PEN: Prediction-Explanation Network to Forecast Stock Price Movement with Better Explainability

Shuqi Li[1*], **Weiheng Liao**[2*], **Yuhan Chen**[1*], **Rui Yan**[1, 3†]

[1]Gaoling School of Artificial Intelligence (GSAI), Renmin University of China
[2]MADE by DATA [‡]
[3]Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education
shuqili@ruc.edu.cn, weiheng@madebydata.com, yuhanchen@ruc.edu.cn, ruiyan@ruc.edu.cn

Abstract

Nowadays explainability in stock price movement prediction is attracting increasing attention in banks, hedge funds and asset managers, primarily due to audit or regulatory reasons. Text data such as financial news and social media posts can be part of the reasons for stock price movement. To this end, we propose a novel framework of Prediction-Explanation Network (PEN) jointly modeling text streams and price streams with alignment. The key component of the PEN model is an shared representation learning module that learns which texts are possibly associated with the stock price movement by modeling the interaction between the text data and stock price data with a salient vector characterizing their correlation. In this way, the PEN model is able to predict the stock price movement by identifying and utilizing abundant messages while on the other hand, the selected text messages also explain the stock price movement. Experiments on real-world datasets demonstrate that we are able to kill two birds with one stone: in terms of accuracy, the proposed PEN model outperforms the state-of-art baseline; on explainability, the PEN model are demonstrated to be far superior to attention mechanism, capable of picking out the crucial texts with a very high confidence.

Introduction

Stock price movement prediction, a hugely challenging but equally rewarding task, has always drawn enormous interest from both academia and industry (Frankel and Frankel 1995; Bollen, Mao, and Zeng 2011; Edwards, Magee, and Bassetti 2018). Classical approaches like time series analysis have continued to remain popular due to their simple structures and intepretability. In recent years, deep learning techniques have gained much popularity due to its ability to improve the prediction accuracy when there are large quantities of training data, as often is the case of financial markets.

For better prediction, an external network becomes a possible solution via alignment of financial news commentaries and social media posts, revealing rich information about the market, far beyond price, trading volume or financial KPIs (Xing, Cambria, and Welsch 2018; Jiang 2021; Si et al. 2013; Schumaker and Chen 2009). The aligned text streams are to some extent explaining the prediction from an external perspective other than the stock price time series. The use of deep text mining algorithms to model text data also makes a massive difference. For this kind of models, we observe two facts. 1) Not all the texts are equally useful in predicting the future, and those unimportant texts shall be filtered out. 2) When investors make investments decisions, they only focus on a small subset of key information.

With these motivations in mind, we propose a model named Prediction-Explanation Network (PEN) to jointly predict the stock price movement and explore probable effect factors. The key component of the proposed PEN model is a shared representation learning (SRL) module to analyze text embeddings and stock price series jointly and learn their shared representation from their correlation. To be more specific, this component selects the stock price data and the corresponding salient text information, fuse them into a single representation vector. To better incorporate stochastic factors, we leverage the variant of Variational Auto-Encoders (VAE) to generate stock movements from latent variables. To sum up, the key contributions of our work include:

- A novel framework for stock price movement prediction with better explainability through alignment of stock price streams and text streams.
- A key shared representation learning (SRL) module which models the interaction between text data and stock price data and produces better text and price embeddings.
- A salient regulator that is inspired by individual investors' decision process and specifically designed to work within the PEN framework in order to maximize the explainability of texts.

We conduct extensive experiments on real-world datasets to verify the effectiveness of the proposed PEN model. Experiments on two distinct datasets spanning two different time periods show that PEN outperforms the existing state-of-art models, demonstrating that focusing on the right information can indeed help to make a better prediction.

Related Work

Stock Price Movement Prediction. Traditional approach tends to focus on identifying patterns and correlations in

[*] These authors contributed equally.
[†] Corresponding author: Rui Yan (ruiyan@ruc.edu.cn)
[‡] https://www.madebydata.com

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

结合文本信息 + 股票价格 预测的相关工作

trading prices and volumes to predict the direction of price movement in the future. In recent years, researchers successfully exploited text information in stock price prediction, taking advantage of the unprecedented increase of online text data such as news and social media posts. This type of models are possible because news, announcements and sometimes rumours may have a direct impact on stock price. Li et al. (2014) projected textual news onto the sentiment space and implement a generic stock price prediction framework. Khadjeh Nassirtoussi et al. (2015) proposed predict intraday directional-movements of a currency-pair in the foreign exchange market based on the text of breaking financial news-headlines. Hu et al. (2018) designed a Hybrid Attention Networks to predict the stock trend based on the sequence of recent related news. Xu and Cohen (2018) took a step further, their multi-layer StockNet ingests both stock price data and tweets data, uses attentions and variational autoencoders to extract latent information and makes prediction. Carta et al. (2021) propose a feature engineering process to create an extended set of features extracted using generated lexicons and news. While models with either just text data, or just stock price data have been demonstrated to yield fairly good prediction results, many believe that models that make sense of both type of datasets can be a game changer. This is because stock price is greatly influenced by both what's happening in the market (which is captured by text data) and past patterns (which can be learnt from historical stock price).

Prediction Explainability. Despite big advances in deep stock price prediction models, their adoption is unfortunately very limited due to their "black box" nature. So far only a tiny number of research articles have attempted to address prediction explainability. Past work includes Hu et al. (2018) analyzed attention weights of text corpora to empirically study the importance of different news articles, and Dang, Shah, and Zerfos (2019) proposed a multi-modality neural model to discover news relevant to stock prediction.

In this paper, we take a step further to propose a Shared Representation Learning (SRL) module, which jointly analyses text embeddings and stock price series and learns their shared representation. This module generates a Vector of Salience (VoS) to explain the importance of individual text documents in the corpora, which is then further regulated in the learning objective to maximize explainability and improve prediction accuracy.

Prediction-Explanation Network

Problem Description

We formulate the stock price movement prediction problem as a classification task. For one stock, given the past text corpora C and historical prices dataset $P\,[t-L,\ t-1]$ where L is the fixed lag size, the objective is to predict the movement y of the stock on next day t. The movement y can be constructed as binary:

$$y = \begin{cases} 1, & p_t > p_{t-1} \\ 0, & p_t \leq p_{t-1} \end{cases} \tag{1}$$

where p_t denotes the adjusted close prices at day t for the given stock. The adjusted close price amends a stock's closing price to reflect that stock's value after accounting for any corporate actions, which is often used as an important element to predict stock price movement (Yoo et al. 2021).

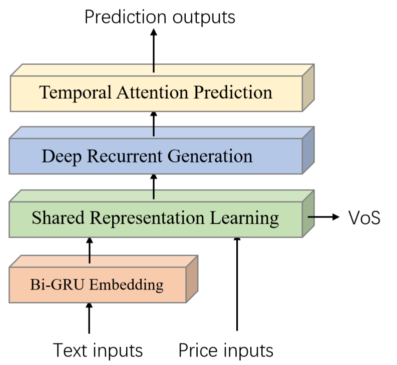
Prediction outputs
Temporal Attention Prediction
Deep Recurrent Generation
Shared Representation Learning
VoS
Bi-GRU Embedding
Text inputs
Price inputs

Figure 1: The overall framework of our model PEN.

Overall Architecture

We summarize the overall architecture of our proposed stock price Prediction-Explanation Network (PEN) in Figure 1. It comprises four main components: Text Embedding Layer (TEL) captures text information and obtains lower-dimensional representations; Shared Representation Learning (SRL) models the interaction between text data and stock price data and produces Vector of Salient (VoS) that indicates the importance of each text; Deep Recurrent Generation (DRG) infers the latent variable Z and decodes y from Z and X; Temporal Attention Prediction (TAP) employs attention mechanism to produce the final prediction from multiple predictions at different time steps.

Text Embedding Layer

In order to capture the information from the past and future words as its context and obtain lower-dimensional representations, we leverage a bi-directional GRU layer. For a stock on tth trading day, the word embedding representation of texts is defined as $C_t = [C_{t1}, C_{t2}, \ldots, C_{tm}, \ldots, C_{tM}] \in R^{M\times l\times h_w}$, where M denotes the number of texts and C_{tm} denotes word embedding matrix for mth text with length l and hidden size h_w. We run the bi-directional GRU layer for every text to obtain text embeddings. Then, all texts in a day can be represent as the matrix $e_t = [e_{t1}, e_{t2}, \ldots, e_{tm}, \ldots, e_{tM}] \in R^{h\times M}$, $e_{tm} \in R^{h\times 1}$ is the embedding of mth text in tth trading day with hidden size h. The maximum number of tokens included in a text and the maximum number of texts on a single trading day are set to 30 and 20, respectively.

Shared Representation Learning

For price data in tth trading day of a stock, The price vector $\tilde{p}_t = [\tilde{p}_t^c, \tilde{p}_t^h, \tilde{p}_t^l] \in R^{L\times 3}$ consists of adjusted close price p_t^c, high price p_t^h and low price p_t^l. After normalization with $p_t = \tilde{p}_t/\tilde{p}_{t-1} - 1$, $p_t \in R^{L\times 3}$ is as input sent into SRL.

SRL ingests the text embeddings $e_t \in R^{h\times M}$ and price data $p_t \in R^{L\times 3}$ and learns their shared representation. Dif-

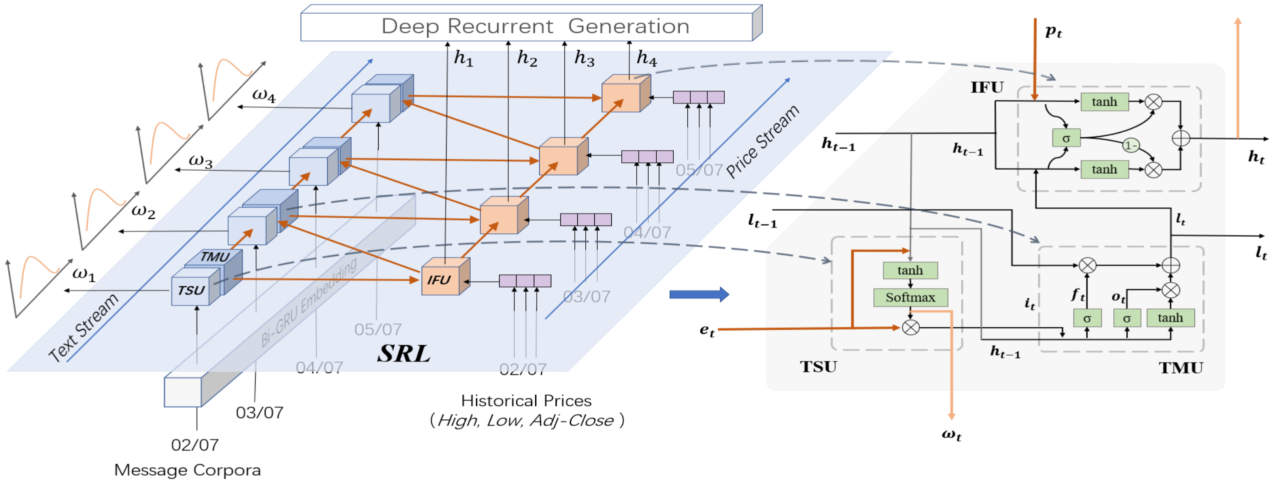

Figure 2: The detailed structure of SRL module. We use information, including texts and prices, of 02/07 - 04/07 to predict stock movement of 05/07 for illustration.

ferent SRL modules are connected at different time steps, resulting in a recurrent structure. Each SRL module consists of three units, Text Selection Unit (TSU), Text Memory Unit (TMU), Information Fusion Unit (IFU). The details of SRL are shown in Figure 2.

Text Selection Unit. Raw text data obtained is usually highly general, and is not step-wise synchronized with stock price time series. Furthermore, large variance in texts quality means some input texts may be totally irrelevant in predicting stock price movement, hence may bring more noise to our prediction objective. Here, we construct a Text Selection Unit (TSU) to select the useful text embeddings in a day and generate a Vector of Salience (VoS) $\omega_t \in R^{M\times 1}$, in which each single scalar represents the importance of individual text.

$$\begin{aligned} \omega_t &= \text{softmax}\left[k_t^T \tanh\left(W_1 h_{t-1} + W_2 e_t + b_1\right)\right] \\ i_t &= e_t \omega_t \end{aligned} \tag{2}$$

We generate initial hidden state $h_0 \in R^{h\times 1}$ by Xavier algorithm (Glorot and Bengio 2010) and h_{t-1} is the former hidden state. $i_t \in R^{h\times 1}$ is text embedding weighted vector. W_1, W_2, k_t are weight matrices and b_1 is bias.

Text Memory Unit. As it takes time for the market to digest news and announcements, the past information contained in texts is valuable and cannot simply be discarded. Taking this into consideration and inspired by the structure of a LSTM cell, we design a Text Memory Unit to preserve important information in text embeddings over time, with forget gate f_t and output gate o_t to regulate information flowing in and out of the cell.

$$\begin{aligned} f_t = \sigma\left(W_3[i_t,\ \mathrm{h}_{t-1}] + b_3\right) \quad o_t = \sigma\left(W_4[i_t,\ \mathrm{h}_{t-1}] + b_4\right) \\ l_t = \tanh\left(W_5[i_t,\ \mathrm{h}_{t-1}] + b_5\right) \quad l_t = f_t l_{t-1} + o_t l_t \end{aligned} \tag{3}$$

$l_t \in R^{h\times 1}$ is text memory state with hidden state h, and it is initialized by Xavier algorithm. W_3, W_4, W_5 are weight matrices and b_3, b_4, b_5 are biases.

Information Fusion Unit. The key idea of the SRL module is to learn the shared representation for texts and stock prices then exploit the interaction patterns to identify the important texts. We introduce an Information Fusion Unit (IFU), which ingests text embeddings and stock prices, and fuses them together.

$$\begin{aligned} d_t = \sigma\left(W_6[p_t, l_t, h_{t-1}]\right) \quad h_l = \tanh\left(W_7[l_t, h_{t-1}]\right) \\ h_p = \tanh\left(W_8[p_t, h_{t-1}]\right) \quad h_t = d_t h_p + (1 - d_t) h_l \end{aligned} \tag{4}$$

where the hidden state $h_t \in R^{h\times 1}$, is the shared representation of text memory l_t and price information p_t. W_6, W_7, W_8 are weight matrices.

It is worth noting that the hidden state h_t of IFU is also an output of the SRL module, which is then used to next SRL module at the next time step $t+1$. We regard the last hidden state h_t, i.e. the hidden state of the last day in time lag, as input of deep recurrent generation module.

Deep Recurrent Generation

A variational auto-encoder provides a probabilistic manner for describing an observation in latent space. Inspired by the stocknet (Xu and Cohen 2018), we use a recurrent variational auto-encoder to generate stock price movements.

Given input $X = [x_1; \ldots; x_T]$ for every trading day $t \in [1,\ \ldots,\ T]$ where $x_t = h_t$ is the output of IFU. In variational auto-encoder, we need to construct latent variable $Z = [z_1; \ldots; z_T]$ and then predict stock movement $y = [y_1, \ldots, y_T]$. Formally, we have to model the conditional probability distribution $p_\theta(y \mid X)$ and its factorization is as follows

$$\begin{aligned} p_\theta(y \mid X) &= \int_Z p_\theta(y, Z \mid X) \\ &= \int_Z p_\theta\left(y_T \mid X, Z\right) p_\theta\left(z_T \mid z_{<T}, X\right) \\ &\quad \prod_{t=1}^{T-1} p_\theta\left(y_t \mid x_{\le t}, z_t\right) p_\theta\left(z_t \mid z_{<t}, x_{\le t}, y_t\right) \end{aligned} \tag{5}$$

As it is shown in above equation, we need to infer the intractable posterior distribution $p_\theta\left(Z \mid X, y\right)$. A common way to solve this problem is to generate a distribution $q_\phi\left(Z \mid X, y\right)$ by variational inference (Jordan et al. 1999) to approximate $p_\theta\left(Z \mid X, y\right)$ and then simulate $q_\phi\left(Z \mid X, y\right)$ by using reparameterization in neural network.

Here an alternative way is used to restrict the family of distributions $q_\phi\left(Z \mid X, y\right)$. Suppose Z can be partitioned into disjoint time classes, we then assume that the approximate distribution factorizes with respect to different time points as follows,

$$q_\phi(Z \mid X, y) = \prod_{t=1}^{T} q_\phi\left(z_t \mid z_{<t}, x_{\leq t}, y_t\right) \tag{6}$$

The likelihood of our target conditional probability distribution $p_\theta(y \mid X)$ can be decomposed into,

$$\begin{aligned} \log p_\theta(y \mid X) = \log \int_Z p_\theta(y, Z \mid X) dZ \\ + E_{q_\phi(Z \mid X, y)}\left[\log p_\theta(y \mid X, Z)\right] \\ - D_{\mathrm{KL}}\left[q_\phi(Z \mid X, y) \| p_\theta(Z \mid X)\right] \end{aligned} \tag{7}$$

Therefore, minimizing the Kullback-Leibler divergence between $p_\theta\left(Z \mid X, y\right)$ and its approximation $q_\phi\left(Z \mid X, y\right)$ equals maximizing the last two terms of Eq. (7). i.e. the variational recurrent lower bound as follows,

$$\begin{aligned} \mathcal{L}(\theta, \phi; X, y) = \sum_{t=1}^{T} E_{q_\phi\left(z_t \mid z_{<t}, x_{\leq t}, y_t\right)}\left\{\log p_\theta\left(y_t \mid x, z\right) - \right. \\ \left. D_{\mathrm{KL}}\left[q_\phi\left(z_t \mid z_{<t}, x_{\leq t}, y_t\right) \| p_\theta\left(z_t \mid z_{<t}, x_{\leq t}\right)\right]\right\} \leq \log p_\theta(y \mid X) \end{aligned} \tag{8}$$

We assume that when $t < T$, y_t is independent of $z_{\leq t}$ so that $p_\theta\left(y_t \,|\, x, z\right)$in the Eq. (8) equals to $p_\theta\left(y_t \,|\, x_{\leq t}, z_t\right)$ while $p_\theta\left(y_t \,|\, x, z\right)$ =$p_\theta\left(y_t \,|\, X, Z\right)$.

Recurrent Variational Encoder & Decoder. We use a Recurrently Variational Auto-encoder based (Li et al. 2017) framework to conduct variational inference and generation. In the encoder and decoder stage, GRU is employed as the basic recurrent model to extract information from input and decode from latent variable instead of Fully Connected Layer. We assume that both the prior and posterior of the latent variables are of Gaussian distribution. namely, $p_\theta\left(z_t \mid z_{<t}, x_{\leq t}\right) \sim \mathcal{N}\left(z_t; \mu_\theta, \sigma_\theta^2 I\right)$ and $q_\phi\left(z_t \mid z_{<t}, x_{\leq t}, y_t\right) \sim \mathcal{N}\left(z_t; \mu_\phi, \sigma_\phi^2 I\right)$.

For the posterior, denoting the hidden state of encoder GRU as $h_t^{enc} = \mathrm{GRU}\left(x_t, h_{t-1}^{enc}\right)$and its shared representation with z_t as $h_t^{z^\phi} = \tanh\left(W_z^\phi\left[z_{t-1}, x_t, h_t^{enc}, y_t\right] + b_z^\phi\right)$, so we can calculate μ_t^ϕ, $\log(\sigma_t^\phi)^2$ and reparameterize the posterior z_t by,

$$\begin{cases} \mu_t^\phi = W_{z,\mu_\phi}^\phi h_t^{z^\phi} + b_{\mu_\phi}^\phi \\ \log(\sigma_t^\phi)^2 = W_{z,\sigma_\phi}^\phi h_t^{z^\phi} + b_{\sigma_\phi}^\phi \end{cases} \Rightarrow z_t^{post} = \mu_t^\phi + \sigma_t^\phi \odot \epsilon \tag{9}$$

where $\epsilon \sim \mathcal{N}(0, I)$ is white Gaussian noise, $W_{z,\mu_\phi}^\phi, W_{z,\sigma_\phi}^\phi$ are weight matrices and $b_{\mu_\phi}^\phi, b_{\sigma_\phi}^\phi$ are bias vectors.

For the prior, the shared representation of z_t and h_t^{enc} is $h_t^{z^\theta} = \tanh\left(W_z^\theta\left[z_{t-1}, x_t, h_t^{enc}\right] + b_z^\theta\right)$, and $\mu_t^\theta, \log(\sigma_t^\theta)^2$ and the prior z_t are calculated by,

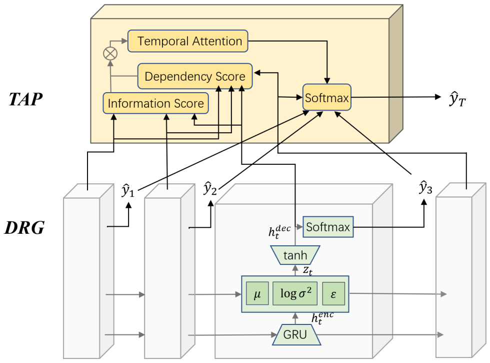

Figure 3: The detailed structure of DRG module and TAP module which work together to predict $\hat{y}_t$.

$$\begin{cases} \mu_t^\theta = W_{o,\mu_\theta}^\theta h_t^{z^\theta} + b_{\mu_\phi}^\theta \\ \log(\sigma_t^\theta)^2 = W_{o,\sigma_\theta}^\theta h_t^{z^\theta} + b_{\sigma_\theta}^\theta \end{cases} \Rightarrow z_t^{prior} = \mu_t^\theta \tag{10}$$

where $W_{o,\mu_\theta}^\theta, W_{o,\sigma_\theta}^\theta$ are weight matrices and $b_{\mu_\theta}^\theta, b_{\sigma_\theta}^\theta$ are bias vectors. Then we integrate the recurrent generative decoding component with the discriminative deterministic decoding component to predict stock price movement $\hat{y}_t$ by,

$$\begin{aligned} h_t^{dec} = \tanh\left(W_{dec}\left[x_t, h_t^{enc}, z_t^{prior}, z_t^{post}\right] + b_{dec}\right) \\ \hat{y}_t = \mathrm{softmax}\left(W_y h_t^{dec} + b_y\right), t < T \end{aligned} \tag{11}$$

where W_{dec}, W_y are weight matrices and b_{dec}, b_y are bias vectors.

Temporal Attention Prediction

To explore the relationship between prediction target $\hat{y}_T$ with its former information $[\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{T-1}]$, we adopt a temporal attention mechanism as shown in Figure 3. The decoder hidden state $H^{dec} = [h_1^{dec}, \ldots, h_{T-1}^{dec}]$ is used to calculate attention weight. The dependency score vector q^{dec}, information score vector k^{dec}, normalized attention weight vector w^{dec} and value vector v^{dec} are denoted as follows, respectively.

$$\begin{aligned} q^{dec} = (h_T^{dec})^\top \tanh\left(W_q H^{dec}\right) \\ k^{dec} = w_k^\top \tanh\left(W_k H^{dec}\right) \\ w^{dec} = \mathrm{softmax}\left(q^{dec} \odot (k^{dec})^\top\right) \\ v^{dec} = [\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_{T-1}] \end{aligned} \tag{12}$$

where W_q, W_k are weight matrices. q^{dec} represents the hidden state of Tth trading day, information score vector k^{dec} includes stock movements information of the past trading days and the weight vector w^{dec} measures the correlation between target day with the past trading days.

Finally, we have target prediction $\hat{y}_t$ as a temporal attention of hidden states as follows,

$$\hat{y}_T = \mathrm{softmax}(W_{dec}^T[v^{dec}(w^{dec})^\top, h_T^{dec}] + b_{dec}) \tag{13}$$

where W_{dec}^T is a weight matrix and b_{dec} is a bias vector.

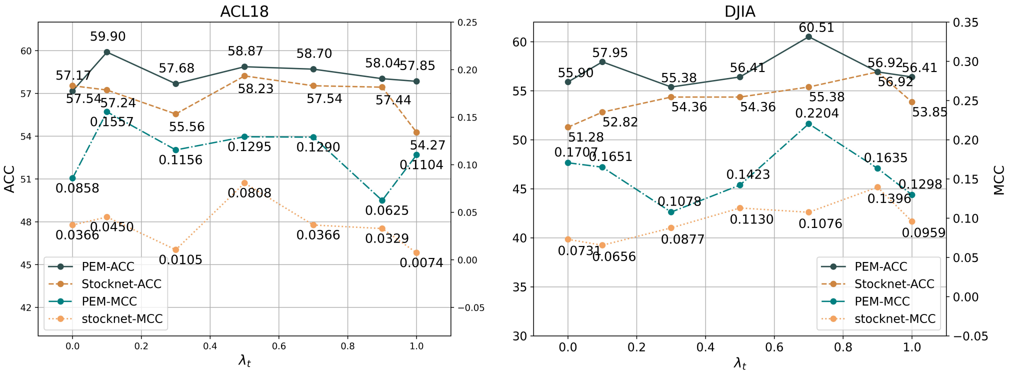

Figure 4: Performance comparison of PEN and Stocknet under different λ_t. The left chart shows the results on ACL18, and the right on DJIA. PEN performs significantly better than Stocknet over all λ_t across both datasets.

Learning Objective

As its name suggests, the learning objective of PEN consists of two parts: to maximize the prediction accuracy and to maximize the explainability of prediction results by selecting the most relevant input texts.

Based on Eq. (8), the objective L_t of tth day $t \in [1, \ldots, T]$ can be isolated as

$$\begin{aligned} L_t &= \log p_\theta \left(y_t \mid x_{\leq t}, z_{\leq t}\right) \\ &\quad - \lambda_r D_{\mathrm{KL}} \left[q_\phi \left(z_t \mid z_{<t}, x_{\leq t}, y_t\right) \| p_\theta \left(z_t \mid z_{<t}, x_{\leq t}\right)\right] \end{aligned} \tag{14}$$

where $\lambda_r \in (0, 1]$ is a decay weight from the KL term annealing trick. We then apply temporal attentions to L_t and average samples to formulate the first optimization objective:

$$\mathcal{L}_1(\theta, \phi; X, y) = \frac{1}{N} \sum_n^N w^{obj} L_t^{(n)} \tag{15}$$

where $w^{obj} = [\lambda_t w^{dec}, 1]$ is the global temporal weight vector, which can be adjusted by the hyper parameter λ_t.

For explainability, we design a salient regulator to strengthen SRL module's capability of information concentration. This can be achieved by maximize the Kullback-Leibler divergence between VoS ω_t and discrete uniform distribution. So formally, another optimization objective is as follows,

$$\mathcal{L}_2(\theta, \phi; X, y) = D_{\mathrm{KL}} \left[\omega_t \| p_u\right].$$

where $p_u \sim \mathcal{U}(M)$ is the discrete uniform distribution, and M is the number of texts in a sample.

Combining the two objectives with equal weight to avoid excessive parameter tuning, we have the overall objective:

$$\mathcal{L}(\theta, \phi; X, y) = \mathcal{L}_1(\theta, \phi; X, y) + \mathcal{L}_2(\theta, \phi; X, y) \tag{16}$$

Experiments

Experimental Setup

Datasets. We train and evaluate our model on two datasets: **ACL18** (Xu and Cohen 2018) and **Daily News for Stock Price Movement Prediction Dataset (DJAI)** [1]. We choose these two datasets because 1) they span two distinct time periods; 2) **ACL18** is for individual stocks while **DJIA** is for stock market indices; 3) they include two completely different types of text data: news articles and social media posts (tweets). **ACL18** includes the text data and historical price for 88 highly traded US stocks between 2014-01-01 and 2016-01-01 from 9 industries. Texts are tweets retrieved from Twitter and the historical prices data are collected from Yahoo Finance. We process **ACL18** in the same way as proposed in (Xu and Cohen 2018). **DJIA** includes news and price data on Dow Jones Industrial Average from 2008-06-08 to 2016-07-01, where news data is consisted of the top 25 headlines of Reddit WorldNews Channel every day.

Evaluation Metrics. We evaluate the accuracy of model results by two metrics: accuracy (ACC) and Matthews Correlation Coefficient (MCC)(Xu and Cohen 2018).

Parameters Setup. With our PEN model we use Tensorflow to construct the computational graph, initialize all bias zero, all weights with Xavier algorithm (Glorot and Bengio 2010), and optimize the final loss by Adam with learning rate of 1e-3. We use 32 shuffled samples in a batch and a 5-day lag window for model to learn historical context. We set the size of hidden state in SRL module and in word embedding to be 100 and 50, respectively. Besides, we use the input dropout rate of 0.4 to regularize latent variables. As the DJIA dataset is relatively small, we use the models trained on ACL18 as pre-trained models and then fine tune them on DJIA dataset for all baselines.

Baselines. We compare PEN with the following baselines.

•**Random**: randomly generated movement predictions.

•**Random Forest / RF** (Pagolu et al. 2016): a Random Forest classifier using sentiment analysis with Word2vec.

•**HAN** (Hu et al. 2018): a hybrid attention networks based on related news.

•**Stocknet** (Xu and Cohen 2018): a deep generative model jointly exploiting text and price signals.

•**CPC** (Wang et al. 2021): a copula-based contrastive predictive coding method which models relevant macroeconomic variables to improve prediction accuracy.

[1] https://www.kaggle.com/aaron7sun/stocknews.

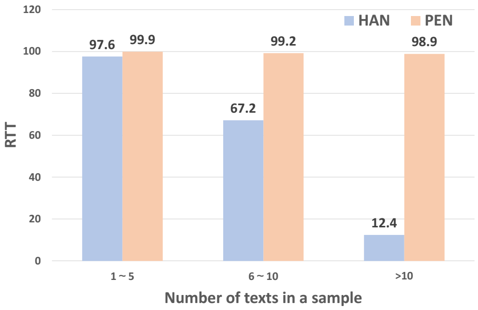

Figure 5: A further comparison on explainability: RTT metric varies with the number of texts in a single sample.

Results of Prediction Accuracy

First we make a comparison on ACC and MCC between PEN and Stocknet with a varied λ_t (see Eq. (15)) which controls the impact of past loss. As is shown in Figure 4, PEN outperforms Stocknet for almost every value of parameter λ_t, and achieves the best performance on ACL18 when $\lambda_t = 0.1$, and on DJIA when $\lambda_t = 0.7$. These two λ_t values are adopted in PEN to carry out the rest of experiments.

Models	ACL18		DJIA	
	ACC(%)	MCC	ACC(%)	MCC
Random	50.8900	-0.002266	50.2435	0.000360
RF	53.0800	0.012929	52.6455	0.050990
HAN	57.6400	0.051800	53.2258	0.060150
Stocknet	58.2300	0.080796	56.9231	0.136909
CPC	59.1100	**0.181700**	-	-
PEN	**59.8976**	0.155652	**60.5128**	**0.220423**

Table 1: Prediction results of baseline models and PEN. These results show that PEN is the best performing model on both datasets under the two accuracy criterion.

In Table 1, we report the prediction performance of the baselines and PEN. As is shown, PEN outperforms all baselines except CPC on accuracy as well as MCC across both datasets. In particular it outperforms Stocknet by (1.67% , 0.07) on ACL18 and (3.59%, 0.08) on DJIA in ACC and MCC respectively. PEN also outperforms CPC, which uses additional macroeconomic data in addition to ACL18 baseline dataset, on all metrics except MCC on ACL18 dataset.

Overall, the results in this section clearly demonstrate that the proposed Shared Representation Learning module and the VoS regulation mechanism can indeed enhance the performance of stock movement prediction.

Results of Explainability

We evaluate the explainability from two aspects:

•**Attention**: is the model able to focus on a small subset of text corpora for explanation?

•**Relevance**: is the small subset of text corpora identified indeed relevant for stock price movement prediction?

As our interviews with several event-driven hedge fund managers reveal they don't look beyond one or two pieces of news to make their investment decisions, we examine the former by calculating a Ratio of Top Two (RTT), which is the percentage of samples where the top two texts (with highest attention or VoS weights) account for over 95% weights over the total number of samples.

For the latter, we select a random 1,000 test examples from ACL18 dataset, generate their respective attention weights (from HAN) and VoS (from PEN), and invite three experienced investors to independently rate each news article/tweet in those samples. For each sample we ask them to pick a single text document they believe are most relevant in predicting the next price movement. Then we calculate a Ratio of Relevance (RoR) score which is the number of samples where the top-weighted text (by HAN or PEN) agrees with at least one of the three investors' top picks, over the total number of samples. We also conduct an agreement evaluation calculating the average Fleiss' kappa score between the top picks by the algorithm and those of three individual investors. The Fleiss' kappa score is defined as $\kappa = (\bar{P} - \bar{P}_e)/(1 - \bar{P}_e)$, where $1 - \bar{P}_e$ gives the degree of agreement that is attainable above chance, and $\bar{P} - \bar{P}_e$ gives the degree of agreement actually achieved above chance. The results are summarized in the Table 2.

Model	RTT (%)	RoR (%)	Kappa score
HAN	68.3	54.8	0.430
PEN	**99.5**	**89.3**	**0.591**

Table 2: A Comparison of HAN and PEN on Explainability Metrics RTT, RoR and Kappa.

It is shown that our SRL modules are able to focus on just one or two salient texts out of all available text corpora for 99.5% of the test samples, significantly more concentrated than standard attention weights. Moreover, among the samples for human inspection, 89.3% of PEN's top rated texts agree with at least one human investor with an average kappa score of 0.591. This further demonstrates the effectiveness of SRL module's selection mechanism.

We further compare the RTT ratios of PEN and HAN over samples with different number of text documents. The results, in Figure 5, show that a large text corpora doesn't seem to present any challenges to SRL, manifested by a consistent RTT despite increased size of texts.

As examples of qualitative evaluation, we present a couple of examples from ACL18 and DJIA datasets to illustrate how VoS generated by SRL compares with normal attention weights. Table 3 show the top texts picked out by PEN, the corresponding VoS of those texts, and attention weights from HAN. Note the columns "Ground Truth" (GT) indicates the number of individual investors who has the text as their top pick over the total number of human assessors (which is 3). The most relevant texts for *Apple* on 10/22/2015 are shown in the top half of Table 3. In this particular sample, SRL assigns a VoS weight of 0.994 to investors' top pick while the second ranked only receives a weight of 0.00386. The top picks by PEN are the same picks by investors. Similarly, the bottom half of Table 3 shows the

11/16/2015 Apple (\$APPL)			
HAN	PEN	GT	Tweets
1.13E-01	9.94E-01	3/3	analysts set apple target price at \$144.65 \$aapl
1.12E-01	3.86E-03	0/3	apple ring : newest patent application is a touch s creen enabled wearable for your finger \$aapl
1.09E-01	1.52E-03	0/3	apple : the new arms race \$aapl
1.06E-01	1.01E-04	0/3	supertrades could buy a freakin house with this recent win ! ! \$mbly \$aapl \$jcp \$tasr \$goog
1.01E-01	9.09E-05	0/3	\$aapl profitsnatcher: well played did not see this drop
04/06/2015 Dow Jones Industrial Average (DJIA)			
8.31E-02	7.88E-01	3/3	Russia Is On The Dawn of a Prolonged Recession as Oil Prices Stay Low and Sanctions Remain in Effect
8.59E-02	2.12E-01	0/3	Istanbul police kill woman carrying bomb near police HQ
6.64E-02	5.97E-05	0/3	Syria: Isis Destroys Tons of 'US-Made Halal Chicken' while Millions Go Hungry
7.67E-02	8.48E-06	0/3	Ban against a single blog post leads Turkish ISPs to censor all of WordPress
6.57E-02	8.41E-08	0/3	HSBC is 'cast-iron certain' to breach banking rules again, executive admits

Table 3: Qualitative comparison of VoS and attention weights on dataset ACL18 and DJIA. We show the top five texts picked by PEN at 10/22/2015 of Apple and at 04/06/2015 of DJIA.

news of Dow Jones on 04/26/2016 from DJIA dataset.

Whether it's the summary statistics in Table 2 or the samples in Tables 3, we observe a great degree of consistency between VoS weights and the ground truth. This clearly demonstrates that SRL modules are highly effective in identifying the important text contexts in predicting the future stock price movement.

Ablation Studies

Variations in PEN Architecture. To understand the contributions of different components in PEN, We conduct a number of ablation experiments where we remove TAP, DRG, SRL, KL-loss from PEN, respectively. The results, shown in Table 4, indicate that each and every part of PEN contributes to the overall model performance. Note when SRL is ablated, we use the normalized attention weights to handle text corpora as described in (Xu and Cohen 2018); this has lead to a reduction in ACC of 3.8% and 4.1% on ACL18 and DJIA, respectively, suggesting SRL modules are far superior to standard attention mechanism in measuring the relevance of text corpora for better prediction.

Model	ACL18		DJIA	
	ACC(%)	MCC	ACC(%)	MCC
w/o TAP	52.3891	0.027687	57.4359	0.0148
w/o DRG	51.7065	0.018908	56.9231	0.1587
w/o SRL	56.1433	0.073811	56.4103	0.1422
w/o KL	56.6553	0.058124	56.8462	0.0940
Full inputs	**59.8976**	**0.155652**	**60.5128**	**0.2204**

Table 4: An ablation study of architecture of PEN.

Stock Price Components. We also carry out a number of ablation experiments where we remove adjusted close prices, high prices and low prices from inputs to see how important they are. The results are summarized in Table 5. We observe that high and low prices are far more important than adjusted close price in this task. This seems to reaffirm the belief by many investors that volatility, reflected by the high and low prices, is a crucial characteristic of a stock. We also notice that full data enables PEN to perform the best, which suggests all types of price are valuable in prediction.

Inputs	ACL18		DJIA	
	ACC(%)	MCC	ACC(%)	MCC
w/o High	45.7338	0.0372	50.3413	0.0086
w/o Low	50.3413	0.0086	50.7692	0.0721
w/o Adj-close	57.6792	0.0890	55.3846	0.1319
Full inputs	**59.8976**	**0.1556**	**60.5128**	**0.2204**

Table 5: An ablation study of inputs on ACL18 and DJIA.

Conclusions

As a step to address the challenges of explainability in stock price movement prediction, we propose a Prediction-Explanation Network (PEN). The core of PEN is the Shared Representation Learning (SRL) module, which models the interaction between the text data and stock price data and outputs a Vector of Salience to explain the importance of texts related to stock prices changes. Inspired by the critical thinking process of investors, we further introduce such a regulation mechanism that SRL module focuses the smallest number of texts possible to maximize the explainability. Our experiments demonstrate that not only SRL module is capable of identifying highly relevant texts that explain the future stock price movements, but also greatly enhances the accuracy of our stock price movement prediction model PEN, which establishes new state-of-art accuracy across two benchmark datasets covering two distinct markets.

Acknowledgments

This work was supported by National Natural Science Foundation of China (NSFC Grant No. 62122089 and No. 61876196), Beijing Outstanding Young Scientist Program NO. BJJWZYJH012019100020098, and Intelligent Social Governance Platform, Major Innovation & Planning Interdisciplinary Platform for the ”Double-First Class” Initiative, Renmin University of China. We also wish to acknowledge the support provided and contribution made by Public Policy and Decision-making Research Lab of RUC. Rui Yan is supported by Beijing Academy of Artificial Intelligence (BAAI) and CCF-Tencent Rhino-Bird Open Research Fund.

References

Bollen, J.; Mao, H.; and Zeng, X. 2011. Twitter mood predicts the stock market. *Journal of computational science*, 2(1): 1–8.

Carta, S. M.; Consoli, S.; Piras, L.; Podda, A. S.; and Recupero, D. R. 2021. Explainable Machine Learning Exploiting News and Domain-Specific Lexicon for Stock Market Forecasting. *IEEE Access*, 9: 30193–30205.

Dang, X.-H.; Shah, S. Y.; and Zerfos, P. 2019. ” The Squawk Bot”: Joint Learning of Time Series and Text Data Modalities for Automated Financial Information Filtering. *arXiv preprint arXiv:1912.10858*.

Edwards, R. D.; Magee, J.; and Bassetti, W. C. 2018. *Technical analysis of stock trends*. CRC press.

Frankel, J. A.; and Frankel, B. 1995. *Financial markets and monetary policy*. MIT Press.

Glorot, X.; and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249–256. JMLR Workshop and Conference Proceedings.

Hu, Z.; Liu, W.; Bian, J.; Liu, X.; and Liu, T.-Y. 2018. Listening to chaotic whispers: A deep learning framework for news-oriented stock trend prediction. In *Proceedings of the eleventh ACM international conference on web search and data mining*, 261–269.

Jiang, W. 2021. Applications of deep learning in stock market prediction: recent progress. *Expert Systems with Applications*, 184: 115537.

Jordan, M. I.; Ghahramani, Z.; Jaakkola, T. S.; and Saul, L. K. 1999. An introduction to variational methods for graphical models. *Machine learning*, 37(2): 183–233.

Khadjeh Nassirtoussi, A.; Aghabozorgi, S.; Ying Wah, T.; and Ngo, D. C. L. 2015. Text mining of news-headlines for FOREX market prediction: A Multi-layer Dimension Reduction Algorithm with semantics and sentiment. *Expert Systems with Applications*, 42(1): 306–324.

Li, P.; Lam, W.; Bing, L.; and Wang, Z. 2017. Deep recurrent generative decoder for abstractive text summarization. *arXiv preprint arXiv:1708.00625*.

Li, X.; Xie, H.; Chen, L.; Wang, J.; and Deng, X. 2014. News impact on stock price return via sentiment analysis. *Knowledge-Based Systems*, 69: 14–23.

Pagolu, V. S.; Reddy, K. N.; Panda, G.; and Majhi, B. 2016. Sentiment analysis of Twitter data for predicting stock market movements. In *2016 international conference on signal processing, communication, power and embedded system (SCOPES)*, 1345–1350. IEEE.

Schumaker, R. P.; and Chen, H. 2009. Textual analysis of stock market prediction using breaking financial news: The AZFin text system. *ACM Transactions on Information Systems (TOIS)*, 27(2): 1–19.

Si, J.; Mukherjee, A.; Liu, B.; Li, Q.; Li, H.; and Deng, X. 2013. Exploiting topic based twitter sentiment for stock prediction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 24–29.

Wang, G.; Cao, L.; Zhao, H.; Liu, Q.; and Chen, E. 2021. Coupling Macro-Sector-Micro Financial Indicators for Learning Stock Representations with Less Uncertainty. *AAAI21*, 1–9.

Xing, F. Z.; Cambria, E.; and Welsch, R. E. 2018. Natural language based financial forecasting: a survey. *Artificial Intelligence Review*, 50(1): 49–73.

Xu, Y.; and Cohen, S. B. 2018. Stock Movement Prediction from Tweets and Historical Prices. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1970–1979. Melbourne, Australia: Association for Computational Linguistics.

Yoo, J.; Soun, Y.; Park, Y.-c.; and Kang, U. 2021. Accurate Multivariate Stock Movement Prediction via Data-Axis Transformer with Multi-Level Contexts. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery &amp; Data Mining*, KDD ’21, 2037–2045. New York, NY, USA: Association for Computing Machinery. ISBN 9781450383325.