21); 0.17,0.85,0.83⟩

Step 5. Enumeration of the performance of alternatives by deleting each criterion:

Table 4.7.6 shows the performance of alternatives by deleting each criterion, separately by using equation 3.7.14.

Table 4.7.6: Enumeration of the performance of alternatives by deleting each criterion

A_1	⟨(0.04,0.13,0.47); 0.17,0.85,0.83⟩	⟨(0.02,0.07,0.31); 0.17,0.85,0.83⟩	⟨(0.02,0.07,0.31); 0.17,0.85,0.83⟩	⟨(0.04,0.13,0.47); 0.17,0.85,0.83⟩	⟨(0.04,0.13,0.47); 0.17,0.85,0.83⟩
A_2	⟨(0.22,0.48,1.12); 0.33,0.75,0.67⟩	⟨(0.12,0.31,0.81); 0.33,0.75,0.67⟩	⟨(0.12,0.31,0.81); 0.33,0.75,0.67⟩	⟨(0.2,0.44,1.02); 0.33,0.75,0.67⟩	⟨(0.22,0.48,1.12); 0.33,0.75,0.67⟩
A_3	⟨(0.1,0.21,0.52); 0.17,0.85,0.83⟩	⟨(0.1,0.23,0.58); 0.17,0.85,0.83⟩	⟨(0,0.03,0.21); 0.17,0.85,0.83⟩	⟨(0.1,0.23,0.58); 0.17,0.85,0.83⟩	⟨(0.1,0.23,0.58); 0.17,0.85,0.83⟩
A_4	⟨(0.2,0.4,0.89); 0.17,0.85,0.83⟩	⟨(0.15,0.33,0.81); 0.17,0.85,0.83⟩	⟨(0.15,0.33,0.81); 0.17,0.85,0.83⟩	⟨(0.25,0.52,1.18); 0.17,0.85,0.83⟩	⟨(0.25,0.52,1.18); 0.17,0.85,0.83⟩
A_5	⟨(0.2,0.4,0.89); 0.17,0.85,0.83⟩	⟨(0.1,0.21,0.52); 0.17,0.85,0.83⟩	⟨(0.1,0.21,0.52); 0.17,0.85,0.83⟩	⟨(0.2,0.4,0.89); 0.17,0.85,0.83⟩	⟨(0.2,0.4,0.89); 0.17,0.85,0.83⟩
A_6	⟨(0.31,0.66,1.42); 0.5,0.5,0.5⟩	⟨(0.31,0.66,1.42); 0.5,0.83,0.5⟩	⟨(0.39,0.8,1.63); 0.83,0.15,0.17⟩	⟨(0.31,0.66,1.42); 0.5,0.5,0.5⟩	⟨(0.31,0.66,1.42); 0.5,0.5,0.5⟩
A_7	⟨(0.1,0.27,0.8); 0.17,0.85,0.83⟩	⟨(0.15,0.35,0.88); 0.17,0.85,0.83⟩	⟨(0.15,0.35,0.88); 0.17,0.85,0.83⟩	⟨(0.2,0.46,1.17); 0.33,0.75,0.67⟩	⟨(0.2,0.44,1.11); 0.17,0.85,0.83⟩
A_8	⟨(0.05,0.15,0.51); 0.17,0.85,0.83⟩	⟨(0.15,0.35,0.88); 0.17,0.85,0.83⟩	⟨(0.15,0.33,0.81); 0.17,0.85,0.83⟩	⟨(0.15,0.35,0.88); 0.17,0.85,0.83⟩	⟨(0.1,0.23,0.58); 0.17,0.85,0.83⟩
A_9	⟨(0.23,0.52,1.22); 0.5,0.5,0.5⟩	⟨(0.26,0.58,1.35); 0.5,0.67,0.5⟩	⟨(0.26,0.58,1.35); 0.5,0.5,0.5⟩	⟨(0.18,0.44,1.14); 0.5,0.5,0.5⟩	⟨(0.18,0.44,1.14); 0.5,0.5,0.5⟩
A_{10}	⟨(0.2,0.44,1.11); 0.17,0.85,0.83⟩	⟨(0.1,0.27,0.8); 0.17,0.85,0.83⟩	⟨(0.15,0.35,0.88); 0.33,0.75,0.67⟩	⟨(0.15,0.35,0.88); 0.17,0.85,0.83⟩	⟨(0.2,0.46,1.17); 0.33,0.75,0.67⟩

Step 6. Computation of the summation of Euclidean distances:

The expulsion impact of the each criterion is calculated using summation of Euclidean distances as given in equation 3.7.15. The results obtained are presented in Table 4.7.7.

Table 4.7.7: The expulsion impact of the criteria

C_1	$\langle(0.06,0.12,0.27); 0.17,0.85,0.83\rangle$
C_2	$\langle(0.07,0.15,0.32); 0.17,0.85,0.83\rangle$
C_3	$\langle(0.07,0.14,0.32); 0.17,0.85,0.83\rangle$
C_4	$\langle(0.06,0.16,0.21); 0.17,0.85,0.83\rangle$
C_5	$\langle(0.15,0.33,0.25); 0.17,0.85,0.83\rangle$

Step 7. Assessment of final weights of the criteria:

Table 4.7.8 shows the final weights of the criteria determined using equation 3.7.16.

Table 4.7.8: Final weights of the criteria

C_1	0.16791
C_2	0.201493
C_3	0.197761
C_4	0.160448
C_5	0.272388

Step 8. Calculate total of the weighted sum and whole of the power weight comparability sequences:

The total of the weighted sum measure (WSM) (S_i) and the whole of the weighted product measure (WPM) (P_i) of each alternative are determined for $g(x) = x$ using equations 3.7.17a and 3.7.17b, respectively. The obtained values of S_i and P_i are given in Table 4.7.9.

Table 4.7.9: WSM and WPM

Securities	S_i	P_i
A_1	$\langle(0.12,0.29,0.66);0.17,0.85,0.83\rangle$	$\langle(1.57,3.72,4.53);0.17,0.85,0.83\rangle$
A_2	$\langle(0.37,0.63,1.04);0.33,0.75,0.67\rangle$	$\langle(3.9,4.48,5.01);0.33,0.75,0.67\rangle$
A_3	$\langle(0.16,0.32,0.67);0.17,0.85,0.83\rangle$	$\langle(1.61,3.73,4.53);0.17,0.85,0.83\rangle$
A_4	$\langle(0.36,0.58,0.97);0.17,0.85,0.83\rangle$	$\langle(2.75,4.21,4.86);0.17,0.85,0.83\rangle$
A_5	$\langle(0.28,0.47,0.83);0.17,0.85,0.83\rangle$	$\langle(1.86,3.94,4.68);0.17,0.85,0.83\rangle$
A_6	$\langle(0.62,0.91,1.34);0.5,0.5,0.5\rangle$	$\langle(4.51,4.89,5.3);0.5,0.5,0.5\rangle$
A_7	$\langle(0.34,0.59,1.02);0.17,0.85,0.83\rangle$	$\langle(3.22,4.35,4.95);0.17,0.85,0.83\rangle$
A_8	$\langle(0.27,0.49,0.89);0.17,0.85,0.83\rangle$	$\langle(2.4,4.08,4.78);0.17,0.85,0.83\rangle$
A_9	$\langle(0.51,0.79,1.23);0.5,0.5,0.5\rangle$	$\langle(4.31,4.74,5.2);0.5,0.5,0.5\rangle$
A_{10}	$\langle(0.34,0.57,0.98);0.17,0.85,0.83\rangle$	$\langle(3.38,4.29,4.91);0.17,0.85,0.83\rangle$

Step 9. Relative weights or balanced compromise scores of the alternative:

Three appraisal score degrees $\left(U_i^{(1)},U_i^{(2)},U_i^{(3)}\right)$ utilized as relatives weights of alternatives are presented in Table 4.7.10.

Table 4.7.10: Appraisal score degrees

	$U_i^{(1)}$	$U_i^{(2)}$	$U_i^{(3)}$
A_1	$\langle(0.029,0.083,0.158);0.17,0.85,0.83\rangle$	$\langle(0.529,2,8.366);0.17,0.85,0.83\rangle$	$\langle(0.255,0.691,1.011);0.17,0.85,0.83\rangle$
A_2	$\langle(0.073,0.106,0.184);0.17,0.85,0.83\rangle$	$\langle(1.426,3.405,11.91);0.17,0.85,0.83\rangle$	$\langle(0.643,0.882,1.179);0.33,0.75,0.67\rangle$
A_3	$\langle(0.03,0.084,0.158);0.17,0.85,0.83\rangle$	$\langle(0.592,2.125,8.509);0.17,0.85,0.83\rangle$	$\langle(0.266,0.698,1.013);0.17,0.85,0.83\rangle$
A_4	$\langle(0.053,0.1,0.177);0.17,0.85,0.83\rangle$	$\langle(1.161,3.134,11.202);0.17,0.85,0.83\rangle$	$\langle(0.469,0.825,1.136);0.17,0.85,0.83\rangle$
A_5	$\langle(0.037,0.092,0.167);0.17,0.85,0.83\rangle$	$\langle(0.836,2.674,9.888);0.17,0.85,0.83\rangle$	$\langle(0.323,0.76,1.073);0.17,0.85,0.83\rangle$
A_6	$\langle(0.088,0.121,0.202);0.17,0.85,0.83\rangle$	$\langle(1.941,4.475,14.588);0.17,0.85,0.83\rangle$	$\langle(0.773,1,1.294);0.5,0.5,0.5\rangle$
A_7	$\langle(0.061,0.103,0.181);0.17,0.85,0.83\rangle$	$\langle(1.234,3.202,11.635);0.17,0.85,0.83\rangle$	$\langle(0.536,0.85,1.162);0.17,0.85,0.83\rangle$
A_8	$\langle(0.046,0.095,0.172);0.17,0.85,0.83\rangle$	$\langle(0.947,2.781,10.444);0.17,0.85,0.83\rangle$	$\langle(0.403,0.788,1.105);0.17,0.85,0.83\rangle$
A_9	$\langle(0.083,0.115,0.196);0.17,0.85,0.83\rangle$	$\langle(1.723,3.994,13.608);0.17,0.85,0.83\rangle$	$\langle(0.726,0.952,1.254);0.5,0.5,0.5\rangle$
A_{10}	$\langle(0.064,0.101,0.179);0.17,0.85,0.83\rangle$	$\langle(1.258,3.116,11.344);0.17,0.85,0.83\rangle$	$\langle(0.559,0.838,1.147);0.17,0.85,0.83\rangle$

Step 10. Compute the final aggregating compromise index of the alternatives and rank them:

The final ranking index (U_i) in STVN determined by using equation 3.7.19. The obtained STVN values are converted in crisp values using score function and then ranked. Table 4.7.11 displays calculated values and final ranking.

Table 4.7.11: Final ranking of the alternatives

	(U_i)	Score value	Rank
A_1	$\langle(0.272,0.963,3.623); 0.17,0.85,0.83\rangle$	0.2976	10
A_2	$\langle(0.737,1.571,5.285); 0.17,0.85,0.83\rangle$	0.4650	3
A_3	$\langle(0.298,1.011,3.681); 0.17,0.85,0.83\rangle$	0.3056	9
A_4	$\langle(0.571,1.439,4.923); 0.17,0.85,0.83\rangle$	0.4246	6
A_5	$\langle(0.402,1.238,4.302); 0.17,0.85,0.83\rangle$	0.3639	8
A_6	$\langle(0.978,2.045,6.631); 0.17,0.85,0.83\rangle$	0.5913	1
A_7	$\langle(0.624,1.478,5.144); 0.17,0.85,0.83\rangle$	0.4438	4
A_8	$\langle(0.471,1.291,4.57); 0.17,0.85,0.83\rangle$	0.3878	7
A_9	$\langle(0.878,1.832,6.132); 0.17,0.85,0.83\rangle$	0.5416	2
A_{10}	$\langle(0.642,1.439,5); 0.17,0.85,0.83\rangle$	0.4337	5

The final ranking of the stocks using the proposed model is:

$A_6 > A_9 > A_2 > A_7 > A_{10} > A_4 > A_8 > A_5 > A_3 > A_1$.

4.7.2 Validation of ranking system

To test the ranking system's validity, a portfolio (including all ten assets) is optimized using Markowitz mean-variance model with rank constraint, i.e., higher weight is allocated to high rank asset. The optimization is done using PyPortfolioOpt library in jupyter notebook. Table 4.7.12 shows the data that is used for optimization. Table 4.7.13 summarizes the output of the optimization, which include ratio allocation to the assets with and without rank constraints. The performances of the rank based and rank irrelevant portfolios are shown in Table 4.7.14. It is observe from the Table 4.7.14 that rank-based portfolio produces higher returns and also has a higher Sharpe ratio than rank-irrelevant portfolio. It demonstrates the usefulness of rank constraint, or the ranking order proposed by the ranking system.

Table 4.7.12: Data of stocks for formulation of objective function

Securities	Return	Standard deviation	Covariance										
				A_1	A_2	A_3	A_4	A_5	A_6	A_7	A_8	A_9	A_{10}
A_1	0.801	0.025	A_1	0.156	0.010	0.033	0.016	0.034	0.020	0.008	0.027	0.028	0.027
A_2	0.229	0.017	A_2	0.010	0.072	0.014	0.010	0.013	0.011	0.015	0.007	0.004	0.023
A_3	0.317	0.020	A_3	0.033	0.014	0.108	0.035	0.038	0.039	0.033	0.054	0.048	0.031
A_4	0.053	0.015	A_4	0.016	0.010	0.035	0.057	0.018	0.023	0.023	0.041	0.028	0.022
A_5	0.289	0.021	A_5	0.034	0.013	0.038	0.018	0.114	0.030	0.022	0.026	0.041	0.038
A_6	0.476	0.022	A_6	0.020	0.011	0.039	0.023	0.030	0.129	0.027	0.051	0.059	0.024
A_7	0.205	0.016	A_7	0.008	0.015	0.033	0.023	0.022	0.027	0.066	0.031	0.033	0.022
A_8	0.537	0.021	A_8	0.027	0.007	0.054	0.041	0.026	0.051	0.031	0.112	0.067	0.026
A_9	0.625	0.026	A_9	0.028	0.004	0.048	0.028	0.041	0.059	0.033	0.067	0.171	0.028
A_{10}	0.489	0.018	A_{10}	0.027	0.023	0.031	0.022	0.038	0.024	0.022	0.026	0.028	0.081

Table 4.7.13: Rank irrelevant and rank based ratio allocation to the stocks

	A_6	A_9	A_2	A_7	A_{10}	A_4	A_8	A_5	A_3	A_1
Rank irrelevant	0.057	0.015	0.255	0.184	0.064	0.267	0.0	0.066	0.00	0.088
Rank Based	0.135	0.135	0.135	0.135	0.135	0.135	0.047	0.047	0.047	0.047

Table 4.7.14: Performance of rank irrelevant and risk based portfolio

	Return	**Risk**	**Sharpe ratio**
Rank irrelevant	29.2	16.8	1.62
Rank Based	41.7	18.7	2.12

4.7.3 Comparison with other models

When compared to other methods, the annual return obtained by the proposed portfolio is the highest. Table 4.7.15 shows the comparison results.

Table 4.7.15 Comparison of proposed model with other models

Models	**Thakur *et al.* (2018)**	**Naveenan (2019)**	**Narang *et al.* (2021)**	**Narang *et al.* (2022)**	**Proposed portfolio**
Year	2016	2019	2020	2020	2021
Annual return	0.1301	0.17	0.2361	0.1672	**0.417**

Summary and Conclusions

Chapter 5 SUMMARY AND CONCLUSION

The amount of money that flows in the global financial market makes the reward of extracting exploitable regularities from financial data very attractive. There is also interest in the designing of new products or achieving a better understanding of financial dynamics. All this has resulted in efforts to gain an edge by any means. The combination of increasing processing power and the exceptional amount of financial data available have extended the range of opportunities for exploring this field. Compared to the past, the need for simplifying assumptions is lower and the share of data-driven modelling is strengthened. Researchers currently have access to computational resources that allow them to obtain, process, and analyze high frequency financial data, price complex financial products or analyze market behavior. Under these circumstances, soft computing offers a solid framework which is well suited for these tasks. Therefore, the current research work focuses on modelling complex multiple-criteria decisions such as stock selection and prediction, as well as portfolio optimization, using a variety of soft computing techniques.

Present research work comprises of following four chapters.

Chapter 1 is purely introductory and is mulled over to show fundamental concepts and foundation which are essential for the study. In this chapter, the general overview of stock market, its challenges in trading and investment, and different methodologies and concepts of stock price forecasting, decision making and portfolio optimization using various soft computing techniques are presented.

Chapter 2 entitled "Review of Literature" encapsulates the wide amount of research work that has been done in the past related to financial time series forecasting, multi-criteria decision making in stock market and portfolio optimization.

Chapter 3 is "Materials and Methods" involves the complete methodology of proposed multi-criteria decision making and stock market forecasting models using different soft computing approaches.

The models developed in this research work are as follows:

Model [1]: Deep reinforcement learning based multi-objective systems for financial trading.

Model [2]: Stock portfolio selection based on fuzzy time series forecasting using fuzzy c-means clustering and deep learning techniques.

Model [3]: An effective hybrid MCDM approach of portfolio construction using modern portfolio theory.

Model [4]: A portfolio construction model based on sector analysis using Dempster-Shafer evidence theory and Granger causal network.

Model [5]: Stock portfolio selection hybridizing fuzzy base-criterion method and evidence theory in triangular fuzzy environment.

Model [6]: A method of intraday stock selection integrating fuzzy TOPSIS and belief divergence measure in evidence theory.

Model [7]: Single valued triangular neutrosophic MEREC-CoCoSo method for multi-criteria decision making and its application in portfolio construction.

Model [1] provides a reinforcement learning based framework for multi-objective financial trading systems. The whole framework is designed with deep learning architecture to make it more reliable for practical use. Feature learning select the salient features and LSTM recurrent neural network improvise the learning of system in decision making by feeding outcome of previous actions to current state. An experiment is conducted on Nifty and Sensex real historic data for verification of the systems and compare them with conventional trading systems. Maximization of profit by adjusting risk is the motivation behind this study but other factors that affect the performance of trading system like market sentiments, future volatility can also append and it will help to get closer with real life situations. This framework can also be extended for multiple assets trading. Financial market is highly uncertain, the credibility of proposed trading systems in different market patterns in real time is still important to work in future.

Model [2] investigates a financial time series forecasting method for stock selection. A financial time series forecasting method is developed using FCM process which is trained using two deep learning techniques (SVM and MLP). A stock selection procedure is also suggested based on the forecasting results. Portfolio constructed by selected stocks is optimized using ACO algorithm. The model is illustrated for price forecasting of NSE stocks. The forecasted stock prices using MLP found very close to actual stock prices. The amalgamation of fuzzy theory and deep learning helped in improving accuracy of forecasting. The portfolio generated by forecasted data is compared to its actual data, and the estimated and actual returns are found to be very close, confirming the applicability of the presented method for investors to select appropriate stocks and estimate their future return goals.

Model [3] combines the fundamental concepts of modern portfolio theory with MCDM to create an effective portfolio. BCM and PROMETHEE I method of partial ranking are unified to avail their advantages in stock selection. Merging the concept of correlation coefficient with PROMETHEE I partial ranking competently judged the stocks over suitable preference functions for every criterion and determined a realistic relationship between the stocks which helped to construct a diversified and profitable portfolio. For preference-based weight allocation, portfolio's Sharpe ratio with rank constraint is formulated, and optimized using PSO to achieve the best solution. The proposed model is implemented on the stocks of NSE. The comparison analysis done in the study proves the veracity of the proposed method for portfolio construction. Incorporating more criteria, analyzing stocks for various preference functions and using hybrid portfolio optimization techniques can enhance the robustness of the proposed method.

Model [4] emphasizes the advantages of sector-based portfolio construction and attempts to merge all aspects of portfolio construction into a unified framework. D-S theory effectively dealt with the uncertainty in the past performances of the sectors measured by evidences and provided reliable conclusions about the future potential of the sectors. The Granger causality test serves to get diversification in the investment. Hence, strong as well as diversified sectors are identified. Leading stocks of selected sectors are easily discovered by this model to form a portfolio. The Sharpe ratio function is modified for optimization using simple deep recurrent neural network for ratio allocation to the assets of the portfolio. Thus, this model gives double risk security, first by selecting non-dependable sectors and second by optimizing risk-aware function for ratio allocation. All these factors favoured the acceptability of the proposed model over the existing models. This study is performed on the sectors of NSE; it could be implemented for portfolio construction by taking any sectors from local and global financial markets. Implementation of other MCDM techniques and incorporation of fuzzy tools could enhance the model and produce more reliable results.

Model [5] integrates the viewpoints of a beginner investor and a stock market specialist by MCDM methods for the stock selection. Both are considered as decision makers' gauging their concerns and fundamental attributes of stock selection. Fuzziness of decision makers is handled by using of triangular fuzzy numbers. The critical attributes are assessed using FDM, and the fuzzy base–criterion approach is utilised to determine their optimal weights. The D-S theory for categorizing bad, medium or good stock is proposed which effectively dealt with the ambiguity and gives reliable ranking. This study tries to alleviate a neophyte investor's fear of the stock market by demonstrating that with little specialist information, a beginner investor can make a lucrative decision. LSTM embedded deep recurrent neural network is used for optimizing Sharpe ratio to get optimal weights to portfolio assets. The application of the model on NSE data confirms its applicability.

Model [6] introduces an MCDM approach to aid the problem of stock selection for intraday trading. The presented method helps intraday trader in selecting suitable stocks for next trading day. A ranking procedure is presented that combines fuzzy TOPSIS and D-S evidence theory with a belief divergence measure to generate a credible stock preference order one day before intraday trading. Belief divergence measure thorough assessments of the collected evidences taking into account the ambiguity and significant conflict in intraday information. The triangular fuzzy arrangement of the proposed method incorporates the vagueness of stock market. Some intraday stock selection criteria that are neither beneficial nor cost criteria, i.e.

have a fixed value as desirable or undesirable value, are effectively processed by the proposed method. The suggested method generates a trustworthy preference order, as illustrated by a real-world case study. The proposed method's lengthy calculations could be a drawback for traders doing it in a short period of time, such as one day. Hence, future work can be done to reduce the model's calculations.

Model [7] presents a stock ranking method for portfolio construction hybridizing two newly developed methods MEREC and CoCoSo through some specific modifications to their main structures. The method is developed in neutrosophic environment to cope up ambiguity and inference of decision-making data. To perform this task, linguistic terms have been used which are converted into their respective single valued triangular neutrosophic numbers (STVNs). The use parabolic measure as performance measure in MEREC method makes it less computational. CoCoSo method helps in handling anomalous data and gave a combined compromised solution. The proposed hybrid STVN-MEREC-CoCoSo approach found efficient for ranking stocks as illustrated by a real case study on NSE data.

Chapter 4 entitled "Results and discussion" deals with the findings of discussed models on real data of stock market. The obtained results have been represented in tabular as well as graphical form.

5.1 Future scope

In future the developed models can be extended using different fuzzy extensions such as pythagorean, type-2, neutrosophic, intuitionistic or hesitant. The accuracy of forecasting models can be improved by using state-of-art hybrid deep learning models. Incorporating qualitative factors that influence the stock market, such as news, public sentiment, government policies, and others, can aid in forecasting stock market more accurately as well as in the development of intelligent trading systems. Other deep learning algorithms can also add significant advancement in real-time portfolio optimization.

Literature Cited

LITERATURE CITED

Aazam, S. V., Zahra, L. and Amirreza, K. 2015. Portfolio selection using ELECTRE III: Evidence from Tehran Stock Exchange. *Decis. Sci. Lett.*, 4(2): 227-236.

Abdelaziz, F. B. and Masmoudi, M. 2014. A multiple objective stochastic portfolio selection problem with random Beta. *Int. Trans. Oper. Res.*, 21(6): 919-933.

Aboussalah, A. M. and Lee, C. G. 2020. Continuous control with Stacked Deep Dynamic Recurrent Reinforcement Learning for portfolio optimization. *Expert Syst. Appl.*, 140: 112891.

Adhikari, R. and Agrawal, R. K. 2014. A combination of artificial neural network and random walk models for financial time series forecasting. *Neural. Comput. Appl.*, 24(6): 1441-1449.

Ahmed, A., Ali, R., Ejaz, A. and Ahmad, M. I. 2018. Sectoral integration and investment diversification opportunities: evidence from Colombo Stock Exchange. *Ent. Sust. Issues*, 5(3): 514–527.

Albadvi, A., Chaharsooghi, S. K. and Esfahanipour, A. 2006. Decision making in stock trading: An application of PROMETHEE. *Eur. J. Oper.*, 177(2): 673–683.

Alexander, G. J. and Resnick, B. G. 1985. Using linear and goal programming to immunize bond portfolios. *J. Bank Financ.*, 9(1): 35–54.

Al-yahyaee, K. H., Mensi, W., Al-Jarrah, I. M. W. and Tiwari, A. K. 2019. Testing for the Granger-causality between returns in the U.S. and GIPSI stock markets. *Physica A*, 531: 120950.

Anagnostopoulos, K. P. and Mamanis, G. 2010. A portfolio optimization model with three objectives and discrete variables. *Comput. Oper. Res.*, 37(7): 1285-1297.

Antonakakis, N., Cunado, J., Filis, G., Gabauer, D. and Perez de Gracia, F. 2018. Oil volatility, oil and gas firms and portfolio diversification. *Energy Econ.*, 70: 499–515.

Aouni, B., Doumpos, M., Pérez-Gladish, B. and Steuer, R. E. 2018. On the increasing importance of multiple criteria decision aid methods for portfolio selection. *J. Oper. Res. Soc.*, 69(10): 1525-1542.

Aranha, C. C. and Hitoshi, I. 2008. A tree-based ga representation for the portfolio optimization problem. *In: Proceedings of the 10th annual conference on Genetic and evolutionary computation.* July 12. pp. 873-880.

Araújo, R. D. A., Vasconcelos, G. C. and Ferreira, T. A. E. 2007. Hybrid differential evolutionary system for financial time series forecasting. *In: 2007 IEEE Congress on Evolutionary Computation*. Sep 25. pp. 4329-4336.

Armano, G., Marchesi, M. and Murru, A. 2005. A hybrid genetic-neural architecture for stock indexes forecasting. *Inf. Sciences*, 170(1): 3-33.

Atiya, A., Talaat, N. and Shaheen, S. 1997. An efficient stock market forecasting model using neural networks. *In: Proceedings of international conference on neural networks (ICNN'97)*. Jun 12. pp. 2112-2115.

Atta, E. F. E., Baafi, M. A., Amowine, N. and Zeng, K. 2020. A hybrid grey MCDM approach for asset allocation: Evidence from China's Shanghai stock exchange. *J. Bus. Econ. Manag.*, 21(2): 446–472.

Aznarte, J. L., Alcalá-Fdez, J., Arauzo-Azofra, A. and Benítez, J. M. 2012. Financial time series forecasting with a bio-inspired fuzzy model. *Expert Syst. Appl.*, 39(16):12302-12309.

Bagheri, N. 2019. Deterministic goal programming approach for Islamic portfolio selection. *Oper. Res.*, 21(3): 1447–1459.

Bas, E., Grosan, C., Egrioglu, E. and Yolcu, U. 2018. High order fuzzy time series method based on pi-sigma neural network. *Eng. Appl. Artif. Intell., 72*: 350-356.

Baydaş, M. and Pamučar, D. 2022. Determining Objective Characteristics of MCDM Methods under Uncertainty: An Exploration Study with Financial Data. *Mathematics*, 10(7): 1115.

Behzadian, M., Khanmohammadi Otaghsara, S., Yazdani, M. and Ignatius, J. 2012. A state-of the-art survey of TOPSIS applications. *Expert Syst. Appl., 39*(17): 13051-13069.

Benayoun, R., Roy, B. and Sussman, B. 1966. ELECTRE: Une méthode pour guider le choix en présence de points de vue multiples. *Note de travail*, *49*: 2-120.

Beynon, M., Cosker, D. and Marshall, D. 2001. An expert system for multi-criteria decision making using Dempster Shafer theory. *Expert Syst. Appl.*, 20(4): 357–367.

Bezdek, J. C. 1981. Pattern Recognition with Fuzzy Objective Function Algorithms. Springer Science & Business Media.

Bhat, A. A. and Kamath, S. S. 2013. Automated stock price prediction and trading framework for Nifty intraday trading. In: *4th International Conference on Computing, Communications and Networking Technologies, ICCCNT.* Jul 4. pp. 1-6.

Bhattacharyya, R., Hossain, S. A. and Kar, S. 2014. Fuzzy cross-entropy, mean, variance, skewness models for portfolio selection. *J. King Saud Univ. - Comput. Inf. Sci.*, 26(1): 79-87.

Bisht, K. and Kumar, S. 2016. Fuzzy time series forecasting method based on hesitant fuzzy sets. *Expert Syst. Appl.*, 64: 557-568.

Bisht, K. and Kumar, S. 2018. Intuitionistic Fuzzy Set-Based Computational Method for Financial Time Series Forecasting. *Fuzzy Inf. Eng.*, 10(3): 307-323.

Bouri, A., Martel, J. M. and Chabchoub, H. 2002. A multi-criterion approach for selecting attractive portfolio. *J. Multi-Criteria Decis. Anal.*, 11(4–5): 269–277.

Brans, J. P., Mareschal, B. and Vincke, P. 1984. Prométhée: a new family of outranking methods in multicriteria analysis. (No. 2013/9305). ULB-Universite Libre de Bruxelles.

Brauers, W. K. 2004. *Optimization methods for a stakeholder society: a revolution in economic thinking by multi-objective optimization* (Vol. 73). Springer Science & Business Media.

Campbell, R. 2004. *The Art of Portfolio Diversification*. LIFE working paper, (04-009).

Chan, M., Wong, C., and, C. L. C. 2000. Financial time series forecasting by neural network using conjugate gradient learning algorithm and multiple linear regression weight initialization. *Com. in Eco. and Fin.*, 61: 326–342.

Chang, T. J., Meade, N., Beasley, J. E. and Sharaiha, Y. M. 2000. Heuristics for cardinality constrained portfolio optimisation. *Comput. Oper. Res.*, 27(13): 1271–1302.

Chen, A. H. L., Liang, Y. C. and Liu, C. C. 2012. An artificial bee colony algorithm for the cardinality-constrained portfolio optimization problems. *In: IEEE Congress on Evolutionary Computation, (CEC).* June 10. pp. 1-8.

Chen, S. M. and Chung, N. Y. 2006. Forecasting enrollments using high-order fuzzy time series and genetic algorithms. *Int. J. Intell. Syst.*, 21(5):485-501.

Chen, W., Li, S. S., Zhang, J. and Mehlawat, M. K. 2020. A comprehensive model for fuzzy multi-objective portfolio selection based on DEA cross-efficiency model. *Soft Comput.*, 24(4): 2515-2526.

Cheng, C. H., Chang, J. R. and Yeh, C. A. 2006. Entropy-based and trapezoid fuzzification-based fuzzy time series approaches for forecasting IT project cost. *Tech. Fore. Soc. Chan.*, 73(5):524-542.

Cheng, C. H., Cheng, G. W. and Wang, J. W. 2008. Multi-attribute fuzzy time series method based on fuzzy clustering. *Expert Syst. Appl.*, 34(2):1235-1242.

Cheng, D., Yang, F., Xiang, S. and Liu, J. 2022. Financial time series forecasting with multi-modality graph neural network. *Pattern Recognit.*, 121: 108218.

Cheng, S. H., Chen, S. M. and Jian, W. S. 2016. Fuzzy time series forecasting based on fuzzy logical relationships and similarity measures. *Inf. Sci.*, 327: 272-287.

Cheung, M. T. and Liao, Z. 2009. Investing in real-world equity markets with an AHP-based decision framework. *J. Dec. Syst.*, *18*(2): 149–163.

Claussmann, L., O'Brien, M., Glaser, S., Najjaran, H. and Gruyer, D. 2018. Multi-Criteria Decision Making for Autonomous Vehicles using Fuzzy Dempster-Shafer Reasoning. In: *IEEE Intelligent Vehicles Symposium, Proceedings*, June 10. pp. 2195–2202.

Coelho, R., Hutzler, S., Repetowicz, P. and Richmond, P. 2007. Sector analysis for a FTSE portfolio of stocks. *Physica A*, 373: 615–626.

Colson, G. and De, B., C. 1989. An integrated multiobjective portfolio management system. In *Models and Methods in Multiple Criteria Decision Making*. pp. 1359-1381. Pergamon.

Cooper, W. W., Lelas, V. and Sueyoshi, T. 1997. Goal programming models and their duality relations for use in evaluating security portfolio and regression relations. *Eur. J. Oper.*, *98*(2): 431-443.

Corazza, M. and Sangalli, A. 2015. Q-Learning and SARSA: A Comparison between Two Intelligent Stochastic Control Approaches for Financial Trading. *University Ca'Foscari of Venice, Dept. of Economics Research Paper Series No. 15*.

Corchado, J., Fyfe, C. and Lees, B. 1998. Unsupervised learning for financial forecasting. *In: IEEE/IAFE Conference on Computational Intelligence for Financial Engineering, Proceedings (CIFEr)*. June 25. pp. 259-263.

Costa, L. F. C., Cortes, O. A. C. and Costa, J. P. A. 2019. A new adaptive meta-heuristic based on a vector evaluated approach for portfolio investment optimization. *Intel. Arti.*, 22(64):85-101.

Cura, T. 2009. Particle swarm optimization approach to portfolio optimization. *Nonlinear Analysis: Real World Appl.*, 10(4): 2396-2406.

Dempster, A. P. 1967. Upper and Lower Probabilities Induced by a Multivalued Mapping., *Springer*, 38(2): 325–339.

Deng, Y., Bao, F., Kong, Y., Ren, Z. and Dai, Q. 2017. Deep Direct Reinforcement Learning for Financial Signal Representation and Trading. *IEEE Trans. Neural Netw. Learn Syst.*, *28*(3): 653-664.

Diakoulaki, D., Mavrotas, G. and Papayannakis, L. 1995. Determining objective weights in multiple criteria problems: The critic method. *Comput. Oper. Res.*, 22(7): 763-770.

Ding, X., Zhang, Y., Liu, T. and Duan, J. 2015. Deep learning for event-driven stock prediction. *In: IJCAI International Joint Conference on Artificial Intelligence*, January 25.

Dorigo, M. 1992. Optimization, learning and natural algorithms. Thesis. Doctor of Philosophy. Politecnico di Milano, Italy.

Doumpos, M., Zopounidis, C. and Pardalos, P. M. 2000. Multicriteria sorting methodology: application to financial decision problems. *Para. Algo. App.*, 15(1–2): 113–129.

Du, J. 2022. Mean–variance portfolio optimization with deep learning based-forecasts for cointegrated stocks. *Expert Syst. Appl.*, 201: 117005.

Dutta, P. 2015. Uncertainty Modeling in Risk Assessment Based on Dempster–Shafer Theory of Evidence with Generalized Fuzzy Focal Elements. *Fuzzy Inf. Engg.*, 7(1): 15–30.

Ece, O. and Uludag, A. S. 2017. Applicability of Fuzzy TOPSIS Method in Optimal Portfolio Selection and an Application in BIST. *Int. J. Eco. Finan.*, 9(10): 107-127.

Egrioglu, E., Aladag, C. H., Yolcu, U., Uslu, V. R. and Erilli, N. A. 2011. Fuzzy time series forecasting method based on Gustafson-Kessel fuzzy clustering. *Expert Syst. Appl.*, *38*(8), 10355-10357.

Ehrgott, M., Klamroth, K. and Schwehm, C. 2004. An MCDM approach to portfolio optimization. *Eur. J. Oper.*, 155(3): 752–770.

Fazli, S. and Jafari, H. 2012. Developing a hybrid multi-criteria model for investment in stock exchange. *Manag. Sci. Lett.*, 2: 457–468.

Fei, L. and Deng, Y. 2016. Meausre divergence degree of basic probability assignment based on Deng relative entropy. *In: Proceedings of the 28th Chinese Control and Decision Conference, (CCDC*). May 28. pp. 3857-3859.

Gabus, A. and Fontela, E. 1972. World problems an invitation to further thought within the framework of DEMATEL. *Battelle Geneva Research Centre*.

Gao, X., Gao, X., Hongkong, S. and Chan, L. 2000. An Algorithm for Trading and Portfolio Management Using Q-learning and Sharpe Ratio Maximization. *In: proceedings of the international conference on neural information processing*, pp. 832-837.

García, F., Guijarro, F. and Moya, I. 2013. A multiobjective model for passive portfolio management: An application on the S&P 100 index. *J. Bus. Econ. Manag.*, 14(4): 758-775.

Goodfellow, I., Bengio, Y. and Courville, A. 2016. Deep learning. An MIT Press Book. In *Nature*.

Graves, A., Mohamed, A. R. and Hinton, G. 2013. Speech recognition with deep recurrent neural networks. *In: ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing – Proceedings.* May 26. pp. 6645-6649.

Gu, L., Guo, H. and Liu, X. 2017. Fuzzy time series forecasting based on information granule and neural network. *Int. J. Comput. Sci. Eng.*, 15(1–2): 146–152.

Gupta, K. K. and Kumar, S. 2019. Hesitant probabilistic fuzzy set based time series forecasting method. *Granul. Comput.*, 4(4): 539-558.

Gupta, P., Mehlawat, M. K. and Saxena, A. 2008. Asset portfolio optimization using fuzzy mathematical programming. *Inf. Sci.*, 178(6): 1734–1755.

Gupta, R. and Basu, P. K. 2009. Sector Analysis And Portfolio Optimisation: The Indian Experience. *Int. Bus. Eco. Res. J.*, 8(1): 119-126.

Hamed, I. M., Hussein, A. S. and Tolba, M. F. 2012. An Intelligent Model for Stock Market Prediction. *Int. J. Comput. Int. Syst.*, 5(4): 639-652.

Haseli, G., Sheikh, R. and Sana, S. S. 2020. Base-criterion on multi-criteria decision-making method and its applications. *Int. J. Manag. Sci. Eng. Manag.*, 15(2): 79–88.

Holland, J. H. 1975. Adaptation in natural and artificial systems, University of Michigan press. *Ann Arbor, MI.*

Hong, Y., Liu, Y. and Wang, S. 2009. Granger causality in risk and detection of extreme risk spillover between financial markets. *J. Econom.*, 150(2): 271–287.

Huang, C. F. 2012. A hybrid stock selection model using genetic algorithms and support vector regression. *Appl. Soft Comput.*, 12(2): 807–818.

Huang, J. J., Tzeng, G. H. and Ong, C. S. 2006. A novel algorithm for uncertain portfolio selection. *Appl. Math. Comput.*, 173(1): 350–359.

Huang, X. and Jiang, G. 2021. Portfolio management with background risk under uncertain mean-variance utility. *Fuzzy Optim. Dec. Mak.*, 20(3): 315-330.

Huarng, K. and Yu, T. H. K. 2006a. The application of neural networks to forecast fuzzy time series. *Physica A, 363*(2): 481-491.

Huarng, K. and Yu, T. H. K. 2006b. Ratio-based lengths of intervals to improve fuzzy time series forecasting. *IEEE Trans. Syst. Man Cybern. Syst.*, 36(2): 328-340.

Hussain, A. J., Knowles, A., Lisboa, P. J. G. and El-Deredy, W. 2008. Financial time series prediction using polynomial pipelined neural networks. *Expert Syst. Appl.*, 35(3): 1186-1199 .

Hwang, C. L. and Yoon, K. 1981. Multiple Attribute Decision Making Methods and Applications A State-of-the-Art Survey. *Lecture Notes in Economics and Mathematical Systems, 186*. CRC press.

Ishikawa, A., Amagasa, M., Shiga, T., Tomizawa, G., Tatsuta, R. and Mieno, H. 1993. The max-min Delphi method and fuzzy Delphi method via fuzzy integration. *Fuzzy Sets Syst.*, 55(3): 241-253.

Jerry, W. R., Tsai, C. L., Tzeng, G. H. and Fang, S. K. 2011. Combined DEMATEL technique with a novel MCDM model for exploring portfolio selection based on CAPM. *Expert Syst. Appl.*, 38(1): 16–25.

Jianchao, Y., Kai, Y., Yihong, G. and Huang, T. 2010. Linear spatial pyramid matching using sparse coding for image classification. *In: IEEE Conference on computer vision and pattern recognition*. June 20. pp. 1794-1801.

Jiang, L. and Wang, S. 2021. Robust Multi-Period And Multi-Objective Portfolio Selection. *J. Ind. Manag.t Opt.*, 17(2): 695-702.

João, R. S., Guidoni, T. F., Bertini, J. R., Nicoletti, M. D. C. and Artero, A. O. 2014. Stock closing price forecasting using ensembles of constructive neural networks. *In: Proceedings - 2014 Brazilian Conference on Intelligent Systems, (BRACIS*). Oct 18. pp. 109-114.

Jog, V., Kaliszewski, I. and Michalowski, W. 1999. Using attribute trade-off information in investment. *J. Multi-Criteria Decis. Anal.*, 8(4): 189-199.

Joshi, M. C. 2013. Factors affecting Indian stock market. *Int. J. Contem. Res. Manag. Eng. Health Sci.*, 1(2): 37-45.

Kamali, S. 2014. Portfolio Optimization Using Particle Swarm Optimization And Genetic Algorithm. *J. Math. Comp. Sci.*, 10(2): 85-90.

Karunamurthy, V. 2003. A Genetic Programming Approach to the Dynamic Portfolio Rebalancing Problem. *Genetic Algorithms and Genetic Programming at Stanford*. pp. 100-108.

Keeney, R. L. and Raiffa, H. 1976. Decisions with multiple objectives. John Wiley & Sons, New York.

Kennedy, J. and Eberhart, R. 1995. Particle swarm optimization. *In: Proceedings of ICNN'95 - International Conference on Neural Networks*, Nov 27. pp. 1942–1948.

Keršulienė, V., Zavadskas, E. K. and Turskis, Z. 2010. Selection of rational dispute resolution method by applying new step-wise weight assessment ratio analysis (SWARA). *J. Bus. Econ. Manag.*, 11(2): 243-258.

Keshavarz, G., M., Amiri, M., Zavadskas, E. K., Turskis, Z. and Antucheviciene, J. 2021. Determination of objective weights using a new method based on the removal effects of criteria (Merec). *Symmetry*, 13(4): 525-231.

Khan, A. T., Cao, X., Li, S., Hu, B. and Katsikis, V. N. 2021. Quantum beetle antennae search: a novel technique for the constrained portfolio optimization problem. *Sci. China Inf. Sci.*, 64(5), 1-4.

Khoury, N., Martel, J.-M. and Veilleux, M. 1993. Méthode multicritère de sélection de portefeuilles indiciels internationaux. *L'Act. Éco.*, *69*(1): 171–190.

Kodogiannis, V. and Lolis, A. 2002. Forecasting financial time series using neural network and fuzzy system-based techniques. *Neural. Comput. Appl.*, 11(2): 90-102.

Konno, H. and Suzuki, K. 1995. A mean-variance-skewness portfolio optimization model. *J. Oper. Res. Soc. Japan*, *38*(2): 173:187.

Korczak, J. and Hemes, M. 2017. Deep learning for financial time series forecasting in A-Trader system. *In: Proceedings of the 2017 Federated Conference on Computer Science and Information Systems*. Sep 3. pp. 905-912.

Kumar, P. C., Philippatos, G. C. and Ezzell, J. R. 1978. Goal programming and the selection of portfolios by dual-purpose funds. *J. Finan.*, 33(1): 303–310.

Lee, S. M. and Chesser, D. L. 1980. Goal programming for portfolio selection. *J. Port. Manag.*, 6(3): 22–26.

Lee, S. M. and Lerro, A. J. 1973. Optimizing the Portfolio Selection for Mutual Funds. *J. Finan.*, *28*(5): 1087:1093.

Lei, W. and Li, S. 2015. A Novel Evolutionary Algorithm with Neighborhood Search for Project Portfolios Optimization Problem. *American J. Ind. Bus. Manag.*, 05(06): 396-407.

Li, S. T., Cheng, Y. C. and Lin, S. Y. 2008. A FCM-based deterministic forecasting model for fuzzy time series. *Comp. Math. App.*, *56*(12): 3052-3063.

Li, Z., Chen, Z. and Hui, Y. 2019. Portfolio selection through Maslow's need hierarchy theory. *Appl. Econ.*, 51(4): 364-372.

Liu, Z., Yao, C., Yu, H. and Wu, T. 2019. Deep reinforcement learning with its application for lung cancer detection in medical Internet of Things. *Future Gener. Comput. Syst.*, 97: 1-9.

Loginov, A., Heywood, M. and Wilson, G. 2021. Stock selection heuristics for performing frequent intraday trading with genetic programming. *Genet. Program. Evolvable Mach.*, 22(1): 35-72.

Lopez, P., M. 2016. Building Diversified Portfolios that Outperform Out-of-Sample. *SSRN Electronic Journal*. *J. Port. Manag.*, 42(4): 59-69.

Ma, X., Wang, H. and Wang, J. 2016. Semisupervised classification for hyperspectral image based on multi-decision labeling and deep feature learning. *ISPRS J. Photogram. Remote Sens.*, 120: 99-107.

Ma, Y., Han, R. and Wang, W. 2021. Portfolio optimization with return prediction using deep learning and machine learning. *Expert Syst. Appl.*, 165: 113973.

Marasović, B. and Babić, Z. 2011. Two-step multi-criteria model for selecting optimal portfolio. *Int. J. Product. Econ.*, 134(1): 58–66.

Marisetty, N. 2021. *Design of Portfolio using Multivariate Analysis*. *Global J. Manag. Bus. Res*, 21: 12-20.

Markowitz, H. M. 1952. *Portfolio Selection: Efficient Diversification of Investments*. Yale University Press.

Masri, H. 2017. A Shariah-compliant portfolio selection model. *J. Oper. Res. Soc.*, 69(10): 1–7.

Mehlawat, M. K. 2016. Credibilistic mean-entropy models for multi-period portfolio selection with multi-choice aspiration levels. *Inf. Sci.*, 345: 9-26.

Mehta, P., Pandya, S. and Kotecha, K. 2021. Harvesting social media sentiment analysis to enhance stock market prediction using deep learning. *PeerJ Comput. Sci.*, 7: 476-*486*.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D. and Riedmiller, M. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.

Moghaddam, A. H., Moghaddam, M. H. and Esfandyari, M. 2016. Stock market index prediction using artificial neural network. *J. Econ., Finan. Administ. Sci.*, 21(41): 89-93.

Moody, J. and Saffell, M. 2001. Learning to trade via direct reinforcement. *IEEE Trans. Neural Netw.*, 12(4): 875-889.

Mukherjee, S., Sadhukhan, B., Sarkar, N., Roy, D. and De, S. 2021. Stock market prediction using deep learning algorithms. *CAAI Trans. Intell. Tech,* 4: 56-63.

Mustaffa, Z., Sulaiman, M. H. and Kahar, M. N. M. 2015. LS-SVM hyper-parameters optimization based on GWO algorithm for time series forecasting. *In: 4th International Conference on Software Engineering and Computer Systems.* Aug 19. pp. 183-188.

Narang, M., Joshi, M. C. and Pal, A. K. 2021. A hybrid fuzzy COPRAS-base-criterion method for multi-criteria decision making. *Soft Comput.*, 25(13): 8391-8399.

Narang, M., Joshi, M. C., Bisht, K. and Pal, A. 2022. Stock portfolio selection using a new decision-making approach based on the integration of fuzzy CoCoSo with Heronian mean operator. *DMAME,* 5(1): 90-112.

Naveenan, R. 2019. Risk and Return Analysis of Portfolio Management Services of Reliance Nippon Asset Management Limited (RNAM). *Global J. of Manag. Busin.*, 6(1): 108–117.

Nayak, S. C., Misra, B. B. and Behera, H. S. 2016. Efficient forecasting of financial time-series data with virtual adaptive neuro-fuzzy inference system. *Int. J. Busin. Forecast. Market. Intell.*, 2(4): 379-402

Nguyen, P. H., Tsai, J. F., Kumar, V. A. G. and Hu, Y. C. 2020. Stock investment of agriculture companies in the Vietnam stock exchange market: An AHP integrated with GRA-TOPSIS-MOORA approaches. *J. Asian Finan., Econo. Busin.*, 7(7): 113-121.

Niedermayer, A. and Niedermayer, D. 2010. Applying Markowitz's Critical Line Algorithm. *Handbook of Portfolio Construction*, 383–400.

Niu, T., Wang, J., Lu, H., Yang, W. and Du, P. 2020. Developing a deep learning framework with two-stage feature selection for multivariate financial time series forecasting. *Expert Syst. Appl.*, 148-113237.

Nti, I. K., Adekoya, A. F. and Weyori, B. A. 2020. A systematic review of fundamental and technical analysis of stock market predictions. *Artif. Intell. Review*, 53(4): 3007-3057.

Oh, K. J., Kim, T. Y. and Min, S. 2005. Using genetic algorithm to support portfolio optimization for index fund management. *Expert Syst. Appl.*, 28(2): 371-379.

Opricovic, S. 1998. Multicriteria optimization of civil engineering systems. *Faculty of Civil Engineering, Belgrade*, 2(1): 5-21.

Pamučar, D. and Ćirović, G. 2015. The selection of transport and handling resources in logistics centers using Multi-Attributive Border Approximation area Comparison (MABAC). *Expert Syst. Appl.*, 42(6): 3016-3028 .

Panigrahi, S. and Behera, H. S. 2018. A computationally efficient method for high order fuzzy time series forecasting. *J. Theor. Appl. Inf. Technol.*, 96(21): 7215-7226.

Panigrahi, S. and Behera, H. S. 2020. A study on leading machine learning techniques for high order fuzzy time series forecasting. *Eng. Appl. Artif. Intell.*, 87:103245.

Papana, A., Kyrtsou, C., Kugiumtzis, D. and Diks, C. 2017. Financial networks based on Granger causality: A case study. *Physica A*, 482: 65–73.

Pardalos, P. M., Sandström, M. and Zopounidis, C. 1994. On the use of optimization models for portfolio selection: A review and some computational results. *Comput. Econo.*, 7(4): 227-244.

Pattanayak, R. M., Panigrahi, S. and Behera, H. S. 2020. High-Order Fuzzy Time Series Forecasting by Using Membership Values Along with Data and Support Vector Machine. *Arab. J. Sci. Eng.*, 45(12): 10311-10325.

Peng, X., Zhang, X. and Luo, Z. 2020. Pythagorean fuzzy MCDM method based on CoCoSo and CRITIC with score function for 5G industry evaluation. *Artif. Intell. Review*, 53(5): 3813-3847

Peng, Y., Chen, W., Wei, P. and Yu, G. 2020. Spillover effect and Granger causality investigation between China's stock market and international oil market: A dynamic multiscale approach. *J. Comput. Appl. Math.*, 367: 112460.

Poklepović, T. and Babić, Z. 2014. Stock selection using a hybrid MCDM approach. *Croatian Oper. Res. Review*, 5(2): 273–290.

Rahiminezhad, G., M., Mokhatab Rafiei, F. and Ghezelbash, M. 2020. Portfolio selection: a fuzzy-ANP approach. *Finan. Innov.*, 6(1):1-34.

Rajan, M. P. and Mathew, J. 2012. Kalman Filter and Financial Time Series Analysis. *Communications in Computer and Information Science, 305 CCIS*, 339–351.

Rezaei, J. 2015. Best-worst multi-criteria decision-making method. *Omega*, 53: 49–57.

Rezaei, P. A., Solimanpur, M. and Jahangoshai, R. M. 2016. Solving multi-objective portfolio optimization problem using invasive weed optimization. *Swarm Evol.Comp.*, 28: 42-57.

Rubio, A., Bermúdez, J. D. and Vercher, E. 2016. Forecasting portfolio returns using weighted fuzzy time series methods. *Int. J. Approx. Reason.*, 75: 1-12.

Rubio, A., Bermúdez, J. D. and Vercher, E. 2017. Improving stock index forecasts by using a new weighted fuzzy-trend time series method. *Expert Syst. Appl*., 76: 12-20.

Rubio, A., Vercher, E. and Bermúdez, J. D. 2017. Introducing a fuzzy-pattern operator in fuzzy time series. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10306 LNCS*.

Saaty, T. L. 2005. Making and validating complex decisions with the AHP/ANP. *J. Syst. Sci. Syst. Eng.*, 14(1): 1-36.

Samaras, G. D., Matsatsinis, N. F. and Zopounidis, C. 2003. A multicriteria DSS for a global stock evaluation. *Oper. Res.*, 3(3): 281–306.

Sand, N. and Mankotia, N. 2018. Perceptual impediments to stock market investment: Factor analysis approach. *SCMS J. Indian Manag.*, 15(1): 1-14.

Sefiane, S. and Benbouzian, M. 2013. A Meta-Heuristic Ant Colony Optimization Method for Solving Portfolio Optimization. *Int. J. Finan. Manag.*, 3(4): 1-9.

Selvamuthu, D., Kumar, V. and Mishra, A. 2019. Indian stock market prediction using artificial neural networks on tick data. *Finan. Innov.*, 5(1): 1-12.

Shafer, G. 1976. A Mathematical Theory of Evidence. Princeton university press.

Sharma, A. R. and Kaushik, P. 2017. Literature survey of statistical, deep and reinforcement learning in natural language processing. *In: Proceeding - IEEE International Conference on Computing, Communication and Automation, ICCCA 2017*, May 5. pp. 350-354.

Sharma, C. and Habib, A. 2019. Mutual information based stock networks and portfolio selection for intraday traders using high frequency data: An Indian market case study. *PLoS ONE*, 14(8): 115-121.

Shoaf, J. and Foster, J. A. 1998. Efficient set GA for stock portfolios. *In*: *Proceedings of the IEEE Conference on Evolutionary Computation*. May 4. pp. 354-359.

Si, W., Li, J., Rao, R. and Ding, P. 2018. A multi-objective deep reinforcement learning approach for stock index futures's intraday trading. *In: Proceedings - 2017 10th International Symposium on Computational Intelligence and Design*, Oct 9. pp. 431–436.

Silva, L. G. D. O. and De Almeida-Filho, A. T. 2016. A multicriteria approach for analysis of conflicts in evidence theory. *Inf. Sci.*, 346: 275–285.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T. and Hassabis, D. 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529: 484-489.

Singh, P. and Borah, B. 2013a. An efficient time series forecasting model based on fuzzy time series. *Eng. Appl. Artif. Intell.*, 26(10): 2443-2457.

Singh, P. and Borah, B. 2013b. High-order fuzzy-neuro expert system for time series forecasting. *Knowl-Based Syst.*, 46: 12-21.

Singh, S., Jaakkola, T., Littman, M. L. and Szepesvári, C. 2000. Convergence results for single-step on-policy reinforcement-learning algorithms. *Mach. Learn.*, 38(3): 287-308.

Song, Q. and Chissom, B. S. 1993. Forecasting enrollments with fuzzy time series - part I. *Fuzzy Sets Syst.*, 54: 1-9.

Song, Q. and Chissom, B. S. 1994. Forecasting enrollments with fuzzy time series - part II. *Fuzzy Sets Syst*., 62: 1-8.

Song, Y. and Deng, Y. 2019. Divergence Measure of Belief Function and Its Application in Data Fusion. *IEEE Access, 7:* 107465-107472.

Stawiarski, B. 2021. Granger Causality and Cointegration During Stock Bubbles and Market Crashes. *Appl. Condi. Monitor.*, 18: 93–107.

Street, B. and Lawley, M. 2014. Modelling and Analysis on Noisy Financial Time Series. *J. Comput. Comm.*, 2(2): 64–69.

Syamala, R. P., Parthasaradhi, V. G. and Durga P. C. 2022. Financial time series forecasting using optimized multistage wavelet regression approach. *Int. J. Inf. Tech.(Singapore)*, 14(4): 2231–2240.

Tang, Y., Xiong, J. J., Luo, Y. and Zhang, Y. C. 2019. How Do the Global Stock Markets Influence One Another? Evidence from Finance Big Data and Granger Causality Directed Network., *I. J. Elec. Commerce*, 23(1): 85-109.

Tarsauliya, A., Kala, R., Tiwari, R. and Shukla, A. 2011. Financial time series forecast using neural network ensembles. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6728 LNCS*(PART 1).

Tarsauliya, A., Kant, S., Kala, R., Tiwari, R. and Shukla, A. 2010. Analysis of Artificial Neural Network for Financial Time Series Forecasting. *Int. J. Comput. App.*, 9(5), 16–22.

Tesauro, G. 1994. TD-Gammon, a Self-Teaching Backgammon Program, Achieves Master-Level Play. *Neural Comput.*, 6(2): 215-219.

Thakur, G. S., Bhattacharyya, R. and Sarkar, S. 2018. Stock portfolio selection using Dempster–Shafer evidence theory. *J. King Saud Univ. - Comput. Inf. Sci.*, 30(2): 223–235.

Tiryaki, F. and Ahlatcioglu, M. 2005. Fuzzy stock selection using a new fuzzy ranking and weighting algorithm. *Appl. Math. Comput.*, 170(1), 144–157.

Tzanetos, A., Vassiliadis, V. and Dounias, G. 2017. A novel hybrid nature-inspired scheme for solving a financial optimization problem. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 10334 LNCS*.

Vapnik, V. N. 1995. The Nature of Statistical Learning Theory. Springer science & business media.

Varma, K. and Sunil Kumar, K. 2012. Criteria analysis aiding portfolio selection using dematel. *Procedia Eng.*, 38: 3649-3661.

Vlasenko, A., Vlasenko, N., Vynokurova, O., Bodyanskiy, Y. and Peleshko, D. 2019. A novel ensemble neuro-fuzzy model for financial time series forecasting. *Data*, *4*(3): 126-132.

Vuković, M., Pivac, S. and Babić, Z. 2020. Comparative analysis of stock selection using a hybrid MCDM approach and modern portfolio theory. *Croatian Review Econ. Bus. Social Stat.*, 6(2): 58-68.

Výrost, T., Lyócsa, Š. and Baumöhl, E. 2015. Granger causality stock market networks: Temporal proximity and preferential attachment. *Physica A*, 427: 262–276.

Walczak, S. 2001. An empirical analysis of data requirements for financial forecasting with neural networks. *J. Manag. Inf. Syst.*, 17(4): 203-222.

Wang, B., Li, Y., Wang, S. and Watada, J. 2018. A Multi-Objective Portfolio Selection Model with Fuzzy Value-At-Risk Ratio. *IEEE Trans. Fuzzy Syst.*, 26(6): 3673-3687.

Wang, H., Deng, X., Jiang, W. and Geng, J. 2021. A new belief divergence measure for Dempster–Shafer theory based on belief and plausibility function and its application in multi-source data fusion. *Eng. Appl. Artif. Intell.*, 97: 104030.

Wang, X., Phua, P. K. H. and Lin, W. 2003. Stock Market Prediction Using Neural Networks: Does Trading Volume Help in Short-term Prediction? *In: Proceedings of the International Joint Conference on Neural Networks*, Jul 20. pp. 2438-2442.

Wen, Z., Liao, H., Kazimieras Zavadskas, E. and Al-Barakati, A. 2019. Selection third-party logistics service providers in supply chain finance by a hesitant fuzzy linguistic combined compromise solution method. *Econ. Res.-Ekonomska Istrazivanja* , 32(1): 4033-4058.

Wong, H. L., Tu, Y. H. and Wang, C. C. 2010. Application of fuzzy time series models for forecasting the amount of Taiwan export. *Expert Syst. Appl.* 37(2): 1465-1470.

Xidonas, P. and Psarras, J. 2009. Equity portfolio management within the MCDM frame: A literature review. *Int. J. Bank. Account. Finan..* 1(3):285–309.

Xidonas, P., Mavrotas, G., Zopounidis, C. and Psarras, J. 2011. IPSSIS: An integrated multicriteria decision support system for equity portfolio construction and selection. *Eur. J. Oper.*, 210(2): 398–409.

Yaakob, A. M. and Gegov, A. 2016. Interactive TOPSIS Based Group Decision Making Methodology Using Z-Numbers. *Int. J. Com. Intell. Syst.*, *9*(2): 311-324.

Yazdani, M., Wen, Z., Liao, H., Banaitis, A. and Turskis, Z. 2019. A grey combined compromise solution (CoCoSo-G) method for supplier selection in construction management. *J. Civil Eng. Manag.*, 25(8), 858-874.

Yazdani, M., Zarate, P., Kazimieras Zavadskas, E. and Turskis, Z. 2019. A combined compromise solution (CoCoSo) method for multi-criteria decision-making problems. *Manag. Dec.*, 57(9): 2501–2519.

Ye, F., Zhang, L., Zhang, D., Fujita, H. and Gong, Z. 2016. A novel forecasting method based on multi-order fuzzy time series and technical analysis. *Inf. Sci.*, 367: 41-57.

Yodmun, S. and Witayakiattilerd, W. 2016. Stock Selection into Portfolio by Fuzzy Quantitative Analysis and Fuzzy Multicriteria Decision Making. *Advances in Oper. Res.*, 20: 116-124.

Yu, L., Wang, S. and Lai, K. K. 2009. A neural-network-based nonlinear metamodeling approach to financial time series forecasting. *Appl. Soft Comp.*, 9(2), 563-574.

Yunusoglu, M. G. and Selim, H. 2013. A fuzzy rule based expert system for stock evaluation and portfolio construction: An application to Istanbul Stock Exchange. *Expert Syst. Appl.*, 40(3): 908–920.

Zadeh, L. A. (1965). Fuzzy sets. *Inf. Control*. 8(3):338-353.

Zavadskas, E. and Kaklauskas, A. S. V. 1994. The new method of multicriteria complex proportional assessment of projects. *Tech. Econ. Develop. Econ.*, 1: 131–139.

Zavadskas, E. K., Turskis, Z., Antucheviciene, J. and Zakarevicius, A. 2012. Optimization of weighted aggregated sum product assessment. *Elektronika Ir Elektrotechnika*, 122(6): 3-6.

Zhang, S. S. 2018. Need for speed: Hard information processing in a high-frequency world. *J. Futures Mark.*, 38(1): 3-21.

Zhang, Z., Zohren, S. and Roberts, S. 2020. Deep Learning for Portfolio Optimization. *The J. Finan. Data Sci.*, 2(4): 8–20.

Zhao, Y., Stasinakis, C., Sermpinis, G. and Shi, Y. 2018. Neural network copula portfolio optimization for exchange traded funds. *Quan. Finan.*, 18(5): 761-775.

Zhora, D. V. 2004. Financial forecasting using Random Subspace Classifier. *In: IEEE International Conference on Neural Networks- Conference Proceedings*. Jul 25. pp. 2735-2740.

Zopounidis, C. and Doumpos, M. 2013. Multicriteria decision systems for financial problems. *TOP*, 21(2): 241-261.

Zou, K., Zhou, X. and Li, B. 2011. Stock-selection decision method based on AHP and TOPSIS algorithm. ICIC Express Letters.

CURRICULUM VITAE

Name	:	Kiran Bisht	**Phone No.**	:	9711341338
Mailing Address	:	House No. E-19, Bukhara Colony, Bijnor, Uttar Pradesh Pin-246701	**Permanent Address**	:	House No. E-19, Bukhara Colony, Bijnor, Uttar Pradesh Pin-246701
E-mail	:	kiranbisht96@gmail.com			

- **Career Objective:** To work in such organization where I can sharpen my skills and learn new things to explore the new field in my subject and utilize my insight, energy and imagination to give something new to my association.
- **Educational Qualification:**

S.No.	Examination passed	Institution	Year	Percentage/ CGPA
1.	M.Sc. (Mathematics)	Vardhman College, Bijnor (M.J.P.R. University)	2015	62.9%
2.	B.Sc.	Vardhman College, Bijnor (M.J.P.R. University)	2013	70.0%
3.	Intermediate	K. P. S. Kanya Inter College, Bijnor (U. P. Board)	2012	83.6%
4.	High School	K. P. S. Kanya Inter College, Bijnor (U. P. Board)	2010	77.8%

- **Specialization**: Financial mathematical modeling, time series forecasting, decision making
- **Publications**:

Paper published/accepted from thesis:

1. **Bisht, K. and Kumar, A. (2021).** A method for fuzzy time series forecasting based on interval index number and membership value using fuzzy c-means clustering. *Evolutionary Intelligence*. https://doi.org/10.1007/s12065-021-00656-0
2. **Bisht, K. and Kumar, A. (2020).** Deep Reinforcement Learning based Multi-Objective Systems for Financial Trading. *In: proc.* 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering *(ICRAIE)*, 2020, pp. 1-6, doi: 10.1109/ICRAIE51050.2020.9358319.
3. **Bisht, K. and Kumar, A.** Stock portfolio selection hybridizing fuzzy base-criterion method and evidence theory in triangular fuzzy environment. (Accepted- Operations Research Forum Springer)
4. **Bisht, K. and Kumar, A.** A method of intraday stock selection integrating fuzzy TOPSIS and belief divergence measure in evidence theory. (Accepted- IEEE)

Papers communicated from thesis

1. **Bisht, K. and Kumar, A.** An effective hybrid MCDM approach of portfolio construction using modern portfolio theory.
2. **Bisht, K. and Kumar, A.** Stock portfolio selection based on fuzzy time series forecasting using fuzzy c-means clustering and deep learning techniques.
3. **Bisht, K. and Kumar, A.** Single valued triangular neutrosophic MEREC-CoCoSo method for multi-criteria decision making and its application in portfolio construction.

- **Conference attended:**
  1. Presented a paper (oral presentation) entitled "Deep Reinforcement Learning based Multi-Objective Systems for Financial Trading" in *5th IEEE International Conference on Recent Advances and Innovations in Engineering (IEEE-ICRAIE-2020)* held at Poornima College of Engineering, Jaipur, India (December 1-3).
  2. Presented a paper (oral presentation) entitled "Stock portfolio selection hybridizing fuzzy base-criterion method and evidence theory" in International Conference on Contemporary Research on Mathematics and Computer Science (ICCRMCS-2022) held at Shri Ram Murti Smarak College of Engineering & Technology, Bareilly, India (April 29-30).
  3. Presented a paper (oral presentation) entitled "Fuzzy TOPSIS and evidence theory based multi-criteria decision making method for intraday stock selection" in 2022 IEEE World Conference on Applied Intelligence and Computing (AIC 2022) held at Rajkiya Engineering College, Sonbhadra, India (April 17-19).
- **Skills:** Modelling of financial time series forecasting, modelling of computational algorithms and soft computing techniques, working with Python programming language, artificial intelligence and machine learning tools, working with Eviews software and paper writing software: Latex, Mendeley, Word, Excel and Power Point.
- **Awards/ Honors/ Achievements:**
  1. CSIR-NET (JRF) awardee during Ph.D.
  2. INPIRE fellowship awardee during B.Sc. and M.Sc.

Place: Pantnagar

Date: July, 2022

(Kiran Bisht)

Evolutionary Intelligence
https://doi.org/10.1007/s12065-021-00656-0

RESEARCH PAPER

A method for fuzzy time series forecasting based on interval index number and membership value using fuzzy c-means clustering

Kiran Bisht[1] · Arun Kumar[1]

Received: 5 January 2021 / Revised: 26 July 2021 / Accepted: 6 August 2021

Abstract
Fuzzy time series forecasting methods are very popular among researchers for predicting future values as they are not based on the strict assumptions of traditional forecasting methods. Non-stochastic methods of fuzzy time series forecasting are preferred by the researchers over the years because these methods are capable to deal with real life uncertainties and provide significant forecast. There are generally, four factors that determine the performance of the forecasting method (1) number of intervals (NOIs) and length of intervals to partition universe of discourse (UOD), (2) fuzzification rules or feature representation of crisp time series, (3) method of establishing fuzzy logic rule (FLRs), (4) defuzzification rule to get crisp forecasted value. Considering, first two factors to improve the forecasting accuracy, we proposed a modified non-stochastic method of fuzzy time series forecasting in which interval index number and membership value are used as input features to predict future value. We suggested a rounding-off range and large step-size method to find the optimal NOIs and used fuzzy c-means clustering process to divide UOD into intervals of unequal length. We implement two techniques (1) regression by support vector machine and (2) neural network by multilayer perceptron to establish FLRs. To test our proposed method by both techniques we conduct a simulated study on eight widely used real time series and compare the performance with some recently developed models. Two performance measures RSME and SMAPE are used for performance analysis and observed better forecasting accuracy by the proposed model.

Keywords Fuzzy time series forecasting (FTSF) · Fuzzy c-means clustering (FCM) · Number of intervals (NOIs) · Support vector machine (SVM) · Multilayer perceptron (MLP)

1 Introduction

Forecasting future values is very common practice in day-to-day life. Forecasting based on past observed values is known as time series forecasting. Both stochastic and non-stochastic methods are employed by researchers for time series forecasting in past years. Stochastic methods like moving average (MA), autoregressive integrated moving average (ARIMA), vector regression and exponential moving average (EMA) based models have limitations in handling complex and highly uncertain real world forecasting problems. Due to these limitations, non-stochastic methods are preferred over stochastic methods. Fuzzy time series forecasting (FTSF) models are highly prevalent in research field because of linguistic representation these methods more closely illustrate real world scenarios and generally give better results than traditional methods.

Zadeh [1] presented the theory of fuzzy sets. Song and Chissom [2–4] were the first to introduce time series forecasting models based on fuzzy sets. They developed the models using min-max composition operation on fuzzy sets that have cumbersome computational process which was improved by Chen [5] using simple arithmetic operations. The whole process of fuzzy time series forecasting has four major steps as follows:

1. Determining universe of discourse (UOD), number of intervals (NOIs) and length of intervals to divide UOD.
2. Obtaining fuzzy sets and fuzzifying the time series.

✉ Kiran Bisht
kiranbisht96@gmail.com

Arun Kumar
arun_pal1969@yahoo.co.in

[1] Department of Mathematics, Statistics and Computer Science, G. B. Pant University of Agriculture and Technology, Pantnagar, Uttarakhand 263145, India

Published online: 16 August 2021

Deep Reinforcement Learning based Multi-Objective Systems for Financial Trading

Kiran Bisht
Dept. of Mathematics, Statistics and Computer Science
G. B. Pant University of Agriculture and Technology
Pantnagar, Uttarakhand, 263145, India
kiranbisht96@gmail.com

Arun Kumar
Dept. of Mathematics, Statistics and Computer Science
G. B. Pant University of Agriculture and Technology
Pantnagar, Uttarakhand, 263145, India
arun_pal1969@yahoo.co.in

2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE) | 978-1-7281-8867-6/20/$31.00 ©2020 IEEE | DOI: 10.1109/ICRAIE51050.2020.9358319

***Abstract*— Because of the risky nature of stock market, most people do not feel a secure option to invest their money in financial trading. Focusing on this basic concern of investors much research efforts are devoted to develop automated trading systems that make intelligent decisions according to the market situation and help investor to make profit beside risk. In contrast, in this paper we proposed multi-objective systems based on deep reinforcement learning for stock trading. Target of the multi-objective systems is to get maximum profit by adjusting risk. We design the whole structure of systems consisting two deep neural networks first is LSTM autoencoder for robust feature extraction and second deep reinforcement learning with LSTM recurrent neural network for decision making in order to achieve the investor's goal. We conduct an experiment on real historic data for verification of the systems and compare them with conventional trading systems.**

Keywords*— *Reinforcement learning (RL), Deep Learning (DL), Long Short Term Memory (LSTM), Financial trading.

I. Introduction

With the evolution of artificial intelligence and machine learning, researchers have been developing intelligent trading systems to trade financial assets and deep learning algorithms making the training process robust [1]. The basic objective of trading systems is to maximize total return and return depends on risk. Investor can increase total return by taking higher risk but it could lead to great loss also and if he takes lesser risk he has to compromise with total return too. Risk plays crucial role in trading environment, so there is need to develop such trading systems that make maximum possible profit while restraining the risk. Therefore, making most precise decision is very important under dynamic market conditions. Reinforcement learning (RL) algorithms [2] have attracted researchers to train trading systems as in reinforcement learning agent explore unknown environmental states and choose most favourable action by past experiences [3]. The learning agent takes input as different price patterns and output discrete trading action signals that optimize the profit and control the risk.

Training the RL agent over large datasets helps agent to inspect broader market situations. The hierarchical functioning of deep learning enables agent to process the vast amount of unstructured datasets [4]. Supervised and unsupervised methods of deep learning are utilized by researchers in various fields [5]. The blend of deep learning (DL) and reinforcement learning (RL) i.e. Deep reinforcement learning (DRL) [6] has blown the mind of researchers by its remarkable success in Alpha Go [7], playing Atari games [8], natural language processing [9], health care [10], [11] and many more. Using DRL for training the trading systems can improves its performance so that it can make better decision during trading.

The problem of dealing with large datasets overcome with deep learning but the other challenge is the noise and uncertainty present in the data received in online manner. To discover intricate structures and extract relevant features, Deep feature learning techniques are quite promising. Deep feature learning not only extract features but also improve accuracy, reduce over fitting, speed up training and improve data visualization. Therefore, firstly deep feature learning ensures robust key features summarization and extraction directly from the data and then deep reinforcement learning ensures robustness of decision making process.

Now as there are different types of stock in the financial market having different leverages, market patterns and investment policies. To understand the market position and pattern of a stock, it is required to enquire it with different parameters. So, in the context of dealing with different stocks and achieving investor's ultimate goal, in this paper we discuss two different multi-objective structures and make a comparison between them. Each model comprises two fully connected deep neural networks. First neural network is LSTM (long short term memory) autoencoder [12] for deep feature learning and second neural network is recurrent LSTM network [13] for deep reinforcement learning.

The rest of this paper is structured as follows. Related work briefly discussed in Section II. The proposed multi-objective structures are introduced in Section III and deep neuralnetwork setup for proposed model is presented in Section IV. Verification of proposed trading systems on realdata is performed in Section V. At last conclusion and a few future directions are indicated in Section VI.

II. Related Work

Trading stocks can make significant profit if we learn and improve trading strategies. Various methods of forecasting financial variables like stock prices, stock exchange indices and prices of different financial derivatives were proposed to make wise trading decisions [14], [15], [16]. These methods were based on supervised learning. But supervised learning is unable to examine the effect on total profit/loss by previous trading decisions. Then after the development of reinforcement learning [2] researchers found more effective technique to train the trading systems [17], [18], [19].

978-1-7281-8867-6/20/$31.00 ©2020 IEEE

Name	:	Kiran Bisht	**Id. No.**	:	53998
Sem. and year of admission	:	I, 2018-2019	**Degree**	:	Ph.D.
Department	:	Mathematics, Statistics and Computer Science			
Major	:	Mathematics	**Minor**	:	Computer Science
Thesis Title	:	**“FORECASTING AND MULTI-CRITERIA DECISION MAKING FOR TRADING AND INVESTMENT IN STOCK MARKET USING SOFT COMPUTING TECHNIQUES”**			
No. of Pages	:	202	**Advisor**	:	Dr. Arun Kumar

ABSTRACT

During the last decades the globalization of economies, intensifying competition among firms, financial institutions and organizations as well as the rapid economic, social and technological changes, have led to an increasing uncertainty and instability in the financial market. Within this context, the importance of making efficient financial decisions while handling complexities has increased. Soft computing covers a wide range of techniques offering tolerance to the presence of uncertainty and imprecision, making them ideal for modelling financial decisions. In light of the foregoing facts, the present research is centered to propose financial decision making models based on soft computing techniques which include fuzzy sets, neural networks, and evolutionary algorithms. In this study, seven models are introduced. **Model [1]** presents multi-objective financial trading systems based on deep reinforcement learning. The systems have been designed to give signals (buy, hold or sell) in the live market in order to get maximum return with minimum risk. Reward-based deep reinforcement learning architectures have been used to simulate the models for gaining better results. **Model [2]** proposes a financial time series forecasting method for stock selection. Fuzzy time series forecasting procedure using fuzzy c-means clustering and two deep learning architectures (SVM and MLP) has been developed for forecasting daily stock prices. Portfolio construction based on stock prices and ACO algorithm for portfolio optimization is derived. **Model [3]** considers the concept of diversification addressed by modern portfolio theory and suggests a hybrid MCDM technique for diverse stock selection. Neutrosophic base-criteria method has been devised for criteria weight assessment. The concept of correlation coefficient between assets with PROMETHEE partial ranking are merged for finding realistic relationship between the stocks to construct a diversified and profitable portfolio. PSO algorithm for optimizing Portfolio’s Sharpe ratio with rank constraint has been derived. **Model [4]** provides an effective portfolio construction method based on sector analysis. Dempster-Shafer theory and Granger causality test have been employed for indentifying strong and diverse sector of economy. Construction of portfolio by picking leading stocks of strong and diverse sectors and its optimization using deep recurrent neural network is presented. **Model [5]** offers a stock selection method integrating the concerns of a novice investor and a stock market specialist. An integrated framework unifying fuzzy delphi method, fuzzy base-criterion method and Dempster-Shafer theory has been developed for assessing important criteria, their weights and reliable ranking of stocks. Portfolio optimization is done using LSTM embedded deep recurrent neural network. **Model [6]** introduces fuzzy TOPSIS and evidence theory based intraday stock selection procedure. To counter the ambiguity and conflict in intraday data belief divergence measure has been employed for credible ranking of intraday stocks one day before trading. **Model [7]** presents an MCDM method hybridizing MEREC and CoCoSo method through some specific modifications to their main structures in context to its application in ranking stocks. The method has been developed in neutrosophic environment to cope up ambiguity and inference of decision making data. Parabolic measure has been used as performance measure in MEREC method to reduce its complexity. All the present models have been implemented on real data of Indian stock market (NSE and BSE) and detailed analysis have been done to verify their practicality.

Kiran Bisht.

(Arun Kumar)
Advisor

(Kiran Bisht)
Authoress

नाम	:	किरन बिष्ट	परिचायंक सं.	:	53998
सत्र एवं प्रवेश वर्ष	:	प्रथम, 2018-2019	उपाधि	:	पीएच.डी.
विभाग	:	गणित, सांख्यिकी एवं संगणक विज्ञान			
मुख्य विषय	:	गणित	गौण विषय	:	संगणक विज्ञान
शोध शीर्षक	:	"सॉफ्ट कंप्यूटिंग तकनीकों का उपयोग करके स्टॉक मार्केट में व्यापार और निवेश के लिए पूर्वानुमान और बहु-मापदंड निर्णय"			
पृष्ठों की संख्या	:	202	सलाहकार	:	डॉ. अरुण कुमार

सारांश

पिछले दशकों के दौरान अर्थव्यवस्थाओं के वैश्वीकरण, उद्योगों, वित्तीय संस्थानों और संगठनों के साथ-साथ आर्थिक, सामाजिक और तकनीकी परिवर्तनों के बीच तीव्र प्रतिस्पर्धा ने वित्तीय बाजार में बढ़ती अनिश्चितता और अस्थिरता को जन्म दिया है। इस संदर्भ में, जटिलताओं से निपटने के दौरान कुशल वित्तीय निर्णय लेने का महत्व बढ़ गया है। सॉफ्ट कंप्यूटिंग अनिश्चितता और अशुद्धि की उपस्थिति के प्रति सहिष्णुता प्रदान करने वाली तकनीकों की एक विस्तृत श्रृंखला है, जो उन्हें वित्तीय निर्णयों के मॉडलिंग के लिए आदर्श बनाती है। उपरोक्त तथ्यों के संदर्भ में, वर्तमान शोध सॉफ्ट कंप्यूटिंग तकनीकों के आधार पर वित्तीय निर्णय लेने के विभिन्न मॉडल का प्रस्ताव करने के लिए केंद्रित है जिसमें फ़ज़ी सेट, न्युरल नेटवर्क और एवोलुशनरी एल्गोरिदम शामिल हैं। इस अध्ययन में सात मॉडल पेश किए गए हैं। मॉडल [1] डीप रिइंफोर्समेंट लर्निंग के आधार पर मल्टी- ऑब्जेक्टिव फाइनेंसियल ट्रेडिंग सिस्टम प्रस्तुत करता है। सिस्टम को न्यूनतम जोखिम के साथ अधिकतम रिटर्न प्राप्त करने के लिए लाइव मार्केट में सिग्नल (खरीद, होल्ड या बेच) देने के लिए डिज़ाइन किया गया है। बेहतर परिणाम प्राप्त करने के लिए मॉडल का अनुकरण पुरस्कार-आधारित डीप रिइंफोर्समेंट लर्निंग आर्किटेक्चर का उपयोग किया गया है। मॉडल [2] स्टॉक चयन के लिए एक फाइनेंसियल टाइम सीरीज़ फॉर्कैस्टिंग पद्धति का प्रस्ताव करता है। फ़ज़ी सी-मीन्स क्लस्टरिंग और दो डीप लर्निंग आर्किटेक्चर (एसवीएम और एमएलपी) का उपयोग करके फ़ज़ी टाइम सीरीज़ फॉर्कैस्टिंग प्रक्रिया को दैनिक स्टॉक कीमतों के पूर्वानुमान के लिए विकसित किया गया है। स्टॉक की कीमतों के आधार पर पोर्टफोलियो निर्माण और पोर्टफोलियो ऑप्टिमाइज़ेशन के लिए एसीओ एल्गोरिथम व्युत्पन्न किया गया है। मॉडल [3] आधुनिक पोर्टफोलियो सिद्धांत द्वारा संबोधित विविधीकरण की अवधारणा पर विचार करता है और विविध स्टॉक चयन के लिए एक हाइब्रिड एमसीडीएम तकनीक का सुझाव देता है। मानदंड वजन मूल्यांकन के लिए न्यूट्रोसोफिक बेस-क्राइटिरीअन विधि तैयार की गई है। प्रोमेथी आंशिक रैंकिंग वाली संपत्तियों के बीच सहसंबंध गुणांक की अवधारणा को एक विविध और लाभदायक पोर्टफोलियो के निर्माण के लिए शेयरों के बीच यथार्थवादी संबंध खोजने के लिए मिला दिया गया है। रैंक प्रतिबंध के साथ पोर्टफोलियो के शार्प अनुपात को ऑप्टिमाइज़ करने के लिए पीएसओ एल्गोरिथम प्रयोग किया गया है। मॉडल [4] सेक्टर विश्लेषण के आधार पर एक प्रभावी पोर्टफोलियो निर्माण पद्धति प्रदान करता है। अर्थव्यवस्था के मजबूत और विविध क्षेत्र की पहचान के लिए डेम्पस्टर-शाफर सिद्धांत और ग्रेंजर कॉसलिटी परीक्षण को नियोजित किया गया है। मजबूत और विविध क्षेत्रों के प्रमुख शेयरों को चुनकर पोर्टफोलियो का निर्माण और डीप रिकरन्ट न्युरल नेटवर्क का उपयोग करके इसका अनुकूलन प्रस्तुत किया गया है। मॉडल [5] एक नौसिखिया निवेशक और एक शेयर बाजार विशेषज्ञ की चिंताओं को एकीकृत करते हुए एक स्टॉक चयन पद्धति प्रदान करता है। महत्वपूर्ण मानदंड, उनके भार और स्टॉक की विश्वसनीय रैंकिंग का आकलन करने के लिए फ़ज़ी डेल्फ़ी पद्धति, फ़ज़ी बेस-क्राइटिरीअन पद्धति और डेम्पस्टर-शेफ़र सिद्धांत को एकीकृत करने वाला एक ढांचा विकसित किया गया है। पोर्टफोलियो अनुकूलन एलएसटीएम एम्बेडेड डीप रिकरन्ट न्युरल नेटवर्क का उपयोग करके किया गया है। मॉडल [6] फ़ज़ी टॉपसिस और एविडन्स सिद्धांत आधारित इंट्राडे स्टॉक चयन प्रक्रिया का परिचय देता है। इंट्राडे डेटा में अस्पष्टता और संघर्ष का मुकाबला करने के लिए ट्रेडिंग से एक दिन पहले इंट्राडे स्टॉक की विश्वसनीय रैंकिंग के लिए बिलीफ डाइवर्जन्स मेशर को नियोजित किया गया है। मॉडल [7] एक एमसीडीएम पद्धति प्रस्तुत करता है जो मेरेक और कोकोसो पद्धति में कुछ विशिष्ट संशोधनों के माध्यम से स्टॉक रैंकिंग में इसके प्रयोग के संदर्भ में उनके मुख्य ढांचे में संकरण करता है। निर्णय लेने वाले डेटा की अस्पष्टता और अनुमान का सामना करने के लिए विधि को न्यूट्रोसोफिक वातावरण में विकसित किया गया है। इसकी जटिलता को कम करने के लिए मेरेक पद्धति में प्रदर्शन माप के रूप में परवलयिक माप का उपयोग किया गया है। सभी मौजूदा मॉडल भारतीय शेयर बाजार (एनएसई और बीएसई) के वास्तविक आंकड़ों पर लागू किए गए हैं और उनकी व्यावहारिकता को सत्यापित करने के लिए विस्तृत विश्लेषण किया गया है।

(अरुण कुमार)
सलाहकार

(किरन बिष्ट)
लेखिका