A study of the impact of COVID-19 on the Chinese stock market based on a new textual multiple ARMA model

Weijun Xu¹ | Zhineng Fu¹ | Hongyi Li² | Jinglong Huang¹ | Weidong Xu³ | Yiyang Luo⁴

¹School of Business Administration, South China University of Technology, Guangzhou, China
²Business School, The Chinese University of Hong Kong, Shatin, Hong Kong
³School of Management, Zhejiang University, Hangzhou, China
⁴School of Economics and Management, South China Normal University, Guangzhou, China

Correspondence
Yiyang Luo, School of Economics and Management, South China Normal University, Guangzhou, 510631, China.
Email: yiyang.luo@hotmail.com

Funding information
Guangdong Basic and Applied Basic Research Foundation, Grant/Award Numbers: 2019A1515011752, 2021A1515110876; National Natural Science Foundation of China, Grant/Award Numbers: 71720107002, 71771091; Philosophy and Social Science Foundation of Guangdong Province, Grant/Award Number: GD19YYJ11

Abstract
Coronavirus 2019 (COVID-19) has caused violent fluctuation in stock markets, and led to heated discussion in stock forums. The rise and fall of any specific stock is influenced by many other stocks and emotions expressed in forum discussions. Considering the transmission effect of emotions, we propose a new Textual Multiple Auto Regressive Moving Average (TM-ARMA) model to study the impact of COVID-19 on the Chinese stock market. The TM-ARMA model contains a new cross-textual term and a new cross-auto regressive (AR) term that measure the cross impacts of textual emotions and price fluctuations, respectively, and the adjacent matrix which measures the relationships among stocks is updated dynamically. We compute the textual sentiment scores by an emotion dictionary-based method, and estimate the parameter matrices by a maximum likelihood method. Our dataset includes the textual posts from the Eastmoney Stock Forum and the price data for the constituent stocks of the FTSE China A50 Index. We conduct a sliding-window online forecast approach to simulate the real-trading situations. The results show that TM-ARMA performs very well even after the attack of COVID-19.

KEYWORDS
COVID-19, FTSE China A50 Index, multiple ARMA, stock forum, textual sentiment

1 | INTRODUCTION

In the end of 2019, coronavirus 2019 (COVID-19) suddenly attacked human race, and caused serious damage to the economy of China and even the whole world. The financial market fluctuated violently in the following months and caused huge losses to investors. Four circuit breakers were triggered in the U.S. stock market and the Shanghai Composite Index fell sharply and violently fluctuated between 3127.17 and 2646.8. The spread of panic was the main reason for the stock market crash. As we know, the rise and fall of any specific stock is also influenced by the fluctuations of many other stocks. Web textual information has great influence on the emotions of investors [3]. The discussions in stock forums affect the emotions of investors who participate in the discussions and browsing the posts, and they spread the sentiment to other investors. Investors and stocks in the stock market form a complex system, which leads to herd behavior.

There are some researches that study the impact of COVID-19 on the stock markets. The research of Al-Awadhi et al. [2] based on a panel data found that the
number of confirmed cases and death of COVID-19 have significant negative impact on Chinese stocks. Zhang et al. [25] showed that COVID-19 caused the increase of stock markets’ risk all over the world. Akhtaruzzaman et al. [1] showed that the conditional correlations between the stock returns of G7 countries and China increased significantly, which is the result of the transmission of financial contagion. However, these literatures rarely researched the textual sentiment and the propagation effect among the investors and stocks.

The Chinese stock market was one of the earliest stock markets hit by COVID-19, so it is meaningful to research the impact of COVID-19 on the Chinese stock market. To better measure the impact of investors’ emotions, we take textual information into account. Our textual information is collected from the Eastmoney Stock Forum (https://guba.eastmoney.com), a leading stock forum in China. There is a specific stock forum for every stock listed in Chinese stock market, but some stock forums are inactive, with relatively few textual information. Fortunately, the stock forums for most of the large capital stocks are active, because many investors pay attention to them. Considering the average number of posts per stock forum, we choose the constituent stocks of the FTSE China A50 Index as the research object. Therefore, our dataset are the posts in Eastmoney Stock Forum and price data for constituent stocks of the FTSE China A50 Index from January 1, 2019 to December 31, 2020. We compute the daily textual sentiment score for every stock by an emotion dictionary-based method.

To verify the change of Chinese stock market before and after the attack of COVID-19, we compare the excess returns of prediction models in the two different stages. Autoregressive Moving Average model (ARMA) is a commonly used model for price prediction. Considering the cross impact among stocks and textual information, we propose a new Textual Multiple ARMA (TM-ARMA) model. TM-ARMA constructs a new multiple cross autoregressive term and a new multiple cross textual term to measure the cross impact on price fluctuations and textual emotions, respectively. The adjacent matrix that measures the relationships of stocks is very important for multiple ARMA models. Different from Vector Auto Regression (VAR), Vector ARMA (VARMA) and Vector ARMA with eXogenous regressors (VARMAX) models [17] which consider the adjacent matrix as an all-one matrix, TM-ARMA updates the adjacent matrix every day by the correlation coefficient matrix for the return series of the last 20 trading days, which reflects the relationships of stocks more accurately. To estimate the parameter matrices, we construct a log-likelihood objective function for prediction errors, and apply matrix-form gradient descent methods based on the solutions for partial derivatives of matrices provided by Petersen and Pedersen [11].

ARMA and its extended models are widely used for financial predictions [6, 16]. Many researchers value the importance of interactions between assets and improved multivariate ARMA models. Thornton [21] proposed a mixed-frequency multiple ARMA by employing mixed stock flow data, and it has a good performance in the simulation. Lennon and Yuan [13] proposed a multivariable ARMA model combining digitized Gaussian and Monte Carlo Expectation Maximization method (EM-ARMA) which was shown to be effective in the simulation.

Textual information from stock forums is used in the research of sentiment in recent years. Yao et al. [23] constructed an investor attention index by the posts from the Eastmoney Stock Forum and Hexun Stock Forum. Yang et al. [22] used the textual information from Eastmoney Stock Forum to build a panic sentiment index, and studied the impact of textual sentiment on crash risk. The emotion dictionaries are widely used for the computation of sentiment scores. Li et al. [14] mapped the words from the news into a sentiment space of four different sentiment dictionaries, and built sentiment scores for stocks in Hong Kong. Groß-Klußmann et al. [9] proposed an unsupervised learning method based on the positive and negative word lists from emotion dictionaries to build sentiment scores of microblog messages. Siering [20] applied a dictionary-based method that computes the textual sentiment by the number of positive and negative words.

Maximum likelihood estimation and gradient descent methods are widely applied to estimate parameters [5, 12, 15]. It is also commonly used in the extended ARMA models. Gong et al. [8] modified the conventional two-stage maximum likelihood estimation method by non-Gaussian QML estimator for the ARMA-GJR-GARCH process. Lennon and Yuan [13] estimated the parameters for their proposed multivariable ARMA model by a log-likelihood function and partial derivatives.

Grid search is a commonly considered way of searching for the possible hyperparameter combinations, especially when the number of combinations to be searched is not very large [7, 18, 19, 24]. The performance of models is fully evaluated by grid search. It is also widely used in some extended ARMA models [10].

To make our empirical study consistent with the real world trading, we propose a sliding-window online approach. Before 9:30, the model forecasts price change rates of all the 50 stocks for the new day, and select stocks that are predicted to rise and put them into a portfolio. At 9:30, the strategy suggests to buy all the stocks in the portfolio with equal funds and hold them until 9:30 of the
next trading day. Transaction fees and stamp duties are taken into account.

The rest of this paper is arranged as follows. Section 2 discusses our TM-ARMA model and its solutions for parameter matrices. Section 3 deploys the empirical study. Section 4 analyses the empirical results. Section 5 is the conclusion.

2 | PROPOSED MODEL

2.1 | TM-ARMA model

We denote $y_t \in R^{Nx1}$ as the daily returns for the $N$ stocks at day $t$, and $\hat{y}_t \in R^{Nx1}$ is the predicted vector of $y_t$, where $1 \leq t \leq T$, and $T$ is the number of trading days.

Considering the transmission effect of trading sentiment in stock market and textual investor sentiment in stock forums, we construct a new cross-AR term and a new cross-textual term, respectively. We propose a new TM-ARMA model as shown in Equation (1), where $\otimes$ is the Hadamard product operator, $p$, $q$, and $r$ are hyperparameters, $\Theta$, $\Psi$, $\Phi \in R^{Nx1}$ are parameter vectors. $A \in R^{NxN}$ is a dynamic adjacency matrix. The larger the $A_{ij}$, the greater the cross impact of stock $i$ on stock $j$. $x_t \in R^{Np}$ is an exogenous vector which reflects the textual sentiment of each stock at day $t$. In the right hand side of Equation (1), the first part $\zeta \in R^{Nx1}$ is the intercept vector. The second part is a multiple autoregressive term of order $p$, which measures the cross trading sentiment among stocks in prices. The third part consists of the moving average terms of order $q$. The fourth part is an exogenous cross-textual term of order $r$, which measures the cross textual sentiment in stock forums.

$$\hat{y}_t = \zeta + \sum_{i=1}^{p} A_{i} y_{t-i} \otimes \Theta_{i} + \sum_{i=1}^{q} e_{t-i} \otimes \Psi_{i} + \sum_{i=1}^{r} A_{2} k x_{t-i} \otimes \Phi_{i}$$

The adjacency matrix of $A_1$ and $A_2$ plays important roles in the cross-AR term and cross-textual term. The elements in $A_1$ and $A_2$ measure the cross impact between two stocks. We denote $Y \in R^{Nxt}$ as the series of daily returns, $X \in R^{Nxt}$ as the series of textual sentiment, and $E \in R^{Nxt}$ as the series of prediction error. $Y_{{a:b}} = (y_{a}, y_{a+1}, \ldots, y_{b}) \in R^{N \times (b-a+1)}$ denotes the daily returns from $a$ to $b$, where $1 \leq a < b \leq T$. Similarly, $X_{{a:b}} \in R^{N \times (b-a+1)}$ and $E_{{a:b}} \in R^{N \times (b-a+1)}$ denote the textual sentiment and prediction error from $a$ to $b$, respectively. It is well known that the strength of the interaction between stocks change dynamically in response to market sentiment in the past few weeks. To better adapt to the dynamic changes, we set $A_1$ as the correlation coefficient matrix of the return series in the past $W$ trading days, and $A_2$ as the correlation coefficient matrix of stocks’ sentiment index series in the past $W$ trading days, as shown in Equation (2), where $Corr$ is the function to calculate the correlation coefficient matrix, and $W$ is the length of the training window. $A_1$ and $A_2$ are updated daily.

$$A_1 = Corr(Y_{t-W:t-1})$$
$$A_2 = Corr(X_{t-W:t-1})$$

To better understand the model, we write Equation (1) as Equation (3), where $1_1 \in R^{nx1}$, $1_2 \in R^{nx1}$, $1_3 \in R^{nx1}$ are unit vectors. In $\Theta \in R^{Nxp}$, $\Psi \in R^{Nqx}$, $\Phi \in R^{Nxr}$ are parameter matrices. $y_t$, $x_t$, $\Theta_t$, $\Psi_t$, $\Phi_t$ and $e_{t-i}$ are the $i$th columns in $Y$, $X$, $\Theta$, $\Psi$, $\Phi$ and $E$, respectively.

$$\hat{y}_t = \zeta + ((A_1 Y_{t-p:t-1} \otimes \Theta) 1_1 + (E_{t-q:t-1} \otimes \Psi) 1_2 + (A_2 X_{t-r:t-1} \otimes \Phi) 1_3$$

We can also easily find out TM-ARMA’s improvement on VARMA and VARMAX by Equation (3). Firstly, the adjacency matrix $A$ is an all-one constant matrix in VARMA and VARMAX, but in TM-ARMA $A$ is dynamically updated every day. In VARMA and VARMAX, the relationships of stocks are constant and the historical data of all stocks are of equal importance when forecasting the return of one specific stock. However, as we know, the short-term relationships of stocks change over time with market sentiment. Secondly, in VARMA and VARMAX, the third term in the right hand side is the cross MA term that is similar to the Cross-AR term in form, which combines the prediction errors of all the $N$ stocks when forecasting. Our experiment shows that the cross MA term causes too big unstable noises and results in instability of prediction values. As a result, to avoid unstable noises, TM-ARMA applies the single MA term that considers only the errors of the stock to be predicted in Equation (3).

2.2 | Textual sentiment

As the posts of stock forums do not have the labels of sentiment polarity, the methods of supervised learning do not work. In this paper, the textual sentiment series $X$ in TM-ARMA is computed based on the emotion dictionaries. We have a list of positive words and a list of negative words, and they do not share the same words, which are described in detail in Section 3.1.1.

We assume that there are postnum's posts in the stock forum of stock $s$ in day $t$ (from 9:30 in day $t$ to 9:30 of the
next trading day), and we denote the $h$th post as post$_h$. Then the sentiment score of stock $s$ in day $t$ is shown in Equation (4), where Senti is the function to compute the sentiment score of a post. In a specific post post$_h$, we assume that there are posnum$_h$ words in the positive list and negnum$_h$ words in the negative list. Following Siering [20], we compute the sentiment score of the post as shown in Equation (5).

$$X_st = \sum_{h=1}^{posnum'} Senti(post_h)$$ (4)

$$Senti(post_h) = \begin{cases} \frac{posnum_h - negnum_h}{posnum_h + negnum_h} & \text{if } posnum_h \geq 0 \\ 0 & \text{otherwise} \end{cases}$$ (5)

### 2.3 Likelihood function

We denote the noises of $N$ stocks as a random vector $\epsilon$, and $\epsilon_t$ is the prediction error in day $t$, namely

$$\epsilon_t = y_t - \hat{y}_t$$ (6)

Following earlier studies, we assume that $\epsilon$ is a vector of white noise and $\epsilon \sim N(0, \Sigma)$, where $\Sigma \in \mathbb{R}^{N \times N}$ is the covariance matrix of $\epsilon$. The probability density function for $\epsilon$ is shown in Equation (7), where $\Omega = (\zeta, \Theta, \Psi, \Phi, \Sigma)$ is the set of parameter matrices, and $\Sigma$ is a symmetric non-singular non-negative definite matrix with $| \Sigma | > 0$.

$$f_\Omega(\epsilon) = \frac{1}{\sqrt{(2\pi)^N | \Sigma |}} e^{-\frac{1}{2} \epsilon^T \Sigma^{-1} \epsilon}$$ (7)

To simulate the trading in real world, we train the model every morning based on the historical data in the past-$W$ days, where $W$ is the length of training window. In a specific day $t$, the joint probability density function for the pre-$W$ days is $\prod_{t=-W}^{t-1} f_\Omega(\epsilon_t)$, and its log form is as shown in Equation (8), where $\ln$ is the natural logarithm function.

$$\ln \left( \prod_{t=-W}^{t-1} f_\Omega(\epsilon_t) \right) = - \sum_{t=-W}^{t-1} \left( \frac{N}{2} \ln(2\pi) + \frac{1}{2} \ln(|\Sigma|) \right)$$

$$+ \frac{1}{2} (y_u - \hat{y}_u)^T \Sigma^{-1} (y_u - \hat{y}_u)$$ (8)

By dropping the constant, we have the log-likelihood function in day $t$ as shown in Equation (9).

$$L(\Omega) = - \sum_{t=-W}^{t-1} \left( \ln(|\Sigma|) + (y_u - \hat{y}_u)^T \Sigma^{-1} (y_u - \hat{y}_u) \right)$$ (9)

### Table 1

| Variable | Definition | Value or size |
|----------|------------|---------------|
| $N$      | Number of stocks | 50            |
| $T$      | Number of trading days | 323          |
| $S$      | Number of days arranged for initialization | 25          |
| $W$      | The length of training window | 20          |
| $p, q, r$ | Hyperparameters | $\in \{1, 2, 3, 4, 5\}$ |
| $\lambda$ | Learning rate | 0.001         |
| $y_t$    | Returns in day $t$ | $N \times 1$ |
| $x_t$    | Textual sentiment in day $t$ | $N \times 1$ |
| $\hat{y}_t$ | Predicted value of $y_t$ | $N \times 1$ |
| $\epsilon_t$ | Prediction error in day $t$ | $N \times 1$ |
| $\zeta$  | Parameter vector for intercept | $N \times 1$ |
| $\Theta$ | Parameter matrix for the cross-AR term | $N \times p$ |
| $\Psi$   | Parameter matrix for MA term | $N \times q$ |
| $\Phi$   | Parameter matrix for the cross-temporal term | $N \times r$ |
| $Y$      | Series of daily returns | $N \times T$ |
| $X$      | Series of textual sentiment | $N \times T$ |
| $E$      | Series of prediction error | $N \times T$ |
| $A$      | Adjacency matrix | $N \times N$ |
| $\Sigma$ | Covariance matrix of $\epsilon$ | $N \times N$ |
| $\Omega$ | The set of parameter matrices | $(\zeta, \Theta, \Psi, \Phi, \Sigma)$ |

To help better reading, we summarize the important symbols in this paper in Table 1.

### 2.4 Estimation for the parameter matrices

We apply a maximum likelihood method to estimate the parameter matrices, and derive the partial derivative of the objective function with respect to each parameter matrix with other parameters being fixed. Considering that the parameters are in matrix form, we apply matrix-form gradient descent methods based on partial derivatives of matrices. The estimation works are done every day $t$ before the prediction work.
2.4.1 Update \( \zeta \) with other parameters being fixed

We derive the partial derivative for \( \zeta \) in day \( t \) as Equation (10),\(^1\) where \( B_t = y_{u_t} - (\hat{y}_u - \zeta) = y_u - (\lambda (AY_{u-p}, u-1) \odot \Theta) 1_1 + (E_{u-q}, u-1 \odot \Psi) 1_2 + (FA_{u-r}, u-1 \odot \Phi) 1_3), \) and \( \frac{\partial K_1}{\partial \zeta} = 0. \) Then we update \( \zeta \) by Equation (11), where \( \lambda \) is the learning rate. In the empirical analysis, Equation (11) iterates a fixed number of times to estimate \( \zeta. \)

\[
\frac{\partial L(\Omega)}{\partial \zeta} = \sum_{u=t-W}^{t-1} \frac{\partial \left( y_u - (\hat{y}_u - \zeta) \right)^T \Sigma^{-1} \left( y_u - (\hat{y}_u - \zeta) \right)}{\partial \zeta}
= \sum_{u=t-W}^{t-1} \left( \Sigma^{-1} (\zeta - B_1) \right)
= \sum_{u=t-W}^{t-1} \left( -2 \Sigma^{-1} (y_u - \hat{y}_u) \right)
= \frac{\partial L(\Omega)}{\partial \zeta} \zeta = \zeta - \lambda \frac{1}{W} \frac{\partial L(\Omega)}{\partial \zeta}
\]

2.4.2 Update \( \Sigma \) with other parameters being fixed

The partial derivative for \( \Sigma \) in day \( t \) is derived as Equation (12).\(^2\) Then we update \( \Sigma \) by Equation (13).

\[
\frac{\partial L(\Omega)}{\partial \Sigma} = \sum_{u=t-W}^{t-1} \frac{\partial \left( \ln |\Sigma| + \epsilon T \Sigma^{-1} \epsilon \right)}{\partial \Sigma}
= \sum_{u=t-W}^{t-1} \left( \Sigma^{-T} - \Sigma^{-T} \epsilon \epsilon^T \Sigma^{-T} \right)
\]

\[
\frac{\partial L(\Omega)}{\partial \Sigma} \Sigma = \Sigma - \lambda \frac{1}{W} \frac{\partial L(\Omega)}{\partial \Sigma}
\]

2.4.3 Update \( \Theta \) with other parameters being fixed

We denote \( V_{\Theta} = \frac{\partial L(\Omega)}{\partial \Theta} \) as the partial derivative for \( \Theta \), and it is derived as Equation (14). The elements in \( V_{\Theta} \)

\[
\frac{\partial L(\Omega)}{\partial \Theta} = \sum_{u=t-W}^{t-1} \frac{\partial \left( \ln |\Sigma| + \epsilon T \Sigma^{-1} \epsilon \right)}{\partial \Theta}
= \sum_{u=t-W}^{t-1} \left( \Sigma^{-T} - \Sigma^{-T} \epsilon \epsilon^T \Sigma^{-T} \right)
\]

\[
\frac{\partial L(\Omega)}{\partial \Theta} \Theta = \Theta - \lambda \frac{1}{W} \nabla_{\Theta}
\]

2.4.4 Update \( \Psi \) with other parameters being fixed

We denote \( V_{\Psi} = \frac{\partial L(\Omega)}{\partial \Psi} \) as the partial derivative for \( \Psi \). Similarly with Equations (14) and (17), the elements in \( V_{\Psi} \)
are derived as Equation (18), where \( M_3 = \Theta^{t} \odot \zeta \) is a matrix shown in Equation (19). Then we update \( \Psi \) by Equation (20).

\[
\frac{\partial L(\Omega)}{\partial \Psi} = \sum_{u=t-W}^{t-1} \nabla_{\Psi} \left( -2 \Sigma^{-1} (y_u - \hat{y}_u) 1_2^T \right)
\]

\[
\frac{\partial L(\Omega)}{\partial \Psi} \Psi = \Psi - \lambda \frac{1}{W} \nabla_{\Psi}
\]

\(^1\) The proof of Equation (12) in this paper uses the Formula (85) in Matrix Cookbook [11].

\(^2\) The proof of Equation (14) in this paper uses Formulas (57) and (61) in Matrix Cookbook [11].
2.4.5 Update $\Phi$ with other parameters being fixed

We denote $\nabla_\Phi = \frac{\partial l(\Omega)}{\partial \Phi}$ as the partial derivative for $\Phi$. Similarly with Equations (14) and (15), the elements in $\nabla_\Phi$ are derived as Equation (21), where $M_3(i, j) \in R^{N \times r}$ is a matrix shown in Equation (22). Then we update $\Phi$ by Equation (23).

$$[\nabla_\Phi]_j = \sum_{u=t-W}^{t-1} \text{Tr} \left( -2 \Sigma^{-1} (\mathbf{y}_u - \hat{\mathbf{y}}_u) (\mathbf{y}_u - \hat{\mathbf{y}}_u)^T M_3(i, j) \right)$$  
\hspace{10cm} (21)

$$[M_3(i, j)]_{kl} = \begin{cases} \left[ AX_{u-r;u-1} \right]_{kl} & \text{if } k = i \text{ and } j = l \\ 0 & \text{otherwise} \end{cases} \hspace{10cm} (22)$$

$$\Phi = \Phi - \lambda \frac{1}{W} \nabla_\Phi \hspace{10cm} (23)$$

2.5 Sliding-window approach

2.5.1 Initialization

As shown in Equation (3), $E_{r-\max(p,q);t-1}$ and $\Omega$ are needed in forecasting of TM-ARMA in day $t$. So we should initialize $E$ and $\Omega$ before the forecasting. We arrange the first S days for initialization, and forecast the daily returns from day $S + 1$ to day $T$, and $S > \max(p, q, r) + W$.

Firstly, we build $N (p + r)$-dimensional linear regression models with the input of $Y_{1:S-1}$ and $X_{1:S-1}$, then we set the prediction error as $E_{2:S}$. Secondly, we build $N (p + q + r)$-dimensional linear regression models with the input of $Y_{1:S-1}$, $E_{1:S-1}$ and $X_{1:S-1}$. Thirdly, we update $E_{2:S}$ by the new prediction error, set the covariance matrix of $E_{2:S}$ as $\Sigma$, set the intercept term as $\zeta$, and set the coefficients of $Y_{1:S-1}$, $E_{1:S-1}$ and $X_{1:S-1}$ as the elements of $\Theta, \Psi$ and $\Phi$, respectively.

2.5.2 Forecast and parameters updating approach

To better simulate the real trading, we propose a sliding-window online forecast approach, which is shown in Figure 1 and Algorithm 1. In TM-ARMA, the adjacent matrix $A$ and parameter matrices $\Omega$ are updated every day, as shown in Algorithm 1, where $G$ is the training frequency. The model forecasts the returns for the next day from day $S + 1$, and slides forward day by day. The model trains the parameter matrices based on the data in the past $W$ days before the forecasting in every morning.

Algorithm 1. Training and forecast algorithm of TM-ARMA

Input: $Y, X, p, q, r$
Output: $\zeta, \Theta, \Psi, \Phi, \Sigma$

1: Initialize: $\zeta, \Theta, \Psi, \Phi, \Sigma, E$
2: for $t = S, S + 1, S + 2, ..., T$ do
3: Update $A$ based on Equation (2).
4: for $g = 1, 2, ..., G$ do
5: Compute the partial derivatives of $\zeta, \Theta, \Psi, \Phi$ and $\Sigma$ based on Equations (10), (15), (18), (21) and (12), respectively;
6: Update $\zeta, \Theta, \Psi, \Phi$ and $\Sigma$ based on Equations (11), (17), (20), (23) and (13), respectively.
7: end for
8: Compute $\hat{\mathbf{y}}_t$ based on Equation (3);
9: Update $\epsilon_t$ based on Equation (6);
10: end for

3 EMPIRICAL IMPLEMENTATION AND RESULTS

3.1 Dataset and preprocessing

Our dataset is based on the constituent stocks of the FTSE China A50 Index from January 1, 2019 to December 31, 2020. There are 487 trading days in our dataset, including the first $S$ days arranged for initialization. The first official announcement about COVID-19 is in December 31, 2019 by the Wuhan Municipal Health Commission. So we divide our dataset into two stages, namely the Pre-COVID-19 Stage from January 1, 2019 to December 30, 2019 (243 trading days), and the Post-COVID-19 Stage from December 31, 2019 to December 31, 2020 (244 trading days).

3.1.1 Textual data

In the Eastmoney Stock Forum, there is a specific stock forum for every stock listed in the Chinese stock market. The posts in stock forums of unpopular stock are very few, which cannot support the construction of sentiment scores. Many constituent stocks of commodity selection index (CSI) 500 and CSI 800 are not so popular and there are few posts in their stock forums. So we do not take CSI 500 and CSI 800 into account. Because of the attention from investors, stock forums of large capital stocks and hot stocks are active and generate many posts every day. However, hot stocks are not constant and they change over time. When a stock is no longer a hot stock, investors
pay less attention to it and posts in its forum become very few. As a result, we cannot build textual sentiment for hot stocks and have to choose the large capital stocks. Another consideration for choosing large capital stocks is that large capital stocks have fewer suspensions, and the missing data caused by suspension is not good for model’s training and forecasting.

To make the textual sentiment scores effective, the textual data should be as many as possible. We count the total number of posts in the Eastmoney Stock Forum for constituent stocks of the major stock indices in China, and calculate the average daily number of posts per stock, as shown in Table 2. From Table 2, we find that the constituent stocks of FTSE China A50 Index have more average daily posts than the constituent stocks of any other stock indices. So FTSE China A50 is the best choice. We also consider that the FTSE China A50 Index is internationally compiled with its futures contracts traded in Singapore, while the SSE 50 is not representative because it does not include any stock from the Shenzhen Stock Exchange. As a result, we choose the constituent stocks of FTSE China A50 Index as the research object.

The constituent stocks of FTSE China A50 Index are adjusted quarterly but the stocks should be fixed in empirical experiment. Considering that there is only little change of FTSE China A50 Index’s constituent stocks from January 1, 2019 to December 31, 2020, we choose the constituent stocks in December 31, 2020 of the FTSE China A50 Index as our research object.

We obtain 1,686,259 posts in the stock forums for the constituent stocks of the FTSE China A50 Index from January 1, 2019 to December 31, 2020. To make the sentiment scores better reflect the real sentiment of investors, we remove some invalid posts. For example, the institutional accounts of the top 20 users often sends some irrelevant routine news, which has nothing to do with the sentiment of a specific stock and lead to noise. After deleting these posts, we obtain 1,560,365 valid posts.

As we know, words segmentation is necessary for text mining of Chinese textual information. Our words segmentation work is implemented by Pkuseg (https://github.com/lancopku/PKUSeg-python), a powerful words segmentation tool developed by the Language Computing and Machine Learning Group of Peking University. To improve the segmentation accuracy, we apply four external lexicons for the word segmentation, namely Sogou Financial Accounting Lexicon (https://pinyin.sogou.com/dict/detail/index/20659), Baidu Stop Words Lexicon (https://github.com/goto456/stopwords/blob/master/baidu_stopwords.txt) and two emotion dictionaries (which are introduced below).

Colloquial expression and financial terminologies are the two characteristics of the forum posts. Firstly, considering that the language of the forum is colloquial and belongs to non-standard text, we introduce the BosonNLP emotion dictionary (https://bosonnlp.com/dev/resource), a powerful social media polarity emotion dictionary constructed from millions of emotion labeling data from microblog, news, and forums. BosonNLP emotion dictionary includes
many internet terms and informal abbreviations, and has a high coverage of non-standard texts. Secondly, considering that there are many financial terminologies, we apply the Chinese Financial Sentiment Dictionary (https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3446388), a professional Chinese finance emotion dictionary which covers most of the common terms in the securities market. We merge the two dictionaries and obtain the list for positive words and list for negative words, which are applied in the computation of textual sentiment scores.

### 3.1.2 Price data

The trading time of Chinese stocks is from 9:30 to 15:00, but the textual data is available everyday 24 hours, which are shown in Figure 2. The announcement of the new policies and breaking news often occur during 15:00 to 9:30 of the next day, which led to a heated discussion on the stock forums. So the textual data from 15:00 to 9:30 of the next day is very important for forecasting. Therefore, TM-ARMA predicts the daily return rate at 9:30 in every trading day.

Following earlier research [4], we set $y_t$ as the daily log return times 100, as shown in Equation (24), where $\text{Open}_t \in \mathbb{R}^{N \times 1}$ is the prices at 9:30 in day $t$. It should be noted that when computing the profits for the strategies, we use the original daily returns $(\text{Open}_{t+1}/\text{Open}_t - 1)$, which is consistent with the real trading.

$$y_t = 100 \times (\ln (\text{Open}_{t+1}) - \ln (\text{Open}_t)) \quad (24)$$

Our price data is obtained from the database of Tushare, and the data has been adjusted for stock split, rights offerings, and dividends. When suspension happens to some stocks in a particular day, we set the daily returns and corresponding predicted values of the suspended stocks to 0. This solution solves the problem of missing data, and do not influence the training and forecasting of normal stocks, and it is consistent with the real trading situations.

In fact, the Chinese stock market implements the call auction from 9:15 to 9:25 and the continuous bidding starts at 9:30. Our model sells the stocks of the previous day during the call auction (between 9:15 and 9:25) and buys today’s stocks at 9:30. Therefore, the profitability of the model mainly depends on the price gap between 9:30 on the current day and 9:25 on the next trading day. In order to make the time series model works well, the $y_t$ in our model is still set as the price of 9:30 every day. But when evaluating the model’s performance, we calculate the profit based on the price at 9:25 on the next trading day and 9:30 on the current trading day, which is consistent with the real trading scenario.

### 3.2 Empirical approach

We conduct all the experiments by the Python language. AR, ARMA, VAR, EM-ARMA [13] and TM-AR are chosen as the baseline models to compare with TM-ARMA, where TM-AR model is composed of cross-AR term and cross-textual term, as shown in Equation (25). As we know, stationary series is required for ARMA and its extended models. Our empirical results show that the second-order differencing makes the dataset stationary and model converges well. VARMA and VARMAX have cross MA terms. We have tried hard to apply VARMA and VARMAX on our dataset, but they fail to converge on the data even after differencing. However, the models with single MA terms (such as ARMA and TM-ARMA) converge well. As a result, VARMA and VARMAX are not in the baseline models.

$$\mathbf{\hat{y}}_t = \zeta + ((A_1Y_{t-p:t-1} \odot \Theta) \mathbf{1}_1 + ((A_2X_{t-r:t-1}) \odot \Phi) \mathbf{1}_3 \quad (25)$$

Our research shows that too big training window length and hyperparameters do not help models perform better, but lead to heavy computation. According to the conclusions of Markov process, the information decays over time and too old information has little effect on price prediction. Most of traders agree that stock prices are strongly influenced by sentiment over the past 5 days and are correlated with volatility over the past 20 days, as the number of trading days in a week and month is approximately equal.
to 5 and 20, respectively. So we set the grid search scope of hyperparameters as 5, and the length of training window $W = 20$. Considering that $S \geq \max(p, q, r) + W$, we set $S = 25$. The learning rate $\lambda$ is set as 0.001, as a lot of earlier research did. Our research shows that models converge well under the training frequency of 50, so we set $G = 50$. For comparison purposes, all models share the same initial settings, including $W$, $S$, $\lambda$, $G$ and the grid search scope for hyperparameters.

We set a rule-based investment strategy for these models. The following steps are performed at 9:30 in every trading day. Firstly, each model is trained based on the historical data, and forecasts the returns of the new day (from 9:30 to the 9:30 of the next trading day, as shown in Equation (24)). Secondly, the strategy builds a new portfolio for the stocks with positive predicted returns. Thirdly, the strategy allocates all the funds equally on all the stocks in the new portfolio and holds them until 9:30 of the next trading day. We set the original net asset value to “1” for all models’ strategies, and calculate the net asset values daily. Considering that the commissions and stamp duties are about 0.01% and 0.1% of the turnover, respectively, we set the transaction cost rate as 0.01% and 0.11% for buy and sell operations. To save transaction costs in the third step, the strategy sells the stocks that are in the old portfolio and not in the new portfolio, buys the stocks that are in the new portfolio and not in the old portfolio, and buys or sells part of the stocks that are both in new and old portfolios.

### 3.3 Performance evaluation

Models are trained to reduce the prediction error, but the real trading requires the strategies’ profitability. So we evaluate the performance of models from both prediction performance and investment performance. Root Mean Square Error (RMSE) shown in Equation (26) is the most common evaluation indicator for prediction error, where $y_t, \hat{y}_t \in \mathbb{R}^{N \times 1}$ represent the real and predicted returns in day $t$, respectively.

$$\text{RMSE} = \sqrt{\frac{1}{T-S} \sum_{t=S+1}^{T} (y_t - \hat{y}_t)^2} \quad (26)$$

Annualized Return Rate (ARR), Maximum Drawdown Ratio (MDR) and Calmar Ratio (CR) are common indicators for investment performance [26]. The drawdown for day $t$ is the difference of the net asset value in day $t$ and the highest previous net asset value. The MDR equals the biggest drawdown from day $S + 1$ to $T$ divides into the corresponding highest previous net asset value. CR is the quotient of ARR and MDR.

### Table 3 Average performance indicators of the all hyperparameter combinations

| Model    | RMSE ↓ | ARR ↑ | MDR ↓ | CR ↑ |
|----------|--------|-------|-------|------|
| AR       | 26.625 | 0.115 | 0.245 | 0.483 |
| VAR      | 24.926 | 0.126 | 0.189 | 0.702 |
| ARMA     | 23.851 | 0.118 | 0.184 | 0.685 |
| EM-ARMA  | 21.428 | 0.146 | 0.162 | 0.925 |
| TM-AR    | 22.245 | 0.164 | 0.149 | 1.164 |
| TM-ARMA  | 19.354 | 0.173 | 0.142 | 1.235 |

### Table 4 Performance indicators of the hyperparameter combination with the best RMSE

| Model    | RMSE ↓ | ARR ↑ | MDR ↓ | CR ↑ |
|----------|--------|-------|-------|------|
| AR       | 25.925 | 0.124 | 0.251 | 0.494 |
| VAR      | 21.672 | 0.107 | 0.192 | 0.557 |
| ARMA     | 22.349 | 0.115 | 0.184 | 0.625 |
| EM-ARMA  | 21.135 | 0.143 | 0.16  | 0.894 |
| TM-AR    | 21.529 | 0.158 | 0.152 | 1.039 |
| TM-ARMA  | 18.872 | 0.165 | 0.148 | 1.115 |

### Table 5 Performance indicators of hyperparameter combination with the best ARR

| Model    | RMSE ↓ | ARR ↑ | MDR ↓ | CR ↑ |
|----------|--------|-------|-------|------|
| AR       | 26.435 | 0.176 | 0.204 | 0.863 |
| VAR      | 24.742 | 0.192 | 0.174 | 1.103 |
| ARMA     | 23.674 | 0.182 | 0.173 | 1.055 |
| EM-ARMA  | 21.537 | 0.262 | 0.134 | 1.955 |
| TM-AR    | 22.105 | 0.273 | 0.137 | 1.993 |
| TM-ARMA  | 19.093 | 0.309 | 0.134 | 2.297 |

The RMSE is a negative indicator measures the forecast error. ARR is a positive indicator measures the profitability of the model. MDR is a negative indicator measures the sustainability of loss. CR is a positive indicator that measures the stability of profits.

### 4 RESULTS AND ANALYSIS

To compare the general performance of different hyperparameter combinations, we calculate the average performance indicators of all hyperparameter combinations for all models, as shown in Table 3, where “↑” and “↓” mean positive indicators and negative indicators, respectively. The results in Table 3 show that EM-ARMA works much better than AR, ARMA and VAR, and TM-ARMA performs the best.
To compare the prediction performance of all models, we report the performance for the hyperparameter combination of the best RMSE in Table 4. We can find that the prediction performances of EM-ARMA TM-AR are significantly better than the traditional models, and TM-ARMA has the smallest prediction error. It is worth mentioning that the investment performance of TM-ARMA, TM-AR and EM-ARMA are also very good, which means that they give good consideration to both reduction of prediction error and increase of investment returns.

To compare the investment performance of all models, we show the performance for the hyperparameter combination of the best ARR in Table 5. We can find that TM-ARMA outperforms all baseline models. Comparing Tables 4 and 5, we find that the RMSE of EM-ARMA model is smaller than TM-AR, but TM-AR has better investment performance and TM-ARMA performs best in both aspects. To intuitively compare the investment performance, we draw the net asset value curves in Figure 3 for models with the hyperparameter combination same as Table 5. The net value is sampled every 20 trading days when plot net value curves in Figure 3.

From Figure 3, we can find that the traditional models have poor performance, but TM-ARMA, TM-AR and EM-ARMA show better recovery capability under the attract of COVID-19. To verify the impact of COVID-19 on profitability of all models, we compare ARR and excess ARR of all models in two stages in Table 6. The excess ARR equals the difference between ARR of the model and ARR of the FTSE China A50 Index, and it measures the model’s ability to achieve a higher return than the benchmark. From Table 6, we can find that TM-ARMA performs well both in Pre-COVID stage and Post-COVID stage.

5 | CONCLUSIONS

This paper proposes a new TM-ARMA model that considering the cross impacts of textual emotions and price fluctuations, with the adjacent matrix updated dynamically. The textual sentiment scores are computed by a emotion dictionary-based method, and the parameter matrices are estimated by the maximum likelihood method. We conduct the empirical study based on the textual data and daily price data for constituent stocks of the FTSE China A50 Index from January 1, 2019 to December 31, 2020. The results show that TM-ARMA outperforms all the baseline
models, and its performance in the Post-COVID-19 Stage is even better than that in the Pre-COVID-19 Stage.

ACKNOWLEDGMENTS
The authors wish to thank Guodong Long, Guifang Liu, Qin Zhang, and Youle Wang for the suggestions for this paper. This work is supported by National Natural Science Foundation of China (71771091, 71720107002), Guangdong Basic and Applied Basic Research Foundation (2019A1515011752, 2021A1515110876) and Philosophy and Social Science Foundation of Guangdong Province (GD19YYJ11).

DATA AVAILABILITY STATEMENT
The data that supports the findings of this study can be divided into price data and textual data. The price data is available on Ricequant platform (ricequant.com), in the stock form of every specific stock. The textual data is available on Eastmoney Stock Terminal (wind.com.cn), CSMAR database (gtarsc.com) and many other financial databases such as WIND financial terminal (wind.com.cn), CSMAR database (gtarsc.com) and so on. The textual data is available on Eastmoney Stock Forum (guba.eastmoney.com), in the stock form of every specific stock.

ORCID
Weijun Xu https://orcid.org/0000-0001-9489-4487

REFERENCES
1. M. Akhtaruzzaman, S. Boubaker, and A. Sensoy, Financial contagion during covid–19 crisis, Financ. Res. Lett. 38 (2020), 1–29.
2. A. M. Al-Awadhi, K. Al-Saifi, A. Al-Awadhi, and S. Alhamadi, Death and contagious infectious diseases: Impact of the covid-19 virus on stock market returns, J. Behav. Exp. Financ. 27 (2020), 1–5.
3. B. N. Ashraf, Stock markets' reaction to COVID-19: Cases or fatalities? Res. Int. Bus. Financ. 54 (2020), 1–7.
4. E. Bouri, X. V. Vo, and T. Saeed, Return equicorrelation in the cryptocurrency market: Analysis and determinants, Financ. Res. Lett. 38 (2020), 1–8.
5. G. Cheng, Moment consistency of the exchangeably weighted bootstrap for semiparametric m-estimation, Scand. J. Stat. 42 (2015), no. 3, 665–684.
6. S. Dimitrakopoulos and M. Tsionas, Ordinal-response GARCH models for transaction data: A forecasting exercise, Int. J. Forecast. 35 (2019), no. 4, 1273–1287.
7. X. Ganggang, Z. Shang, and G. Cheng, Distributed generalized cross-validation for divide-and-conquer kernel ridge regression and its asymptotic optimality, J. Comput. Graph. Stat. 28 (2019), no. 4, 891–908.
8. X.-L. Gong, X.-H. Liu, and X. Xiong, Measuring tail risk with gas time varying copula, fat tailed garch model and hedging for crude oil futures, Pac. Basin Financ. J. 55 (2019), 95–109.
9. A. Groß-Klußmann, S. König, and M. Ebner, Buzzwords build momentum: Global financial twitter sentiment and the aggregate stock market, Expert Syst. Appl. 136 (2019), 171–186.
10. H. Hassani and E. S. Silva, Forecasting UK consumer price inflation using inflation forecasts, Res. Econ. 72 (2018), no. 3, 367–378.
11. Kaare Brandt Petersen and Michael Syskind Pedersen. The matrix cookbook. Technical University of Denmark, 2012. version 20121115.
12. Y. Kang, R. J. Hyndman, and F. Li, Gratis: Generating time series with diverse and controllable characteristics, Stat. Anal. Data Min. 13 (2020), no. 4, 354–376.
13. H. Lennon and J. Yuan, Estimation of a digitised Gaussian ARMA model by Monte Carlo expectation maximisation, Comput. Stat. Data Anal. 133 (2019), 277–284.
14. X. Li, P. Wu, and W. Wang, Incorporating stock prices and news sentiments for stock market prediction: A case of Hong Kong, Inf. Process. Manag. 57 (2020), 102212.
15. X. Lyu, W. W. Sun, Z. Wang, H. Liu, J. Yang, and G. Cheng, Tensor graphical model: Non-convex optimization and statistical inference, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2019), no. 8, 2024–2037.
16. S. Mohammadi, Neural network for univariate and multivariate nonlinearity tests, Stat. Anal. Data Min. 13 (2020), no. 1, 50–70.
17. R. Östermark and H. Saxén, Varmax-modelling of blast furnace process variables, Eur. J. Oper. Res. 90 (1996), no. 1, 85–101.
18. C. Qian, Q. Tran-Dinh, F. Sheng, C. Zou, and Y. Liu, Robust multivariate support matrix machines, Math. Program. 176 (2019), no. 1–2, 429–463.
19. F. Sheng, S. Zhang, and Y. Liu, Adaptively weighted large-margin angle-based classifiers, J. Multivar. Anal. 166 (2018), 282–299.
20. M. Siering, The economics of stock touting during internet-based pump and dump campaigns, Inf. Syst. J. 29 (2019), no. 2, 456–483.
21. M. A. Thornton, Exact discrete representations of linear continuous time models with mixed frequency data, J. Time Ser. Anal. 40 (2019), 951–967.
22. X. Yang, Y. Zhu, and T. Y. Cheng, How the individual investors took on big data: The effect of panic from the internet stock message boards on stock price crash, Pac. Basin Financ. J. 59 (2020), 1–11.
23. S. Yao, C. Wang, X. Cui, and Z. Fang, Idiosyncratic skewness, gambling preference, and cross-section of stock returns: Evidence from China, Pac. Basin Financ. J. 53 (2019), 464–483.
24. C. Zhang, M. Pham, F. Sheng, and Y. Liu, Robust multiccategroy support vector machines using difference convex algorithm, Math. Program. 169 (2018), no. 1, 277–305.
25. D. Zhang, M. Hu, and Q. Ji, Financial markets under the global pandemic of covid-19, Financ. Res. Lett. 36 (2020), 1–6.
26. F. Zhineng, X. Weijun, H. Rruzqi, G. Long, and J. Jiang, Mhier-encoder: Modelling the highfrequency changes across stocks, Knowl.-Based Syst. 224 (2021), no. 9, 107092.

How to cite this article: W. Xu, Z. Fu, H. Li, J. Huang, W. Xu, and Y. Luo. A study of the impact of COVID-19 on the Chinese stock market based on a new textual multiple ARMA model, Stat. Anal. Data Min.: ASA Data Sci. J. 16 (2023), 5–15. https://doi.org/10.1002/sam.11582