Prediction of Future Stock Trends by Using Two-Stage Hierarchical Systems Based on the Segment Categorization and Recognition of Series of Category Symbols Using the Genetic Programming

時永, 祥三
九州大学大学院経済学研究院

岸川, 善紀
The Faculty of System Science and Technology, Akita Prefectural University

https://doi.org/10.15017/7632

出版情報：經濟學研究. 72 (2/3), pp.21-43, 2005-12-08. 九州大学経済学会
バージョン：
権利関係：

Prediction of Future Stock Trends by Using Two-Stage Hierarchical Systems Based on the Segment Categorization and Recognition of Series of Category Symbols Using the Genetic Programming

Shozo Tokinaga and Yoshinori Kishikawa

1 Introduction

In the investment decision of securities market, traders usually use fundamental analysis for selecting appropriate stocks, and simultaneously they use the technical analysis of financial time series to determine the time for buy/sell stocks. For example, traders expect a rise of stock price if some shapes of triangle are found in the stock prices. It may be valuable to formalize these kinds of knowledge (rules) by expertise systematically, and to utilize these rules for investment decision.

This paper deals with the prediction of future stock trends by using two-stage hierarchical systems based on the segment categorization and recognition of series of category symbols using the Genetic Programming (GP) [18]. We apply the GP procedure in learning phase of the first stage system where we improve the nonlinear functional forms to approximate the models used to generate time series [2]-[4][9]-[13] [22][25]. We also use the GP procedure in the second stage system to generate rules for predicting future stock trends by using the series of category symbols of time series segments.

Among conventional methods for pattern recognition, hierarchical systems are widely used to improve the classification and to simplify the overall system configuration by separating the recognition of primitives and higher level reasoning. The systems resemble to system configuration for understanding natural language where the features are described according to a kind of syntax so that we read and understand the meaning of target [19][20][23][24]. The syntactical recognition provides us better results than conventional numerical approaches such as the Fourie Transform, the Radial Basis Functions and neural networks.

The GP procedure has been successfully applied to the estimation of chaotic dynamics using the observed time series, and a direct control method for chaotic dynamics is proposed based on the GP [1]-[4][9]-[13][22][25]. Moreover, the GP method has been widely used to emulate the agents' behavior in various markets such as the stock market.

In this paper, we extend the GP method to estimate and approximate non-linear functional forms to fit the segments of time series, and its application to categorization of time series. The GP procedure is also used to generate rules for predicting future stock trends. The GP method is effective in categorization phase as well as in learning phase. These non-linear functional forms are represented as tree structure (called individuals), and one tree corresponds to a model to generate time series. We have many individuals (pool) for the categorization whether a time series belongs to a certain category. The individuals are improved by using the GP procedure in learning phase so that the estimation becomes to be better. The scheme to maintain the pool of individuals is necessary for the GP procedure, but it also contributes to absorb the variation of the time series in clusters. Then, the variation of the individuals with relatively high capability in the pool can cope with categorization for various kinds of time series which are seemed to belong to the same category. Since the time series of stock price usually consists of concatenation of typical time series segments, we use the sliding window method to find the beginning and ending of the segments by finding category of segments. In the method, we use overlapping the time window along the time so that the nonlinear modeling using the GP and the detection of the time interval are simultaneously realized. By introducing logical variables for depicting the observed segment in the past times series, we can apply the same GP procedure to generate prediction rules. In the GP scheme, we replace the arithmetic variables wit logical variables, and arithmetic operators with logical operators in original GP procedure. As an application, we use the system to predicting future stock trends typically found in the technical analysis. Simulation studies show about 80%～60% true recognition for the two-way prediction, and about 60%～79% true recognition for three-way prediction.

In the following, in Section 2, we describe the overview of hierarchical configuration of the system. In Section 3, we explain the basics of the GP procedure. Section 4 describes the application to categorization of time series segments, and Section 5 shows the silumation result of prediction of future stock trends.

2 Overview of Hierarchical System Configuration

2.1 Hierarchical systems

At the beginning of the paper, we explain the overview of the hierarchical system configuration for the prediction of stock trends based on the GP as shown in Fig.1. We assume that the time series of stock price is composed of concatenation of segments of time series which are categorized into 8 kinds of typical segments whose length is about 35 (observations on 35 trading days). These 8 kinds of segments are referred as categories of segments, and the procedure to find one of these categories is called as categorization.

It is also assumed that segment data for 8 categories is prepared for learning in the first stage system. Then, we obtain 8 pools of individuals for 8 categories where each individual corresponds to time series model to approximate the generation of time series. Since these 8 pools provide us the way to detect and identify the category of segment, we can know the series of symbols representing these categories (called as the category symbol in the following) for input stock price. However, generally the length of time segment and the boundaries of these segment are not given beforehand in the stock price, then we apply the sliding window to detect the boundaries and to identify the kind of segment, simultaneously.

Ultimately, we obtain series of category symbol for a time series for stock price. Then, the GP procedure is applied to generate rules for predicting future stock trends. We assume that learning data for GP procedure is also prepared for rule generation. Namely, we have data for series of category symbol and the data for rise/fall of stock price. Then, we obtain the GP pools for logical expression (rules) which suggest us the rise/fall of the stock trends. The GP procedure for generating rules is the same as for the first stage system, where we replace arithmetic variables by logical variables and arithmetic operators by logical operators, respectively.

2.2 Overview of first stage system

In the following, we show the overview of the first stage system for categorization method for segments based on the GP.

(1)Learning data

It is assumed that the time series data is stored and available in the system, each of which is divided into the same length. Moreover, it is assumed that the time series data used for learning process is available, each of which is accom-

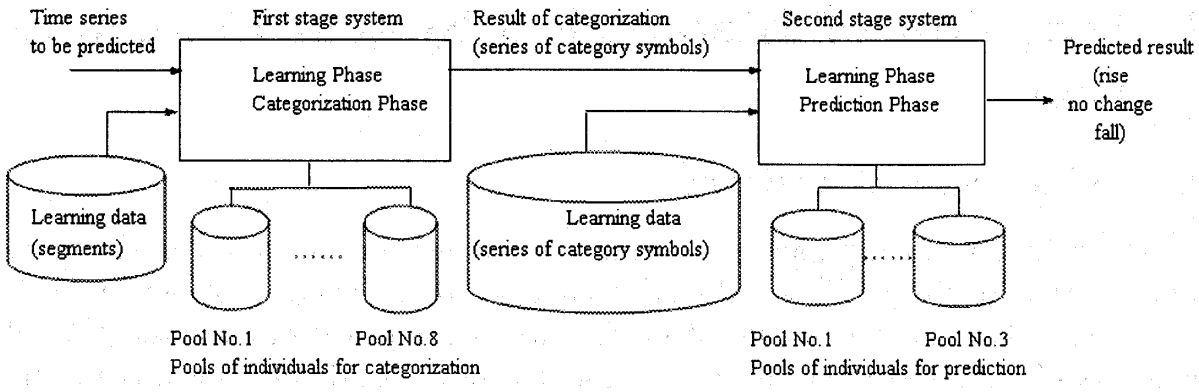

Figure 1: Overview of systems

panied with the category to which the underlying time series is expected to be classified. In Figure 2 depicting the learning phase, it is assumed that such kinds of learning data are prepared for each category $i(i = 1 \sim 8)$.

(2)Learning based on the GP

At first, the approximations of functional forms for each category are obtained to describe the generating models of time series. These functional forms correspond to the individuals in pool in the GP procedure. The GP method is used for the approximation of functional forms. The functional form (individual) is not a single form, but is composed of a set of forms to approximate various generating models, while learning data includes variation of time series obtained from a single basic form by expanding or shrinking the time scale. Then, in the pool used for the GP method a number of individuals having relatively higher ability (called fitness) of approximation are retained in the system, and are used for clustering.

Since we use a set of learning data repeatedly to improve the fitness of individuals, the maximum value of the fitness of individuals in the pool does not increase monotonically, which is different form ordinary optimization processes or approximation using the GP.

Therefore, the iteration of the GP procedure is carried out until sufficient variation of functional form (individuals) for approximating the feature of time series is obtained.

As is shown in Figure 2, in the Learning Phase the independent pools of individuals are organized for each category $i(i = 1 \sim 8)$.

(3)Calculation of fitness of individuals

After applying the Learning Phase, we calculate the fitness of individuals in the Categorization Phase in Figure 2 for every individual stored in each pool i(i=1

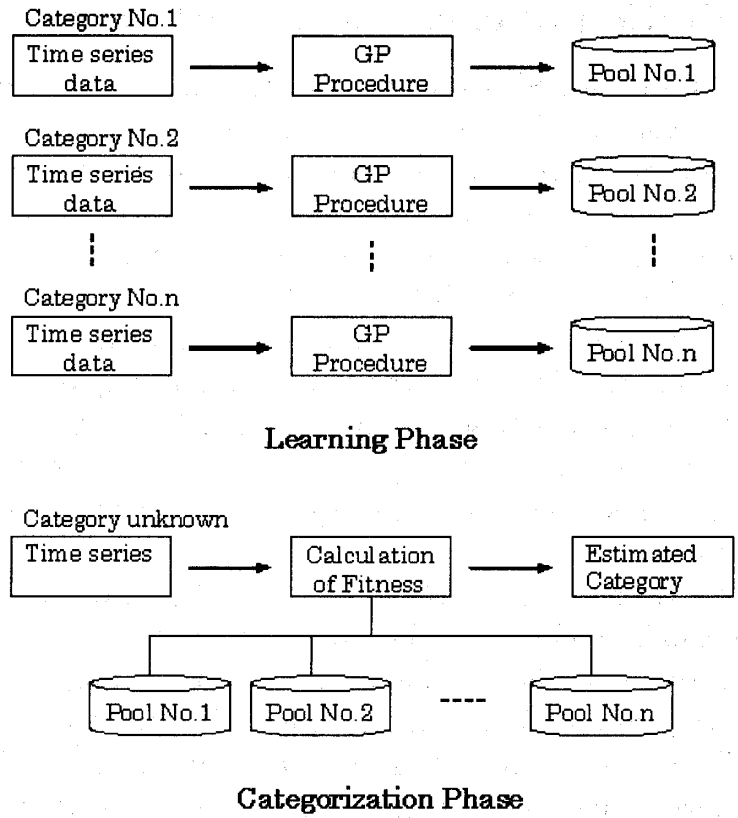

Figure 2: Overview of first stage system

~n) by adopting (fitting) the observed data $x(t)$ of underlying time series whose category is not known and is needed to be determined by the system. Since the individual represented in the functional form is regarded as the prediction of time series at time t using data up to time $t-1$, we can calculate the prediction of $x(t)$ denoted as $\hat{x}(t)$. If the root mean square error (called $rmse$) between $x(t)$ and $\hat{x}(t)$ is small, then the fitness (ability) of a certain individual i realizing the underlying functional form is relatively high. Then, the fitness f_i of individual i is defined by the inverse of $rmse$.

In the Categorization Phase, we calculate the fitness f_i for every individual in every pool by fitting the observation $x(t)$ of time series with known category. Then, we estimate (determine) the category K of the time series by selecting the highest f_{max} among f_i where the individual i belongs to the K th pool.

2.3 Categorization of continuous stock prices)

In previous sections, we showed segments of stock prices are categorized by using the GP pools. However, in a long record of time series, it may happen the cases where the time point of the beginning and ending of the segment are not known beforehand. Then, we examine the capability of categorization scheme of the paper even in the cases where we must also find the time interval in which the segments of the stock prices are included.

Similar to the ordinary method for the recognition of time interval in a time series, we use the sliding window method to find the beginning and ending of the segments. The method is based on overlapping the time window along the time so that the nonlinear modeling using the GP and the detection of the time interval are simultaneously realized. The algorithm is summarized as follows.

(1)Sliding windows

Given the whole time series as $x(t), t = t_1, t_2, ..., t_M$, and the interval T where the segments are usually included in the time interval T. We call T as the length of window. Then, we define the fraction of window, for example, $I = T/5$. Then, we move the starting point T_s and the ending point T_e of the window such as $T_s = 1 + k \times I, T_e = T_s + T$ where k is integer as shown in Fig.3. Then, the all of the set of windows (called as sliding windows) cover the whole time series by changing the starting point and ending point incrementally.

By using the scheme of sliding window,, we can find the true interval in the time domain in which the segments of the time series are included by observing the maximum fitness of categorization system

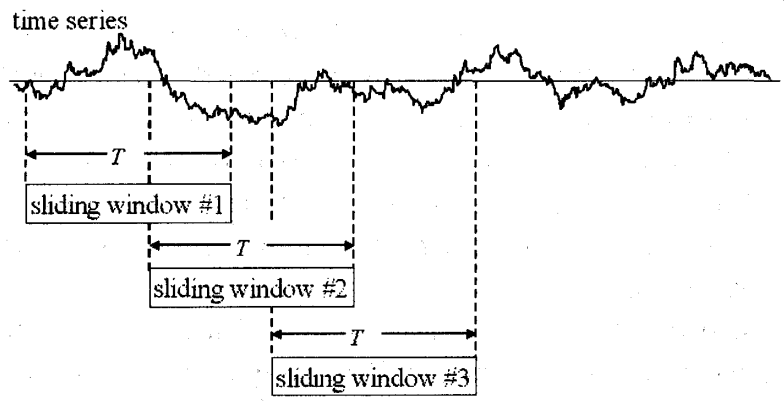

Figure 3: Overview of sliding windows

(2)Record of estimated category and maximum fitness

Then, we apply categorization method to the portion of time series which are

taken out from the whole time series by using the sliding windows. According to categorization, we have a certain estimated category to which the portion of time series to be classified, and the maximum value of fitness obtained by the system. Then, we take a record of these two values along the time (denoted as K_i for the cluster and $F_i, i = 1, 2, .., 5 \times M$ for the fitness, respectively. It is expected that the value F_i is large enough, the window in the time period captures the true segment of time series, otherwise we estimate that the system fails to recognize true segments.

(3)Miscellaneous segments

Now, we assume that in the whole time series of stock price $x(t)$, we fined miscellaneous segments different from the basic segments. These miscellaneous segments are assumed to be the segments found in ordinary stock price but having no dominant feature of characteristics observed in real stock prices.

However, we must note that these miscellaneous segments are artificially generated, but quite different from typical segments. Otherwise, the clustering algorithm does not work efficiently.

(4)Finding segment and threshold value of F_i

As we explained, by using the categorization system we obtain the records of K_i and F_i by using the sliding windows and the pool of individuals assigned to each category. But, simultaneously, as we described, we mix the miscellaneous segments different from the 8 segments. Therefore, for the time period when the sliding window covers the miscellaneous segments, the maximum fitness F_i will be limited to be a small value. Moreover, if the sliding window is placed on the boundary between two segments, the maximum fitness F_i will be also small.

For these reasons, we define a threshold value F_s of F_i. We assume if the value of F_i is less than F_s, the sliding window is placed on the boundary between two segments, or the system recognize the miscellaneous segments different from the targets.

2.4 Second stage recognition systems using GP

In the following, we describe the second stage recognition systems using GP. As we showed in previous sections, we obtain series of category symbol for time series segments. Then, we generate rules to predict rise/fall of stock trends.

It may be possible to represent rules in logical expressions, but we utilize relatively simple forms for rules. We assume that we have category symbols in times going back to the past such as $t_1, t_2, ..., t_m$, and the variables $v_1, v_2, ..., v_m$ are assumed to be assigned one of category symbols observed at time $t_1, t_2, ..., t_m$. Then, we define logical variables using $v_1, v_2, ..., v_m$ to express rules to predict rise/fall of price.

(1) Learning data

We assume that the time series of stock price is represented by series of category symbols. Moreover, The data denoting rise/fall of price at the boundaries of segments is also stored and available in the system. The change of price is defined in cases, namely, including only rise/fall (denoted as two-way prediction), and rise/fall and no change (called as three-way prediction.

(2)Learning based on the GP

At first, we predict future price change by using logical expressions (rules) correspond to individuals. The logical forms (individuals) are composed of a set of forms to approximate various prediction rules to cope with variations of time series. If the prediction of a certain individual is the same as observation, then the fitness of the individual is increased. Then the genetic operations are applied to improve the prediction of rules. Then, in the pool after sufficient GP operations remain in the system a number of individuals having relatively higher ability (called fitness) of prediction. For example, in the two-way prediction, we retain N_{high} individuals having higher fitness in pool for rise prediction and in pool for fall prediction, respectively. Different from first stage system, we use only these individuals for prediction.

(3)Prediction of future change

In the Prediction Phase, we calculate the prediction of N_{high} individuals in every pool by fitting the observation of series of category symbols obtained by applying first stage system for a certain time series. We estimate future price by using the relative number of rules which are expressing true value as results. For example, in two-way prediction, if the number of individuals expressing true value in pool for rise prediction is larger than that of pool for fall prediction, then we estimate (determine) rise of stock price.

3 Applying the GP procedures

3.1 GP procedure for approximation of equations

We consider following equation which generate the time series denoted as $x(t)$.

$$x(t+1) = F(x(t-1), x(t-2), ..., x(t-n)) \tag{1}$$

where $x(t-1), x(t-2), ..., x(t-n)$ are used to predict the value of $x(t)$. Suppose that some of $x(t)$ is actually observed, and then we estimate the system equations $F(x(t-1), x(t-2), ..., x(t-n))$ by using the GP.

The GP is an extension of the conventional GA in which each individual in the population (pool of individuals) is a computer program composed of the arithmetic operations, standard mathematical expressions and variables [10]-[15].

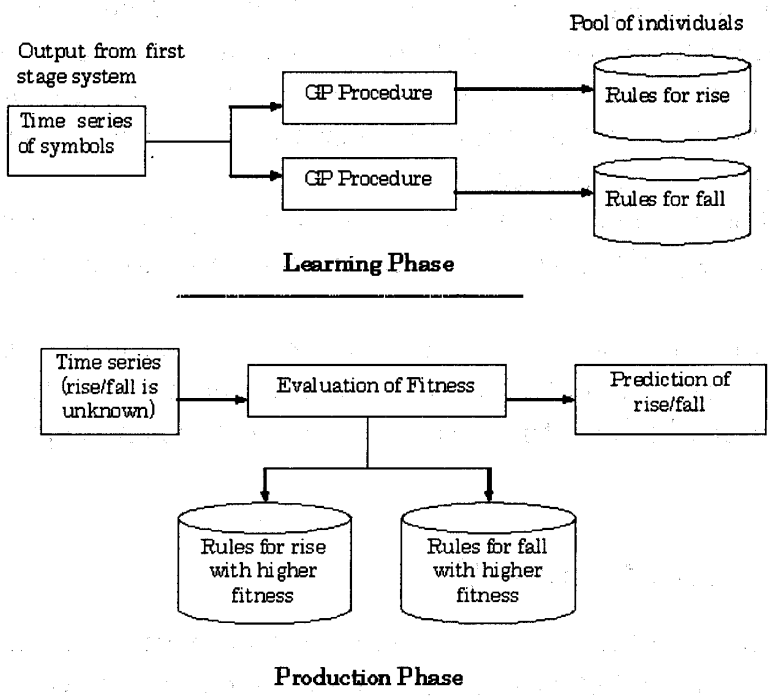

Figure 4: Overview of second stage system

In the GP, the system equations are represented in the tree structure (called individuals). The prefix representation is approved in [15] based upon the comparative study with other representation such as pointer based implementation and the postfix approach. In the parse tree the node non-terminal are taken from some well-defined functions such as binomial operation $+, -, \times, /$, and the operation taking the square root of variable. Terminal nodes consist of arguments chosen from set of constant and variable. The pool of variable consists of the variable $x(t)$ itself and the time lag of $x(t)$ such as $x(t-1)$.

The prefix representation follows traditional representation by using the Lisp syntax. For example, we have the next prefix representation.

$$x(t) = (3{\times}x(t{-}1){-}x(t{-}2)){\times}(x(t{-}3){-}4) \rightarrow {\times}{-}{\times}3x(t{-}1)x(t{-}2){-}x(t{-}3)4 \quad (2)$$

The equation represented by using the prefix are interpreted based upon the stack operation. We begin to scan the prefix representation, and if we meet the terminal (operand) then we push down the term into the stack by storing the result into the stack again.

For checking the validity of underlying parse tree, the so-called stack count (denoted as *StackCount* in the paper) is useful [2]-[4][9]-[15] [17][25]. The *StackCount* is the number of arguments it places on minus the number of arguments

it takes off from the stack. The cumulative *StackCount* never becomes positive until we reach the end at which point the overall sum still needs to be 1.

By using the *StackCount* we can see which loci on the prefix expression is available to cut off the tree for the crossover operation, and we can validate whether the mutation operation is allowed.

If final count is 1, then the prefix representation (tree) corresponds properly to a system equation. Otherwise, the tree structure is not relevant to represent the equation.

Usually, we calculate the root mean square error ($rmse$) between $x(t)$ and $\tilde{x}(t)$ where $\tilde{x}(t)$ is the prediction of $x(t)$, and use it as the fitness. By selecting a pair of individuals having higher fitness, the crossover operation is applied to generate new individuals.

3.2 Algorithm of the GP

By using the measure of fitness to evaluate each individual, we apply the GP to the population to derive better description for the dynamics which generates the targeted time series.

Crossover operations

Contrary to the operation in GA, the crossover operation in GP is applied to restricted cases. Then, we can not choose arbitrary loci in the string of individuals and replace the parts of two tree structures. The two subtrees are extracted and swapped each other.

To keep the crossover operation always producing syntactically and semantically valid programs, we look for the nodes which can be a subtree in the crossover operation and check for no violation. By using the *StackCount* already mentioned, we know the subtrees which are the candidate for the crossover operation. The basic rule is that any two loci on the two parents genomes can serve as crossover points as long as the ongoing *StackCount* just before those points is the same. The crossover operation creates new offsprings by exchanging sub-trees between two parents.

Mutation

The goal of the mutation operation is the reintroduction of some diversity in an population. Two types of mutation operation in GP is utilized to replace a part of the tree by another element.

(Global mutation :G-mutation)

Generate a individual I_s, and select a subtree which satisfies the consistency of prefix representation. Then, select at random a terminal in the individual, and replace the terminal by the subtree of the individual I_s.

(Local mutation:L-mutation)

Select at random a locus in a parse tree to which the mutation is applied, we replace the place by another value (a primitive function or a variable).

We iteratively perform the following steps until the termination criterion has been satisfied.

(Step 1)

Generate an initial population of random composition of possible functions and terminals for the problem at hand. The random tree must be syntactically correct program.

(Step 2)

Execute each individual (evaluation of system equation) in population by applying the optimization of the constants included in the individual. Then, assign it a fitness value giving partial credit for getting close to the correct output. Then, sort the individuals according to the fitness S_i.

(Step 3)

Select a pair of individuals chosen with a probability p_i based on the fitness. The probability p_i is defined for ith individual as follows.

$$p_i = (S_i - S_{min}) / \sum^{N} (S_i - S_{min}) \tag{3}$$

where S_{min} is the minimum value of S_i, and N is the population size.

(Step 4)

Then, create new individuals (offsprings) from the selected pair by genetically recombining randomly chosen parts of two existing individuals using the crossover operation applied at a randomly chosen crossover point. The algorithm is the same as the roulette strategy. If the individual having highest fitness is not included, then we apply the strategy of elite preservation. Iterate the procedure several times to replace individuals with lower fitness.

(Step 5)

If the result designation is obtained by the GP (the maximum value of the fitness become larger than the prescribed value), then terminate the algorithm, otherwise go to Step 2.

3.3 GP procedure for logical expressions

In previous works, we applied the GP method to improve production rules in credit assessment and modeling of artificial agents. To simplify the syytems, we assume that logical expressions are restricted to forms of binary trees consisting two predicates combined with single logical operator, but the restriction brings no serious problems.

Then, the genetic operations can be applied in a similar manner as in arithmetic expressions by giving following replacement of operators and operands.

arithmetic variables vi → logical variables Xi
arithmetic operators +, × → logical operators And,Or

In the second stage systems of the paper, we use the GP procedure to improve prediction rules for stock prices by utilizing inputs of series of category symbols obtained by the first stage systems which categorize the segments of time series using the GP methods. We use relatively simple forms of production rules to predict future stock prices.

We assume that we have category symbols in times going back to the past such as $t_1, t_2, ..., t_m$, and the variables $v_1, v_2, ..., v_m$ are assumed to be assigned one of category symbols observed at time $t_1, t_2, ..., t_m$. For example, symbols c and e are observed at time t_1 and t_2, then we have

$$v_1 = c, v_2 = e$$

By using the expressions as predicates in the logical formula, we can describe the production rules to predict future stock prices. We give following example.

X_{ij}=True if $v_i = s_j$ otherwise False

Each individual in GP pools is represented as a rule (logical expression) including these logical variables to predict rise/fall of stock price. The fitness of individuals of logical expressions is calculated as follows.

(1) Calculate values of logical variables

By checking the symbol observed at time t_i, determine the values of logical variables X_{ij}.

(2)interpret logical expressions

In the scheme, the predicates are given as logical variables, then interpret the logical values of logical expressions.

(3)Calculation of fitness

By using data for learning consisting of series of category symbols and rise/fall observation, we examine the predicted value (rise/fall) of individuals with observations. If these two values are the same, then we increase the fitness of individuals, otherwise do nothing.

4 Application to prediction of stock prices

4.1 Categorization of segments

To test the capability of categorization method of the paper, we apply first stage system to the segments of time series. In the technical analysis of stock price, the parts of the stock price (called as segments) are characterized with their features as almost standardized forms [21]. Figure 5 shows these basic 8 patterns of the

segments as rough sketches. It is known that traders forecast the rise/fall of underlying stock by checking and recognizing the appearance of these 8 patterns of segments The notations and characteristics of these 8 segments are summarized as follows. (a)downtrends: descending parallel trend channel
(b)uptrends: ascending parallel trend channel
(c)double top: also called as "M" formation
(d)broadning a rise and fall in expanding triangle
(e)breakout: there is a feature wave form in the former steps, and the stock price keeping monotonously rising
(f)rectangular: a rise and fall in two balanced lines
(g)rounding: a big curve that is in upward (or downward) curves, and the peak (bottom) value of stock prices appear at both ends
(h)triangle: a rise and fall in symmetrical triangle.

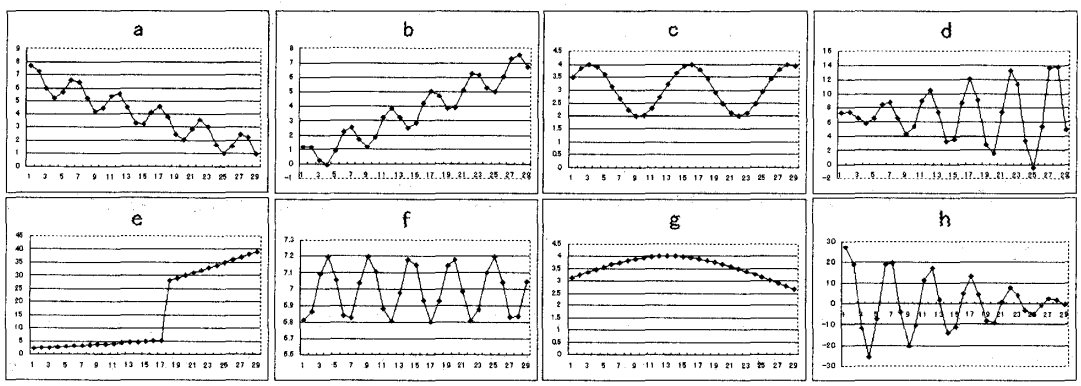

Figure 5: Overview of 8 patterns of segments in stock prices

Since the process of categorization of the time series segments is described in our preceding works, we concisely summarize only the result of simulation for clustering of segments. The condition for the simulation study is as follows:
Number of time series for each cluster:30
Maximum length of individual:20
Operators included in individuals:, $+, -, \times, abs$
Operands included in individuals : $x(t-1), x(t-2), ..., x(t-10)$
Number of individuals in pool:1000
Maximum generation of GP:600

For simplicity, the data for learning and testing are the same in the simulation study.

Table 1 shows the result of categorization. In Table 1, the numbers in vertical column mean the original categories to which the segments belong, and the numbers in horizontal row mean the estimated categories obtained by the system

which is determined by categorization process. The values in the table denote the rate of categorization. Theoretically, only the diagonal elements take the value of 100 %, but due to the categorization error, a part of segments are misclassified into another clusters.

However, the rate of proper categorization ranges from 70 % through 97%, and the average value of proper categorization is about 85 %. Then we can confirm the ability of categorization system.

Table 1: Result of categorization of segments (p:%)

segment No.	p	segment No.	p
a	83	b	87
c	97	d	77
e	97	f	93
g	70	h	70

4.2 Generation of category symbols by using sliding window

In the following, we summarize the simulation study for clustering the segments of stock price based on the artificially generated time series mentioned above. We select at random one of the 8 segments and concatenate it with a miscellaneous segment. Then, we repeat the process until we obtain a sufficient length of time series to be examined.

The simulation result is also summarized in our previous works, then we show examples recognition of segments using the sliding windows in Table 2. In Table 2, the values mean the average rate of true classification for the segments. We define the false result of categorization as the cases where the category of the segment is not the same as the original segment, and the miscellaneous segments are recognized as targeted segments. Moreover, the false estimation may happen when the target segments are missed to be detected by the system.

As is seen from Table 2, we have about 67%～87% correct categorization by the system proposed in the paper.

Table 2: Result of categorization using sliding windows (p:%)

segment No.	p	segment No.	p
a	79	b	80
c	83	d	73
e	87	f	87
g	67	h	70

5 Prediction of future stock trends

5.1 Two-way prediction

In the following, we apply the hierarchical system for predicting future stock trends of the paper to the time series data of Japanese securities exchange market. It is assumed that the first stage categorization method is already applied to the stock time series, and we have series of category symbols of segments. We have also the price changes on the boundaries of segments.

The conditions for the simulation studies are summarized as follows.
length of time series:2000 samples (2000 trading days)
segments obtained for a time series:200
data for learning and testing:2/3 of segments are used for learning, and rest 1/3 is used for testing

We select data for learning and testing from sets of segments at random. After preparing pools of individuals for prediction using the GP, we apply the rules for prediction to the data for testing.

We assume two cases of simulation studies.
Case 1: T days after prediction

We predict price change after T days from present time where T is the same as the length of sliding window.
Case 2: $2T$ days after prediction

We predict price change after T days from present time

We simply define rise (fall) of stock price if the future price after T or $2T$ days is higher than (lower than) present value.

The simulation results for two-way prediction are summarized in Table 3. In Table 4, we show best six rules for predicting rise/fall of prices in the logical expressions.

As we explained earlier, the first suffix in logical variables means the time point going back to the past, and the second suffix means the categories of segments estimated in the first stage system. For example, the following rule means future

price will rise if in t_1, t_2, t_4 categories c, d, f or categories c, d, h are observed

`And X13 And X24 Or X46 X48`

As is seen from the result, we have 78,79% and 72,75% true prediction for Case 1 and 2 of two-way changes, and the rate of recognition is sufficiently good.

Table 3: Result of prediction of stock price (two-way)

price change	Case 1	Case 2
rise	78	72
fall	79	75

Table 4: Examples of rise/fall prediction rules

rise rules	And X_{21} And And And X_{18} X_{35} X_{53} X_{44}
	And Or X_{17} X_{45} And Or X_{52} X_{35} X_{21}
	Or X_{21} Or X_{14} And X_{38} And X_{54} X_{42}
	And And X_{13} X_{28} And And X_{36} X_{41} X_{62}
	And And X_{22} X_{18} And X_{43} And X_{36} X_{61}
	And X_{33} Or And X_{11} X_{28} And X_{41} X_{57}
fall rules	And X_{44} And X_{24} And X_{37} And X_{11} X_{66}
	Or X_{37} And And X_{28} And X_{41} X_{16} X_{54}
	And X_{16} And Or X_{28} And X_{32} X_{53} X_{45}
	And And Or And X_{83} X_{62} X_{18} X_{71} X_{24}
	Or X_{15} And Or X_{26} X_{42} And X_{66} X_{34}
	And Or X_{43} And Or X_{24} X_{55} X_{32} X_{61}

5.2 Three-way prediction

Then, we apply the prediction method for three-way prediction. We also define two cases for simulation studies, where the range of rise/fall of price must be considered.

Case 1: T days after prediction

We predict price change after T days from present time where T is the same as the length of sliding window. If the range of change T days after is less than 15% of present price, then we regard them as "no change", otherwise define then as rise or fall.

Case 2: $2T$ days after prediction

We predict price change after T days from present time. If the range of change T days after is less than 23% of present price, then we regard them as "no change", otherwise define then as rise or fall.

Table 5 shows the simulation results for three-way. As is seen from the result, we have 75,81,80% and 72,81,75% true prediction for Case 1 and 2 in three-way changes , and the rate of recognition is sufficiently good.

Table 5: Result of prediction of stock price (three way)

price change	case 1	case 2
rise	75	72
no change	81	78
fall	80	75

5.3 Comparison with multi-stage fuzzy systems

To compare the result of prediction obtained by the method of the paper, we show the results of similar simulation studies given by multi-stage fuzzy inference systems that we proposed in previous works. In general, in the fuzzy inference systems, if the number of input becomes large, then the number of rules become very large and untractable. To suppress the number of rules, we introduce multi-stage fuzzy inference systems where the input variables are used distributed manner. Therefore, the prediction of the multi-stage fuzzy inference systems are seemed to provide us relatively good results of prediction. In our previous studies it is shown that the result of prediction is better than that of neural networks under the same condition.

The details of the multi-stage fuzzy inference systems is omitted here due to the limitation of length of paper. Usually, the input variable at time t to predict future value $x(t+1)$ to the multi-stage fuzzy inference are selected to be $x(t-1), x(t-2), ..., x(t-m)$, but in our works, we utilized the wavelet transform of the time series $x(t)$ to increase the rate of prediction.

The wavelet transform of the time series $x(t)$ is defined as follows [16].

$$x(t) = \sum_n \sum_m x_n^m \psi_n^m(t). \tag{4}$$

$$x_n^m = \int_{-\infty}^{\infty} x(t)\psi_n^m(t)dt. \tag{5}$$

$$\psi_n^m(t) = 2^{m/2}\psi(2^m t - n). \tag{6}$$

where integer m and n are the dilation and translation index. For simplicity, we assume that the sampling interval of the time series $x(t)$ is 1. Since the Fouries Transform of each wavelet function ψ_n^m is not overlapped, then the integer m and n are restricted as follows.

$$m = 2^K, n = Jm(J = 1, 2, ..., K = 0, 1, ...) \tag{7}$$

Other than the wavelet coefficients, we also use the fractal dimension of the stock price as an input variable. As is seen from many examples, the stock price can be modeled as a fractal time series. The fractal dimension corresponds to the complexity of the time series.

The spectrum should decrease as the frequency and belong to $1/f$ family. The condition is represented by the variance of the wavelet coefficients.

$$var(x_n^m) = \sigma^2 2^{-\gamma m}, \gamma = 5 - 2D \tag{8}$$

As is seen, the fractal dimension D is included in equation (8). By taking the logarithm of equation (8), we see that $\log(var(x_n^m))$ is approximated as the linear regression of the index m.

$$R_w = [\sum_m (\log(var(x_n^m)) - c_0 - c_1 m)^2]^{1/2}/(M_s X_r) \tag{9}$$

where M_s is the number of index m, X_r is the range of x_n^m , and c_0, c_1 are regression coefficients. Then, by taking the root mean square error between the $log(var(x_n^m))$ and the regression line, we can define the measure to test whether the time series is approximated as the fractal. In the multi-stage fuzzy inference system, the wavelet coefficients obtained from the stock trends are used as the input variables. The number of an input variable can be made larger than that of a single stage inference system.

However, in a real application, the number of available coefficients is restricted. As is shown in equation (5), the maximum number of the dilation index m is defined by using the length L of the time series $x(t)$. Namely, the number of wavelet coefficient for the given dilation index m is determined by

$$N_m = L/2^m \tag{9}$$

If we choose N_m by considering the feature extraction in the time domain, then the maximum number of m is determined that corresponds to the resolution in the frequency domain.

Since we usually find structural changes of stock price in several years, the maximum value of L is limited to about to 1000. Then, we determine that the maximum number of m is 6. As is already mentioned, the fractal dimension and

the variance of the stock trend are also used as the input variable.

Then, the input variable to each stage of the fuzzy inference system is given as follows.

Number of membership function:fivr
time series $x(t)$: normalizae into $x(t) \leq 150$
number of stages: three
1-st stage:fractal dimension, variance, x_n^0, x_n^1
2-nd stage:x_n^2, x_n^3, x_n^4
3-nd stage:x_n^5, x_n^6

In the simulation study, the inference system is evaluated for the following two cases.

(1)two-way prediction
(2)three-way prediction

In case of rise of stock price, if the inference system estimate that $S(t+T_p) - x(t) > U$ is held, then the system predict a rise of stock price where $x(t)$ and $S(t+T_p)$ are present price and predicted price at $t+T_p$, respectively. On the other hand if $S(t+T_p) - x(t) < D$ is estimated, then the system predict a fall of stock price. In three-way prediction , the estimation rule for rise/fall are the same as for two-way prediction. For the estimation of no-change, we use rule $S(t+T_p) - x(t) < U$ and $x(t) - S(t+T_p) > D$. In the simulation studies, data for learning are used from 2/3 of total segments, and rest 1/3 are used for testing

Table 6: Result of two-way prediction (%)

T_p	U=20,D=10	U=15,D=15	U=20,D=20
20	67	67	68
30	65	69	71
50	67	71	70

Table 7: Result of Three-way prediction (%)

T_p	U=20,D=10	U=15,D=15	U=20,D=20
20	62	67	66
30	63	64	70
50	65	67	68

Table 6 and 7 show the ratio of correct prediction of stock price for two-way prediction and three-way prediction. As is seen from the result, the stock price is predicted in average at the probability 68%. The result of prediction is slightly

worse than that of the two-way prediction. But, the rate of correct prediction is over 60 % in spite of the way of prediction is three.

We find no significant difference of prediction depending on the combination among U, D and T_p. The fact means if we choose $20 \leq T_p \leq 50, 15 \leq U \leq 20$, and $10 \leq D \leq 20$, we can predict future price change at the probability of about 70%.

Then,we can conclude that the prediction obtained by the hierarchical recognition systems based on the GP treated in the paper is better than the predictions obtained by the multi-stage fuzzy inference systems.

6 Conclusion

This paper showed the prediction of future stock trends by using two-stage hierarchical systems based on the segment categorization and recognition of series of category symbols using the GP. The GP procedure was applied to realize categorization of segments and rule generation for prediction. The system was applied to the recognition of future stock trends using the series of category symbols as input data, and we obtained sufficient result of prediction for two-way and three-way prediction.

It is still remained to extend the method of the paper to be applicable to real trading systems, and further works will be done by the authors.

References

[1] Chen.S.H. and C.H.Yeh.C.H.,(2001),Evolving traders and the business school with genetic programming : A new architecture of the agent-based artificial stock market, Journal of Economic Dynamic and Control, vol.25,pp.363-393.

[2] Chen.X. and Tokinaga.S. (2002),Approximation of chaotic dynamics for input pricing at service facilities based on the GP and the control of chaos , Trans. IEICE, vol.E85-A,no.9,pp.2107-2117.

[3] Chen.X. and S.Tokinaga.S.(2003), Synthesis of multi-agent systems based on the co-evolutionary genetic Programming and its applications to the analysis of artificial markets, Trans. IEICE,vol.E86-A,no.10,pp.1038-1048.

[4] Chen.X. and S.Tokinaga.S. (2004),Multi-agent modeling of artificial stock markets by using the co-evolutionary GP approach, Journal of Operations Research Society of Japan, vol.47, no.3, pp.163-181.

[5] J.J.Fernandez.J.J, Farry.K.A, and Cheatham.J.B (1996), Waveform recognition using Genetic Programming: The myoelectric signal recognition problem in (eds J.Koza) Genetic Programming.

[6] Freitas.A.A (2002), Data Mining and Knowledge Discovery with Evolutionary Algorithms, Springer-Verlag, 2002.

[7] K.S.Fu.K.S (1974),Syntactical Method in Pattern Recognition,Academic Press.

[8] Ikeda.Y. (2002),Estimation of the chaotic ordinary differential equations by co-evolutionary genetic programming, Trans. IEICE, vol.E85-A,no.4,pp.424-433.

[9] Ikeda.Y and Tokinaga.S (2000),Approximation of chaotic dynamics by using smaller number of data based upon the genetic programming, Trans. IEICE, vol.E83-A,no.8, pp.1599-1607.

[10] Ikeda.Y and Tokinaga.S (2001),Controlling the chaotic dynamics by using approximated system equations obtained by the genetic programming, Trans. IEICE,vol.E84-A, no.9, pp.2118-2127,2001.

[11] Ikeda.Y. and Tokinaga.S.(2003),Synthesis of multi-agent systems based on the co-evolutionary genetic Programming by considering social learning and its applications to the analysis of artificial stock markets, Journal of Japan Society for Management Information, vol.12,no.3,pp.17-35.

[12] Ikeda.Y. and Tokinaga.S.(2004), Chaoticity and fractality analysis of an artificial stock market by the multi-agent systems based on the co-evolutionary Genetic Programming, Trans. IEICE,vol.E87-A, no.9,pp.2387-2394.

[13] Y.Kishikawa and S.Tokinaga, Pridiction of stock trends by using two-stage hierarchical systems based on the categorization of segments and recognition of series of category using the Genetic Programming and its applications," Proc. of 2005 Fall National Conference of ORSJ, 2005.

[14] Kishikawa.Y. and TokinagaS. (2000),Prediction of stock trend by using the wavelet transform and the multi-stage fuzzy inference system optimized by the GA,IEICE Trans.Fundamentals,vol.E83-A,no.2, pp.357-366.

[15] Kishikawa.Y. and Tokinaga.S.(2001),Approximation of multi-dimensional chaotic dynamics by using multi-stage fuzzy inference systems and the GA IEICE Trans.Fundamentals,vol.E84-A,no.9,pp.1204-1215.

[16] Koza.J.R (1990),Genetic programming:A paradigm for genetically breeding populations of computer programs to solve problems, Report No.STAN-CS-90-1314, Dept.of.Computer Science Stanford University.

[17] Koza.J.R (1991),Evalution and subsumption using genetic programming, Proc of the First European Conference on Artificial Life, MIT Press.

[18] Koza.J.R (1992),Genetic Programming,MIT Press.

[19] Keith.M.J and Martin.M.C (1994),Genetic programming in C++: Implementation issues, in (ed) K.E.Kinnerar,Jr.,Advance in Genetic Programming MIT Press.

[20] Lu.J and Tokinaga.S (2004), An aggregated approximation for modeling of time series based on the Genetic Programming and its application to clustering, Special Interest Group of Operations Research Society of Japan "Modeling under Uncertainty".

[21] Lu.J and Tokinaga.S (2004), Nonlinear modeling of time series based on the Genetic Programming and its applications to clustering of feature in stock prices," submited to JORSJ,2004.

[22] Tan.K and Tokinaga.S (1999), Optimization of fuzzy inference rules by using the genetic algorithm and its application to the bond rating , JORSJ,vol.42,no.3,pp302-315.

[23] Tan.K and Tokinaga.S(1999),The design of multi-stage fuzzy inference systems with smaller number of rules based upon the optimization of rules by using the GA, IEICE Trans.Fundamentals,E82-A,9,pp.1865-1873.

[24] Tokinaga.S (1984),On the management of time-series data by feature description based upon generative grammar- By fitting ARMA model to EEG signals, IEICE.Trans.Fundamentals, vol.J67-D,vol.10,pp.1099-1106.

[25] Tokinaga.S (1986),Automatic EEG classification besed on syntactical pattern recognition method-feature extraction by adaptive ARMA model fitting,IEICE.Trnas.Fundamentals , vol.E69,no.10,pp.1125-1132.

[26] Tokinaga.S and Ishida.Y (1995),An intelligent digital signal processing systems for stock trends based upon transient wave detection by using Gabor representation and knowledge representation of waveform, Trans.IEICE,vol.J78-A, vol.2,pp.169-177.

[27] Tokinaga.S and Tominaga.M,(2003), An improvement method for the workflow management systems based on the reallocation of flows using the Genetic Programming and its applications, Journal of Operations Research Society of Japan, vol.46, no.3, pp.286-305.

[28] Trahanias.P and E.Skordalakis.E (1991),Syntactic pattern recognition of the ECG",IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.12,no.7,pp.648-657.

[29] Tsai.W.H and Fu.K.S (1980),Attributed grammar- a tool for combining syntactical and statistical approaches to pattern resognition", IEEE Transaction on Systems, Man and Cybernetics, vol.SMC-10,no.12,pp.873-885.

[30] Yababe.M and Tokinaga.S,(2002),Applying the genetic Programming to modeling of diffusion processes by using the CNN and its applications to the synchronization,IEICE.Trans. Fundamentals, vol.J85-A,no.5,pp.548-559.

Shozo Tokinaga
(Professor of the Graduate School of Economics,
Kyushu University)

Yoshinori Kishikawa
(Assistant of the Faculty of System Science and Technology,
Akita Prefectural University)