PAPER Special Section on Intelligent Information and Communication Technology and its Applications to Creative Activity Support

Stock Price Prediction by Deep Neural Generative Model of News Articles

Takashi MATSUBARA[†a)], ***Member***, **Ryo AKITA**[†b)], ***Nonmember***, ***and*** **Kuniaki UEHARA**[†c)], ***Member***

SUMMARY In this study, we propose a deep neural generative model for predicting daily stock price movements given news articles. Approaches involving conventional technical analysis have been investigated to identify certain patterns in past price movements, which in turn helps to predict future price movements. However, the financial market is highly sensitive to specific events, including corporate buyouts, product releases, and the like. Therefore, recent research has focused on modeling relationships between these events that appear in the news articles and future price movements; however, a very large number of news articles are published daily, each article containing rich information, which results in overfitting to past price movements used for parameter adjustment. Given the above, we propose a model based on a generative model of news articles that includes price movement as a condition, thereby avoiding excessive overfitting thanks to the nature of the generative model. We evaluate our proposed model using historical price movements of Nikkei 225 and Standard & Poor's 500 Stock Index, confirming that our model predicts future price movements better than such conventional classifiers as support vector machines and multilayer perceptrons. Further, our proposed model extracts significant words from news articles that are directly related to future stock price movements.

key words: *stock price prediction, news articles, deep learning, generative model*

1. Introduction

Financial markets occupy an important position in modern society; as such, arguably no one is independent of the financial climate. Successfully predicting price movements of financial commodities (i.e., stocks, debt obligations, and derivatives) can potentially avoid the harmful effects that a pending financial crisis could have on daily life, as well as provide economic earnings. Many mathematical models of price movements have been proposed in the past (e.g., [1]). Research has also focused on technical analysis methods for identifying certain patterns in past price movements to predict future price movements [2]–[6]; this is typically founded on the belief that the same price movement patterns are often repeated, but the financial market is not a closed system and is instead very sensitive to various events, including corporate buyouts, new product releases, and the like. Even with their success in predicting long-term trends, the pattern-based approaches are limited in terms of predicting event-related and short-term price movements.

The financial market is considered to be efficient [7]. Investors read all publicly available information, such as news articles, to learn about events both in the past and the future, then react via the corresponding financial commodity. The price of the financial commodity then responds to these events rather quickly. Based on this relationship, machine learning approaches have been directed to predict price movements by employing classifiers given news articles [8]–[14]. In other words, these approaches build a model describing the indirect influence that news articles have on prices; however, a large number of news articles are published daily, and each article contains rich information regardless of whether or not it is related to the financial market. More specifically, machine learning approaches are given numerous news articles as explanatory variables that explain the limited number of price movements; such approaches are prone to overfitting to price movements used for parameter adjustment and have not been generalized to accurate prediction of future price movements. In addition, such a model is often a black box, i.e., the model does not provide a reason for its predictions. To counter this problem, several studies have focused on extracting limited features from news articles, such as the number of occurrences of hand-selected words; the other study have averaged features across many news articles [9], [10], [13]. Using these feature selection approaches have the risk of corrupting the information contained in the original news articles.

Given the above, in this paper, we propose a deep neural generative model (DGM) of news articles to predict the price movements; to our knowledge, this is the first time a DGM has been used to tackle such a problem. The DGM is an implementation of generative model on deep neural networks [15]–[17]. Our proposed model generates news articles embedded to vectors [18], [19] given the assumption of future price movement as a condition. Thanks to the nature of generative modeling, our proposed model is expected to have a lower risk of overfitting to past price movements for training.

We evaluate our proposed model using historical datasets of *Nikkei 225* (Nikkei Stock Average) and *Standard & Poor's 500 Stock Index*, as well as related news articles. Our experimental results demonstrate that our proposed model better predicts of the movements of these stock indices versus two key conventionally baseline methods, i.e., support vector machines (SVMs) [3], [4], [9], [10] and multilayer perceptrons (MLPs) [11]–[14]. Results of a sim-

Manuscript received March 8, 2017.
Manuscript revised July 10, 2017.
Manuscript publicized January 19, 2018.

[†]The authors are with the Graduate School of System Informatics Science, Kobe University, Kobe-shi, 657–8501 Japan.
a) E-mail: matsubara@phoenix.kobe-u.ac.jp
b) E-mail: akita@ai.cs.kobe-u.ac.jp
c) E-mail: uehara@kobe-u.ac.jp

DOI: 10.1587/transinf.2016IIP0016

plified market simulation [8], [13] also demonstrate that our proposed model is more capable of making profit versus baseline methods. Finally, arithmetic operations on the generated vectors that embed artificial news articles confirm that our proposed model can visualize significant words contributing to the prediction, i.e., words determining sentiments of the given news articles.

2. Models and Methods

2.1 Generative Model of News Articles and Stock Prices

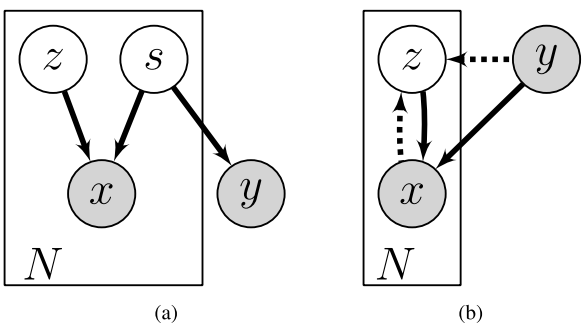
In this study, we propose a generative model $p_\theta(\boldsymbol{x})$ of a set of news articles $\boldsymbol{x} = \{x_i\}$ published in one day for the binary prediction of stock price movements with output y indicating either a predicted increase (i.e., $y = +1$) or a predicted decline (i.e., $y = -1$). On each day, events $i = 0, 1, \ldots$ related to the company and the society at large occur independently and appear in news articles $\boldsymbol{x} = \{x_i\}$. In our model, each news article x_i contains economic information $s_i \in \{+1, -1\}$ and neutral information $z_i \in \mathbb{R}^{n_z}$. Economic information s_i is related to the company's performance and directly motivates investment behavior, whereas neutral information z_i is the remaining information that does not directly motivate investment behavior, including the company's name, business type, etc.

Given the above, in this study, we propose the following generative model of news article x_i, which we also present in Fig. 1 (a); the generative model is

$$\begin{aligned} p_\theta(x_i) &= \sum_{s_i}\sum_{z_i} p_\theta(x_i, z_i, s_i) \\ &= \sum_{s_i}\sum_{z_i} p_\theta(x_i|z_i, s_i)p(z_i)p(s_i), \end{aligned}$$

where $p(s_i)$ and $p(z_i)$ are the prior distributions of economic information s_i and neutral information z_i of individual news article x_i, respectively.

Investors find economic information $\boldsymbol{s} = \{s_i\}$ by reading a set of news articles $\boldsymbol{x} = \{x_i\}$ published on one day, then respond by purchasing or short-selling stocks, all of which influence the stock price on the following day. In other words, economic information $\boldsymbol{s}$ influences stock price movement $y \in \{+1, -1\}$ on the following day through investor behavior (as denoted by the solid arrow across the border of the plate in Fig. 1 (a)). Although it is difficult for non-experts to find economic information s_i from news article x_i, we approximate economic information s_i by the stock price movement y on the following day.



(a) (b)

Fig. 1 (a) Our proposed generative model of a news article and (b) a simplified version.

As a result, as shown in Fig. 1 (b), we consider a conditional generative model $\log p_\theta(\boldsymbol{x}|y = k)$ of a set of the news articles $\boldsymbol{x}$ given stock price movement y on the following day as:

$$\begin{aligned} \log p_\theta(\boldsymbol{x}|y = k) &= \sum_i \log p_\theta(x_i|y = k) \\ &= \sum_i \log \frac{p_\theta(x_i|z_i, y = k)p(z_i)}{p_\theta(z_i|x_i, y = k)}. \end{aligned}$$

Based on the variational method, the conditional generative model $\log p_\theta(\boldsymbol{x}|y = k)$ can be modeled with conditional probability q_ϕ as:

$$\begin{aligned} &\log p_\theta(\boldsymbol{x}|y = k) \\ &\quad = \sum_i \log p_\theta(x_i|y = k) \\ &\quad = \sum_i \mathbb{E}_{q_\phi(z_i|x_i, y=k)}\left[\log \frac{p_\theta(x_i|z_i, y = k)p(z_i)}{p_\theta(z_i|x_i, y = k)}\right] \\ &\quad = \sum_i \Big[- D_{KL}(q_\phi(z_i|x_i, y = k)\|p_\theta(z_i|x_i, y = k)) \\ &\qquad\quad + D_{KL}(q_\phi(z_i|x_i, y = k)\|p(z_i)) \\ &\qquad\quad + \mathbb{E}_{q_\phi(z_i|x_i, y=k)}\left[\log p_\theta(x_i, z_i|y = k)\right]\Big] \\ &\quad \geq \sum_i \Big[- D_{KL}(q_\phi(z_i|x_i, y = k)\|p(z_i)) \\ &\qquad\quad + \mathbb{E}_{q_\phi(z_i|x_i, y=k)}\left[\log p_\theta(x_i, z_i|y = k)\right]\Big] \\ &\quad =: -\mathcal{L}_k(\boldsymbol{x}), \end{aligned} \tag{1}$$

where $D_{KL}(\cdot\|\cdot)$ is the Kullback-Leibler divergence and $-\mathcal{L}_k(\boldsymbol{x})$ is the lower bound.

2.2 Predicting of Stock Prices and the Objective Function

Future price movement y is known in the training dataset, whereas in the testing dataset, future price movement y is not known and is to be predicted. We first approximate log-likelihood $\log p_\theta(\boldsymbol{x}|y = k)$ of the set of news articles $\boldsymbol{x} = \{x_i\}$ by its lower bound $\mathcal{L}_k(\boldsymbol{x})$. Next, using Bayes' theorem, we obtain an approximation of log-probability $\log p(y = k|\boldsymbol{x})$ of the assumption that future stock price movement y takes $k \in \{+1, -1\}$ as follows:

$$\begin{aligned} &\log p_\theta(y = k|\boldsymbol{x}) \\ &\quad = \log \frac{p_\theta(\boldsymbol{x}, y = k)}{p_\theta(\boldsymbol{x}, y = k) + p_\theta(\boldsymbol{x}, y = -k)} \\ &\quad = \log \frac{p_\theta(\boldsymbol{x}|y = k)}{p_\theta(\boldsymbol{x}|y = k) + p_\theta(\boldsymbol{x}|y = -k)} \end{aligned}$$

$$= \log p_\theta(\boldsymbol{x}|y{=}k) - \log\left(p_\theta(\boldsymbol{x}|y{=}k) + p_\theta(\boldsymbol{x}|y{=}-k)\right)$$
$$\simeq -\mathcal{L}_k(\boldsymbol{x}) - \log\left(\exp(-\mathcal{L}_k(\boldsymbol{x})) + \exp(-\mathcal{L}_{-k}(\boldsymbol{x}))\right), \quad (2)$$

where we assume that prior probability $p(y = k)$ of stock price movement y is 0.5. From the approximation, when we find $-\mathcal{L}_{+1}(\boldsymbol{x}) > -\mathcal{L}_{-1}(\boldsymbol{x})$, we consider $\log p_\theta(y = +1|\boldsymbol{x}) > \log p_\theta(y = -1|\boldsymbol{x})$ and predict an increase in the stock price on the following day. In short, the summation of the log-likelihood $\sum_i \log p_\theta(x_i|y = k)$ of each news article x_i published on a given day matters since $\log p_\theta(\boldsymbol{x}|y = k) = \sum_i \log p_\theta(x_i|y = k)$.

From the above, we propose the objective function below, which must be minimized:

$$\mathcal{J}_k = \mathcal{L}_k + \omega \log\left(\exp(-\mathcal{L}_k) + \exp(-\mathcal{L}_{-k})\right). \quad (3)$$

Here, our proposed model can be trained as a conditional generative model or a classifier by adjusting ω as follows: Our proposed model is a conditional generative model as long as $\omega = 0.0$, whereas our proposed model with $\omega = 1.0$ is then a classifier. When $0.0 < \omega < 1.0$, our proposed model works as either a classifier or a generative model with a penalty term.

2.3 Deep Neural Generative Model

As noted above, we propose a new deep neural generative model (DGM), which is an implementation of a generative model using deep neural networks [15]–[17]. More specifically, we implement the aforementioned conditional generative model $\log p_\theta(\boldsymbol{x}|y = k)$ using two neural networks called an *encoder* and a *decoder*, as depicted in Fig. 2. The encoder is given news article x_i and assumption of stock price movement y, then outputs parameters of posterior distribution $q_\phi(z_i|x_i, y)$ of neutral information z_i, inferring posterior distribution $q_\phi(z_i|x_i, y)$. The decoder is given assumption of stock price movement y and a sample z_i from posterior distribution $q_\phi(z_i|x_i, y)$, then generates posterior distribution $p_\theta(x_i|z_i, y)$ of news article x_i.

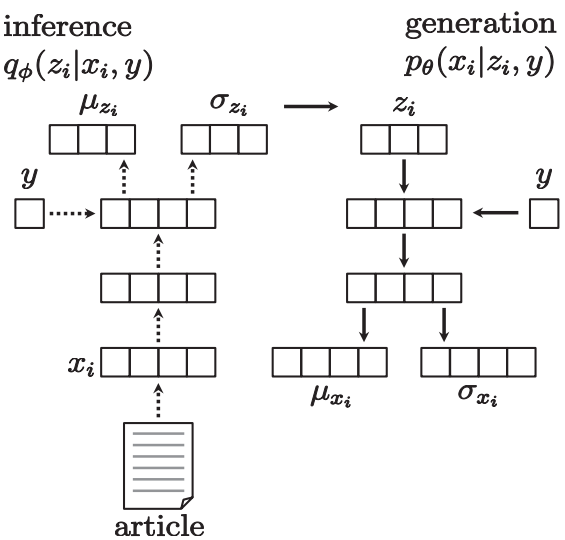


Fig. 2 Architecture of deep neural networks representing our proposed generative model.

The encoder and decoder have u_h hidden layers. The l-th hidden layer consists of $n_h^{(l)}$ units followed by layer normalization [20] and the ReLU activation function [21]. The encoder accepts news article x embedded to an n_x-dimensional vector at its first hidden layer and assumption of stock price movement y at its last hidden layer. The output layer of the encoder consists of $2 \times n_z$ units, with half of the units followed by no activation function and used as vector μ_{z_i} and the other half of the units followed by the exponential function and used as vector σ_{z_i}. Vectors μ_{z_i} and σ_{z_i} are used as parameters of a multivariate Gaussian distribution with a diagonal covariance matrix that represents posterior distribution $q_\phi(z_i|x_i, y)$ of neutral information z_i corresponding to news article x_i. Next, the decoder accepts samples from posterior distribution $q_\phi(z_i|x_i, y)$ and assumption of stock price movement y at its first hidden layer. The output layer of the decoder consists of $2{\times}n_x$ units that are used as parameters of a multivariate Gaussian distribution with a diagonal covariance matrix that represents posterior distribution $p_\theta(x_i|z_i, y)$ of news article x_i in the same way as the encoder.

Using the above, we sampled latent variable z from posterior distribution $q_\phi(z_i|x_i, y)$ exactly $c = 5$ times and calculated lower bound $-\mathcal{L}_k$ of conditional log-likelihood $\log p(\boldsymbol{x}_i|y = k)$ using importance weighted sampling [22]. The encoder and decoder were jointly trained using the Adam optimization algorithm [23] with parameters $\alpha = 10^{-4}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$. We selected hyperparameters from $u_h \in \{1, 2\}$, $n_h^{(l)} \in \{50, 100, 200, 400\}$, $n_z \in \{5, 10, 25, 50, 100\}$, and $\omega \in \{0.0, \ldots, 0.5, \ldots, 1.0\}$, where $n_h^{(l)} \leq n_h^{(l')}$ for $l > l'$.

Our proposed DGM was given a single article at a time and was trained via the objective function in Eq. (3); we call this *the single article*. We also evaluated a simple preprocessing approach that was used in previous studies [12], [13], i.e., we calculated vector $\bar{x}$ as the average of all vectors $\boldsymbol{x} = \{x_i\}$, each embedding a news article published in a given day; we call this *the averaged article*.

3. Experiments and Results

3.1 Models for Comparison

As noted previously, our proposed DGM considers binary change y in the stock prices as a sample from prior distribution $p(y)$ rather than as a result of news articles $\boldsymbol{x}$. For comparison, we also evaluated straightforward classifiers $p(y|\boldsymbol{x})$ used in previous studies as baselines, in particular multilayer perceptrons (MLPs) [11]–[14] and support vector machines (SVMs) [3], [4], [9], [10], each of which we further describe below.

The MLP that we implemented had hidden layers, each of which consisted of $n_h^{(l)}$ units followed by layer normalization [20] and the ReLU activation function [21], just like our proposed DGM. The MLP had a single output unit followed

Table 1 Periods and numbers of news articles included in our datasets.

	Nikkei			S&P 500		
	training	validation	test	training	validation	test
start date	4 Jan. 2001	4 Jan. 2007	4 Jan. 2008	20 Oct. 2006	19 Jun. 2012	22 Feb. 2013
end date	29 Dec. 2006	28 Dec. 2007	30 Dec. 2008	18 Jun. 2012	21 Feb. 2013	21 Nov. 2013
#business days	1,660	236	236	1,426	169	191
#articles	56,666	8,281	8,292	122,550	47,781	48,523
% of increase	48.1	51.3	47.9	54.7	53.8	59.7

by the logistic function. For the single article, the output of the MLP represented posterior probability $q_\phi(y = +1|x_i)$ of the increase in stock price given single news article x_i. The objective function to be minimized was cross-entropy $\sum_i \sum_k -\mathbb{I}(y = k) \log q_\phi(y = k|x_i)$, where $\mathbb{I}(cond.)$ is the indicator function that returns 1 if *cond*. is true and 0 otherwise. The other conditions were the same as those for our proposed DGM.

Once the MLP was trained, we sequentially inputed a set of news articles $\boldsymbol{x} = \{x_i\}$ published in a given day and predicted stock price movement using the summation of the log-probability using the same approach as our proposed DGM, i.e., $\sum_i[\log q_\phi(y = +1|x_i)] > \sum_i[\log q_\phi(y = -1|x_i)]$ was considered to imply $q_\phi(y = +1|\boldsymbol{x}) > q_\phi(y = -1|\boldsymbol{x})$ and vice versa. For the averaged article, we trained the MLP using the same approach as that used for the single article, and single output $q_\phi(y|\bar{x})$ was used as the prediction for the given day.

As for the SVM, we only trained the SVM for the averaged article because convergence of training took a very long time for the single article. Further, we selected parameter C by trading off between classification accuracy and margin maximization from $C \in \{\ldots, 1, 2, 5, 10, 20, 50, 100, \ldots\}$. Note that we also evaluated the fully generative version of our proposed DGM, i.e., our proposed DGM with a hyperparameter of $\omega = 0.0$.

3.2 Datasets

We evaluated our proposed DGM and the other models on two datasets, i.e., *Nikkei* and *S&P 500*. The Nikkei dataset includes *Nikkei 225* daily stock index (Nikkei Stock Average) and news articles from the morning edition of the *Nihon Keizai Shimbun* (Nikkei) newspaper, which is in Japanese. Just as in the previous studies [12], [24], we only extracted the titles from these news articles. Titles were preprocessed using the morphological analyzer MeCab [25] and embedded to n_x-dimensional vectors using the Paragraph Vector algorithm [19]. The Paragraph Vector algorithm used a vector length of 200, negative sampling, a window size of 3, and a learning rate that linearly decreased from 0.1 to 0.0001 over 300 iterations. Since we obtained the news articles from the morning edition, we were able to use these news articles to predict the price movements on the same day. Increase $y = +1$ (or decline $y = -1$) in the stock index was defined as being the situation in which the closing price was higher (or lower) than the corresponding opening price.

We obtained the S&P 500 dataset from [26]; it includes *Standard & Poor's 500 Stock Index* and financial web news obtained from both Reuters [27] and Bloomberg [28], each of which is in English. As with the Nikkei dataset, we also embedded the titles to vectors using the Paragraph Vector algorithm. Further, we used these news articles to predict price movements on the next business day.

3.3 Prediction Performance

To evaluate the prediction performance for the datasets, each dataset was divided into three subsets for training, validation, and testing. Table 1 summarizes the time period and the number of articles for each of these subsets. The prediction performance of each of these models was evaluated by measuring prediction accuracy, which is defined as

$$ACC = \frac{TP + TN}{TP + FP + FN + TN}, \tag{4}$$

where TP, TN, FP, and FN are the number of true positives, true negatives, false positives, and false negatives. Since classes $y = +1$ and $y = -1$ differ in terms of size, we also used the Matthews Correlation Coefficient (MCC) to evaluate prediction performance. The MCC is a well-known balanced measure defined as

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}. \tag{5}$$

We trained each of the models using the corresponding training subset and various hyperparameters, where we adjusted the imbalances in the classes and the imbalance in the number of news articles per day via oversampling. The hyperparameters were selected according to the MCC for the validation subset. Once the hyperparameters were selected, we retrained the models using both the training and validation subsets.

Table 2 summarizes the selected hyperparameters, with the numbers in square brackets indicating the number of the units of the encoders of our proposed DGMs or the MLPs. Tables 3 and 4 summarize prediction performance, with sin. and ave. indicating the single article and the averaged article scenarios. Note that our proposed DGM with a modified objective function of $\omega = 0.9$ achieved the best prediction accuracy for the testing subsets of both datasets for the single article scenario.

Table 2 Hyperparameters.

Model		Nikkei	S&P 500
SVM	ave.	$C = 2$	$C = 10$
MLP	ave.	[200,50,1]	[400,10,1]
	sin.	[50,10,1]	[100,10,1]
$DGM_{\omega=0.0}$	ave.	[400,100]	[100,100,50]
	sin.	[200,100,100]	[100,100,50]
DGM	ave.	[400,50], $\omega=0.9$	[400,100,50], $\omega=0.9$
	sin.	[100,50], $\omega=0.9$	[200,100,50], $\omega=0.9$

Table 3 Prediction performance (accuracy as %).

		Nikkei		S&P 500	
Model	article	validation	test	validation	test
SVM	ave.	54.2	49.2	53.0	59.0
MLP	ave.	60.6	50.4	56.0	55.3
	sin.	62.7	49.2	58.9	52.1
$DGM_{\omega=0.0}$	ave.	61.9	50.0	64.9	51.1
	sin.	62.3	45.8	63.7	51.1
DGM	ave.	58.9	51.3	61.3	48.4
	sin.	62.3	56.4	58.9	61.1

Table 4 Prediction performance (MCC).

		Nikkei		S&P 500	
Model	article	validation	test	validation	test
SVM	ave.	0.091	−0.024	−0.023	0.037
MLP	ave.	0.216	0.039	0.128	0.047
	sin.	0.264	−0.050	0.195	−0.017
$DGM_{\omega=0.0}$	ave.	0.239	0.001	0.293	0.024
	sin.	0.245	−0.088	0.284	0.007
DGM	ave.	0.182	0.030	0.228	−0.114
	sin.	0.247	**0.127**	0.264	**0.138**

3.4 Market Simulation

We also evaluated performance of the models based on whether or not they were capable of making a profit. Following the framework in the previous study [8], we simulated a day trader strategy using each of the models to predict a stock index. More specifically, the day trader creates a portfolio of securities that represents a stock index or considers a financial derivative product that tracks a stock index. If the model predicts an increase in the price of a stock index on a given business day, the simulated day trader purchases a unit of the derivative product at the opening price, holds it for the business hours, then sells it at the closing price. If the stock index will be changed by more than a certain threshold θ during the business hours, our simulated day trader immediately sells the derivative product for a profit-taking or a loss-cutting. Our simulated day trader uses the same strategy for short-selling. We evaluated the capacity of making profit for the testing subset of each dataset with thresholds $\theta = 1\%$ and $\theta = 2\%$. For comparison, we also examined trivial strategies, including random trading (buying and selling with an equal probability of 0.5), all-buying (buying with a probability of 1.0 and selling with a probability of 0.0), and all-selling (selling with a probability of 1.0 and buying with a probability of 0.0). Table 5 summarizes our results, comparing random trading and all-buying/all-selling with our other models. Note that random trading had a mean of zero because the market simulation did not take into account trading commissions. Further, we added the standard deviations of the profits of the random trading strategy. Our proposed DGM with a modified objective function of $\omega = 0.9$ for the single article showed the largest profit between both datasets. Following this were the SVM and the MLP for the averaged article.

Table 5 Prediction performance (profit as %).

		Nikkei		S&P 500	
Model	article	$\theta = 1\%$	$\theta = 2\%$	$\theta = 1\%$	$\theta = 2\%$
random trading					
mean	—	0.00	0.00	0.00	0.00
std.	—	6.38	7.51	5.05	5.19
all-buying	—	−12.06	−11.61	19.37	17.35
all-selling	—	12.06	11.61	−19.37	−17.35
SVM	ave.	16.90	−2.73	17.77	14.32
MLP	ave.	16.66	16.57	15.22	11.98
	sin.	4.42	−1.23	8.78	5.47
$DGM_{\omega=0.0}$	ave.	−11.58	−4.49	2.06	−0.65
	sin.	−22.68	−34.21	−1.94	−1.42
DGM	ave.	0.75	−2.96	0.10	−4.29
	sin.	**37.06**	**16.63**	**22.96**	**20.33**

3.5 Vector Representation of News Articles

In the previous subsections, we have demonstrated that our proposed DGM was able to predict stock price movement y according to news articles embedded to vectors $\boldsymbol{x} = \{x_i\}$ using the Paragraph Vector algorithm [19]. Therefore, our proposed DGM can be described as extracting features (i.e., economic information $\boldsymbol{s}$) related to stock price movement y from vectors $\boldsymbol{x}$. In this subsection, we consider artificial news article $\tilde{x}_i$ that describes the same neutral information z_i and opposite economic information $\tilde{s}_i = -s_i$. The words and meanings that vary between original article x_i and paired artificial news article $\tilde{x}_i$ are considered to be related to economic information s_i and stock price movement y. Given this, we attempted to extract vector r_i that represents such features from news article x_i as:

$$r_i = \mathbb{E}_{q_\phi(z_i|x_i,y=k)}[\arg\max_x p_\theta(x|z_i, y{=}k) - \arg\max_x p_\theta(x|z_i, y{=}-k)]. \quad (6)$$

Here, artificial news article represented by vector $\tilde{x}_i = x_i - r_i$ is expected to represent opposite economic information $\tilde{s}_i = -s_i$. Therefore, we selected several news articles x_i from each testing subset; next, we generated vectors $\tilde{x}_i = x_i - r_i$ that represent artificial news articles, and retrieved news articles embedded to the vectors that are closest to the vectors $\tilde{x}_i$ in cosine similarity. Table 6 summarizes our results.

Table 6 Modification of news articles embedded to vectors.

Dataset	original class	original article $x_i \Rightarrow$ paired article closest to $\tilde{x}_i = x_i - r_i$ (English translations by the authors are in parentheses)
Nikkei	$y = +1$	3 月のビール出荷，2 年 2ヵ月ぶり増—新製品が需要喚起 (Beer shipment increased in March for the first time in 2 years and 2 months—New products aroused demand) ⇒ 第 2 部企業の興亡 (1) 値崩れの波をくぐれ (デフレが蝕む) (Part 2, Enterprises' rise and fall (1): Pass under the wave of the collapsed prices (deflation erodes))
		キリン HD 純利益 12%増，前期 600 億円，年間配当 19–20 円に (Net income of Kirin HD increased by 12%, to 60 billion yen in first half year, annual dividend becoming 19–20 yen) ⇒ TDK の前期，純利益を訂正—333 億円に減額 (Net income in first half year of TDK corrected—decreased to 33.3 billion yen)
	$y = -1$	9 月中間経常，キッコーマン，13%減益—冷夏・円高が打撃 (September ordinary income of Kikkoman decreased by 13 %—hit by cool summer and rising yen) ⇒ 円高一服好感し反転—電機・自動車株が上げ主導 (株式往来) (Rebounding due to good impression by pause of rising yen—Electrical and automotive stocks leading high prices (market overview))
		高成長中国，潜むリスク—日本企業にも影響，鉄鋼は減産，輸出も減少 (Risk lurking in highly growing China—impacting on Japanese companies, steel production cuts, exports also declining) ⇒ 非鉄金属，軒並み高，銅は 7ヵ月ぶり水準，実需買いに影響も (Almost all non-ferrous metals having soaring prices, copper price at a high level for the first time in 7 months, affecting actual buying)
S&P 500	$y = +1$	Nestle, Sara Lee profits lifted by price increases ⇒ Toshiba, Fujitsu hit by price falls, outlook rough
		Hyundai targets 2007 revenue growth spurt ⇒ Qualcomm profit falls, cuts '09 revenue target
	$y = -1$	McClatchy sees ad revenue down in first half of 2007 → SES sees pay TV revenue increasing 34% worldwide in 2016
		January housing starts down 14.3 percent → Instant view: CPI rises; housing starts up 15 percent

4. Discussion

Almost every model evaluated in our present study achieved an accuracy close to 100% for the training subset and a good MCC (> 0.2) for the validation subset with designed hyperparameters. Unfortunately, this does not necessarily indicate that the model predicts stock price movements well. Each model save for our proposed DGM is a classifier and is prone to overfitting to the training subset. Such models sometimes predicted the validation subset well by chance but were not generalized to the testing subset. The MLP and our proposed DGM with $\omega = 0.0$ achieved relatively better generalization abilities for the averaged article than for the single article, because the number of input data points was reduced and the models had a lower risk of overfitting for the averaged article. Our proposed DGM with $\omega = 0.0$ modeled a given dataset regardless of the condition y owing to the high flexibility of deep neural networks; this resulted in limited classification accuracy. In contrast, only our proposed DGM with $\omega > 0.0$ demonstrated significant generalization abilities for the single article, showing the effectiveness of our proposed objective function in Eq. (3). Our proposed DGM with $\omega > 0.0$ can be interpreted as a classifier with a penalty term derived from a formulation of a generative model; therefore, it achieved better results than SVM, MLP, and our proposed DGM with $\omega = 0.0$.

According to the bottom row of Table 1, stock prices increased for approximately 55% of the days in each subset of the S&P 500 dataset. When predictions of a model were biased to stock price increases, the model was able to achieve accuracy levels better than 50%; however, the model with the biased predictions was not given a good evaluation based on the MCC since the MCC is a balanced measure. The SVM and the MLP achieved better accuracy levels and insignificant MCCs for the S&P 500 dataset, implying that their accuracy levels of better than 50% for the S&P 500 dataset were due to biased predictions. Conversely, the DGM with $\alpha = 0.0$ achieved accuracy levels of almost 50%. We highlight here that the DGM with $\alpha = 0.0$ was the only generative model in this study; further, its objective was to reconstruct given news articles x_i, and its predictions were almost balanced because of its prior probability $p(y = +1) = p(y = -1) = 0.5$. In addition, the MCC scores of the DGM with $\alpha = 0.0$ were almost zero, just like the SVM and MLP, indicating that the DGM with $\alpha = 0.0$ failed in predicting the stock price movements, which is why the DGM with $\alpha = 0.0$ achieved accuracy levels of almost 50%. Finally, we note that since the Nikkei dataset is relatively balanced, the SVM, the MLP, and the DGM with $\alpha = 0.0$ all achieved accuracies of almost 50% in the Nikkei dataset.

In the Nikkei dataset with a threshold of $\theta = 2\%$, the MLP for the averaged article earned a profit of 16.57% which is comparable to the 16.63% profit earned by our proposed DGM for the single article despite the large difference in prediction accuracy between our proposed DGM (i.e., 56.4%) and the MLP (i.e., 50.4%). With a threshold of $\theta = 1\%$, the profit earned by the MLP increased only slightly to 16.66%, whereas the profit earned by our proposed DGM increased substantially to 37.06%. All models were trained

to predict the binary change y in stock price independent of the amount of change. Therefore, higher prediction accuracy implies a higher expected profit, but this does not guarantee a higher profit because of the large variance of profits per day. If a model predicts a small change successfully and fails in predicting of a wide swing, the model loses a great deal in spite of its 50% prediction accuracy. We observed that this is why several models earned huge profits or suffered huge losses with a threshold $\theta = 2\%$ in spite of a prediction accuracy of approximately 50%. Following previous studies [8], [13], our market simulation held changes within threshold θ; with a lower threshold θ, the variance of the change in stock price was suppressed and the variance of the profit decreased, resulting in the order of profit resembling the order of prediction accuracy (see also the standard deviation (std.) results in Table 5). Therefore, the profit earned by our proposed DGM with a threshold of $\theta = 1\%$ is more significant than that earned with a threshold of $\theta = 2\%$. The quantitative prediction of stock price movements and market simulations based on such predictions are both beyond the scope of this present study, but we include them as possible future work [29]. A market simulation considering commissions and time lags inherent in trading is also future work [30].

In Sect. 3.5, according to the original news article x_i, we found the paired news article closest to vector $\tilde{x}_i = x_i - r_i$ in cosine similarity. Paired news article $\tilde{x}_i$ is thereby expected to represent a news article describing similar neutral information z_i with opposite economic information $\tilde{s} = -s$. The original and paired news articles share words such as "純利益 (net income)", "億円 (billion yen)", "円高 (rising yen)", "revenue", "housing", and "percent." These words can be considered to correspond to neutral information z_i unrelated to stock price movement y. In contrast, the original and paired news articles also include words with opposite meanings to one another, such as "増 (increase)" versus "減 (decline)" and "starts up" versus "starts down." These word pairs can be considered to correspond to economic information s that are directly related to stock price movement y. Both "increase" and "starts up" do not always have positive meanings, but they are positive when they are used with other positive words, such as "income", "price", and "housing." Therefore, our proposed DGM understands the language model obtained from the Paragraph Vector algorithm [19] and separated features, depending on their relation to stock price movement y.

Turning our attention to viewpoint of graphical model, we can view the MLP as an implicit model in which stock price y is a result of the reactions of investors to news articles $\boldsymbol{x}$. We also consider the model shown in Fig. 1 (a), in which the latent variables $\boldsymbol{z}$ and $\boldsymbol{s}$ influence news articles $\boldsymbol{x}$ and stock price y and are inferred from them using an approach similar to that used in supervised topic models [31]. This model also easily overfits to the training subset due to the massive number of explanatory variables in spite of limited explained variable y. Note that this is why we treat stock price movement y as a condition rather than a result. Finally, society and financial markets are far more complicated than even our best models. As future work, we plan to consider a hierarchical generative model that generates a set of new articles describing the same event and includes individual companies, individual stock prices, multiple stock markets, and the various relationships among all of these components. Our proposed DGM only predicted stock indices on the following day, but can arguably be extended to handle temporal dynamics and the delayed and long-term influences of news articles and stock prices. As such, we also include this as part of our future work.

5. Conclusion

In this study, we propose a deep neural generative model of news articles to predict the stock price movements. Our proposed model is given assumption of future price movements as a condition and generates news articles embedded to vectors. We evaluated our proposed model and other models for comparison by applying them to historical datasets of stock indices. Our proposed model achieved the highest prediction accuracy among all of the models, as well as the most capable of earning a profit in our market simulations. Our proposed model extracts features from news articles embedded to vectors depending on their relation to stock price movements.

Acknowledgments

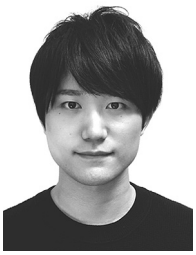
This study was partially supported by the JSPS KAKENHI (16K12487), Kayamori Foundation of Information Science Advancement, and SEI Group CSR Foundation.

References

[1] F. Black and M. Scholes, "The pricing of options and corporate liabilities," Journal of Political Economy, vol.81, no.3, pp.637–654, May 1973.
[2] J.J. Murphy, Technical analysis of the financial markets: A comprehensive guide to trading methods and applications, Prentice Hall Press, 1999.
[3] F.E.H. Tay and L. Cao, "Application of support vector machines in financial time series forecasting," Omega, vol.29, no.4, pp.309–317, 2001.
[4] K.-j. Kim, "Financial time series forecasting using support vector machines," Neurocomputing, vol.55, no.1-2, pp.307–319, 2003.
[5] M.R. Hassan, B. Nath, and M. Kirley, "A fusion model of HMM, ANN and GA for stock market forecasting," Expert Systems with Applications, vol.33, no.1, pp.171–180, 2007.
[6] R.C. Cavalcante, R.C. Brasileiro, V.L.F. Souza, J.P. Nobrega, and A.L. Oliveira, "Computational intelligence and financial markets: A survey and future directions," Expert Systems with Applications, vol.55, pp.194–211, 2016.
[7] E.F. Fama, "The behavior of stock-market prices," Journal of Business, vol.38, no.1, pp.34–105, 1965.
[8] V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan, "Mining of concurrent text and time series," KDD-2000 Workshop on Text Mining, pp.37–44, 2000.
[9] R.P. Schumaker and H. Chen, "Textual analysis of stock market prediction using financial news articles," 12th Americas Conference on Information Systems (AMCIS), vol.27, no.2, pp.1–29, 2006.

[10] R. Luss and A. D'Aspremont, "Predicting abnormal returns from news using text classification," Quantitative Finance, vol.15, no.6, pp.999–1012, March 2012.
[11] A. Yoshihara, K. Fujikawa, K. Seki, and K. Uehara, "Predicting stock market trends by recurrent deep neural networks," Proc. 13th Pacific Rim International Conference on Artificial Intelligence, pp.759–769, 2014.
[12] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Using structured events to predict stock price movement: An empirical investigation," Proc. 2014 Conference on Empirical Methods in Natural Language Processing, pp.1415–1425, 2014.
[13] X. Ding, Y. Zhang, T. Liu, and J. Duan, "Deep learning for event-driven stock prediction," Proc. 25th International Joint Conference on Artificial Intelligence, pp.2327–2333, 2015.
[14] R. Akita, A. Yoshihara, T. Matsubara, and K. Uehara, "Deep learning for stock prediction using numerical and textual information," Proc. IEEE/ACIS 15th International Conference on Computer and Information Science, 2016.
[15] D.P. Kingma and M. Welling, "Auto-encoding variational Bayes," International Conference on Learning Representations, pp.1–14, 2014.
[16] D.P. Kingma, D.J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," Advances in Neural Information Processing Systems, pp.1–9, 2014.
[17] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," Advances in Neural Information Processing Systems, pp.3483–3491, 2015.
[18] T. Mikolov, G. Corrado, K. Chen, and J. Dean, "Efficient estimation of word representations in vector space," Proc. International Conference on Learning Representations, pp.1–12, 2013.
[19] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," Advances in Neural Information Processing Systems, vol.1, pp.3111–3119, Oct. 2013.
[20] J.L. Ba, J.R. Kiros, and G.E. Hinton, "Layer normalization," arXiv, 2016.
[21] V. Nair and G.E. Hinton, "Rectified linear units improve restricted Boltzmann machines," Proc. 27th International Conference on Machine Learning, no.3, pp.807–814, 2010.
[22] Y. Burda, R. Grosse, and R. Salakhutdinov, "Importance weighted autoencoders," International Conference on Learning Representations, pp.1–12, 2015.
[23] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," International Conference on Learning Representations, Dec. 2015.
[24] K. Radinsky, S. Davidovich, and S. Markovitch, "Learning causality for news events prediction," Proc. 21st International Conference on World Wide Web (WWW2012), pp.909–918, 2012.
[25] T. Kudo, "MeCab: Yet another part-of-speech and morphological analyzer," url: http://mecab.sourceforge.net/
[26] S&P 500 dataset, url: http://ir.hit.edu.cn/xding/index_english.htm
[27] Reuters, url: http://www.reuters.com/
[28] Bloomberg, url: http://www.bloomberg.com
[29] J.E. Moody and M. Saffell, "Reinforcement learning for trading," Advances in Neural Information Processing Systems 11, no.1998, pp.917–923, 1999.
[30] Y. Deng, F. Bao, Y. Kong, Z. Ren, and Q. Dai, "Deep direct reinforcement learning for financial signal representation and trading," IEEE Trans. Neural Netw. Learn. Syst., vol.28, no.3, pp.653–664, 2017.
[31] D.M. Blei and J.D. McAuliffe, "Supervised topic models," Advances in Neural Information Processing Systems 20, vol.21, no.1, pp.121–128, 2008.

Takashi Matsubara received his B.E., M.E., and Ph.D. in engineering degrees from Osaka University, Osaka, Japan, in 2011, 2013, and 2015, respectively. He is currently an assistant professor at the Graduate School of System Informatics, Kobe University, Hyogo, Japan. His research interests are in computational intelligence and computational neuroscience.



Ryo Akita was a graduate student of Graduate School of System Informatics, Kobe University, Hyogo, Japan. He received his B.E. and M.E. degrees from Kobe University. He investigated stock trend prediction via news events and natural language processing.

Kuniaki Uehara received his B.E., M.E., and D.E. degrees in information and computer sciences from Osaka University, Osaka, Japan, in 1978, 1980 and 1984, respectively. From 1984 to 1990, he was with the Institute of Scientific and Industrial Research, Osaka University as an Assistant Professor. From 1990 to 1997, he was an Associate Professor with Department of Computer and Systems Engineering of Kobe University. From 1997 to 2002, he was a Professor with the Research Center for Urban Safety and Security of Kobe University. Currently he is a Professor with Graduate School of System Informatics of Kobe University.