Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)
Special Track on AI in FinTech

"The Squawk Bot"*: Joint Learning Of Time Series And Text Data Modalities For Automated Financial Information Filtering

Xuan-Hong Dang , Syed Yousaf Shah and Petros Zerfos
IBM T.J. Watson Research, Yorktown Heights, NY 10598, USA
xuan-hong.dang@ibm.com, {syshah,pzerfos}@us.ibm.com

Abstract

Multimodal analysis that incorporates time series and textual corpora as input data sources is becoming a promising approach, especially in the financial industry. However, the main focus of such analysis has been on achieving high prediction accuracy rather than on understanding the association between the two data modalities. In this work, we address the important problem of automatically discovering a small set of top news articles associated with a given time series. Towards this goal, we propose a novel multi-modal neural model called `MSIN` that jointly learns both the numerical time series and the categorical text articles in order to unearth the correlation between them. Through multiple steps of data interrelation between the two data modalities, `MSIN` learns to focus on a small subset of text articles that best align with the current performance in the time series. This succinct set is timely discovered and presented as recommended documents for the given time series, offering `MSIN` as an automated information filtering system. We empirically evaluate its performance on discovering daily top relevant news articles collected from Thomson Reuters for two given stock time series, AAPL and GOOG, over a period of seven consecutive years. The experimental results demonstrate `MSIN` achieves up to 84.9% and 87.2% respectively in recalling the ground truth articles, superior to SOTA algorithms that rely on conventional attention mechanisms in deep learning.

1 Introduction

Current multimodal analysis that combines time series with text data often focuses on extracting features from a text corpus and incorporating them into a forecasting model to enhance its predictive power [Ardia *et al.*, 2019; Fraiberger *et al.*, 2018]. At the same time, the ability to use text as a means of explanation [Akita *et al.*, 2016; Weng *et al.*, 2017] for the time series performance is an equally important requirement, as it helps to understand the possible reasons behind the patterns observed in the series. In many emerging financial applications such as "quantamental investing" [Wigglesworth, 2018], given a time series of an asset's performance, one often asks to retrieve a small set of text articles that can provide insights into the time series' performance. Also, investors do not solely base their decisions on historical prices when trading an asset, yet their decisions are often made with a careful consideration of macro-/micro-economic events and the most relevant news timely collected from the markets.

However, with the increasing amount of available textual information nowadays [Schumaker *et al.*, 2012], the challenge of finding the most relevant text articles associated with a particular asset series becomes more acute. From Figure 1, given recent m historical values of Apple's stock and today's n news (text) stories collected from the public media as two inputs, "Steve Jobs threatened patent suit" would be identified as one of the most relevant news articles associated with Apple. Certainly, for different stock time series, the set of relevant associated news documents would be different. One can find this important problem in other applications as well, such as cloud monitoring time series in combination with client support tickets and business performance metrics (e.g., sales) associated with the effectiveness of marketing campaigns, etc.

It is also worth mentioning that although filtering relevant text documents can rely on keywords (learning-to-match paradigm), we postulate that accurately identifying keywords for a specific time series is not an easy task, since it requires extensive domain knowledge and expertise. Moreover, for applications involving thousands of time series (e.g. stock markets), it is highly labor-intensive and error-prone in determining accurate keywords associated with each of them. More critically, the nature of both time series and textual contents is time-varying, filtering information based on *fixed* keywords would potentially lose essential and timely information. Hence an *automated* and *data-driven* system is highly desirable and becomes more relevant. In this work, we address the challenging problem of discovering relevant text articles associated with a numerical time series through developing a novel, multi-modal, neural network that jointly learns both data modalities. The discovered articles are returned as a means of recommended documents for describing the current state of the time series. To the best of our knowledge, this is the first study that follows a deep learning approach to address this problem in

*The name of this work is inspired by the popular financial commentary show called "Squawk Box" on CNBC (Squawk Box: Business, Politics, Investors and Traders, https://www.cnbc.com/squawk-box-us , accessed April 28th, 2020)

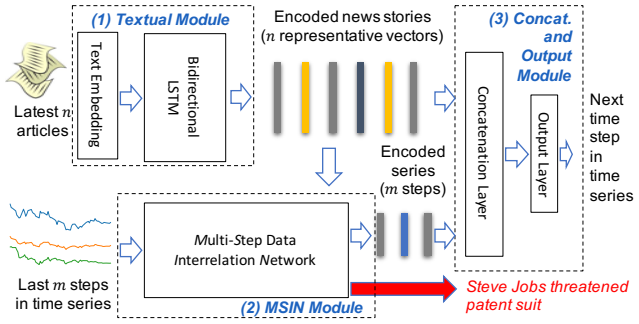

Figure 1: Illustration of our model that learns to discover relevant text articles (daily collected from a news source like Thomson Reuters) associated with the current state (characterized by the most recent m time steps/days) in a given time series.

the financial domain.

We propose a novel neural model called *Multi-Step Interrelation Network* or `MSIN` (shown in Fig.1), which allows the incorporation of semantic information learnt in the text modality to every time step in the modeling of the time series. We argue that *multiple data interrelations* are important and compelling in order to discover the association between the categorical text and numerical time series. This is because not only are they of different data types, but also no prior knowledge is given on guiding the neural model to look at which text clues are relevant for a given time series. Hence, compared to existing multimodal approaches that learn two data modalities either in parallel or sequentially, our proposed `MSIN` network effectively leverages the mutual information impact between them through multiple steps of data convolution. During such process, `MSIN` gradually assigns large attention weights to "important" text clues, while ruling out less relevant ones, and finally converges to an attention distribution over the text articles that best aligns with the current state of the time series. `MSIN` also technically relaxes the strict alignment at the timestamp level between the two data modalities, allowing it to concurrently deal with two input data sequences of different lengths, which currently is beyond the capability of most recurrent neural networks [Cho *et al.*, 2014; Chung *et al.*, 2014].

We perform empirical analysis over large-scale financial news and stock prices datasets spanning over seven years. The model was trained on data of the first six years and evaluated on the last year. We show that `MSIN` achieves strong performance of 84.9% & 87.2% in recalling ground-truth relevant news for two specific time series of Apple (AAPL) and Google (GOOG) stock prices respectively, exhibiting superior performance compared to state-of-the-art, deep learning models relying on conventional attention mechanisms.

2 Learning Text Articles Representation

Our overall model is illustrated in Fig.1 which consists of (1) a *Textual Module* that learns to represent each input text article as a numerical vector so that it is comparable to the time series, (2) the `MSIN` (*Multi-Step Interrelation Network*) that takes as input both sequences of time series and of textual vectors to learn the association between them and hence to discover relevant text documents, (3) a *Concatenation and Output* layer that aggregates information from the two data modalities to output the next value in the time series. This section presents our first *Textual Module*. We start modeling from the level of words as their order within each text document is important in learning semantic dependencies. To this end, our network exploits the long-short term memory (LSTM) [Hochreiter and Schmidhuber, 1997] to learn word dependencies and aggregates them into a single representative vector for each individual text article.

Specifically, at each timestamp t, an input sample to the textual module involves a sequence of n text documents $\{doc_1, \ldots, doc_n\}$ (e.g., n news articles released on day t-th). And it learns to output a sequence of n corresponding representative vectors $\{\mathbf{s}_1, \ldots, \mathbf{s}_n\}$ (t is omitted in $\mathbf{s}_j$'s and doc_j's for simplicity). Each doc_j in turn is a sequence of words represented as $doc_j = \{\mathbf{x}_1^{txt}, \ldots, \mathbf{x}_K^{txt}\}$ (superscript txt denotes text modality). Each $\mathbf{x}_\ell^{txt} \in \mathbb{R}^V$ is a one-hot-vector of the ℓ-th word in doc_j, with V as the vocabulary size. We use an embedding layer to transform each $\mathbf{x}_\ell^{txt}$ into a lower dimensional dense vector $\mathbf{e}_\ell \in \mathbb{R}^{d_w}$ via a transformation:

$$\mathbf{e}_\ell = \mathbf{W}_e * \mathbf{x}_\ell^{txt}, \quad \text{in which} \quad \mathbf{W}_e \in \mathbb{R}^{d_w \times V} \tag{1}$$

Often, $d_w \ll V$ and $\mathbf{W}_e$ can be trained from scratch; however, using or initializing it with pre-trained vectors from GloVe [Pennington *et al.*, 2014] can render more stable results. In our examined datasets (Sec.4), we found that setting $d_w = 50$ is sufficient given $V = 5000$. The sequence of embedded words $\{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_K\}$ for a document article is then fed into an LSTM that learns to produce their corresponding encoded contextual vector. The key components of an LSTM unit are the memory cell which preserves essential information of the input sequence through time, and the non-linear gating units that regulate the information flow in and out of the cell. At each step ℓ (corresponding to ℓ-th word) in the input sequence, LSTM takes in the embedding $\mathbf{e}_\ell$, its previous cell state $\mathbf{c}_{\ell-1}^{txt}$, and the previous output vector $\mathbf{h}_{\ell-1}^{txt}$, to update the memory cell $\mathbf{c}_\ell^{txt}$, and subsequently outputs the hidden representation $\mathbf{h}_\ell^{txt}$ for $\mathbf{e}_\ell$. From this view, we can briefly represent LSTM as a *recurrent* function $\mathbf{f}$ as follows:

$$\mathbf{h}_\ell^{txt} = \mathbf{f}(\mathbf{e}_\ell, \mathbf{c}_{\ell-1}^{txt} \mathbf{h}_{\ell-1}^{txt}), \quad \text{for} \quad \ell = 1, \ldots, K \tag{2}$$

in which the memory cell $\mathbf{c}_\ell^{txt}$ is updated internally. Both $\mathbf{c}_\ell^{txt}, \mathbf{h}_\ell^{txt} \in \mathbb{R}^{d_h}$ with d_h is the number of hidden neurons. Our implementation of LSTM closely follows the one presented in [Zaremba *et al.*, 2014] with two extensions. First, to better exploit the semantic dependencies of ℓ-th word on its both preceding and following contexts, we implement two LSTMs respectively taking the sequence in the forward and backward directions (denoted by head arrows in Eq.(3)). This results in a bidirectional LSTM (*BiLSTM*):

$$\begin{aligned} &\overrightarrow{\mathbf{h}}_\ell^{txt} = \overrightarrow{\mathbf{f}}(\mathbf{e}_\ell, \mathbf{c}_{\ell-1}^{txt}, \overrightarrow{\mathbf{h}}_{\ell-1}^{txt}), \overleftarrow{\mathbf{h}}_\ell^{txt} = \overleftarrow{\mathbf{f}}(\mathbf{e}_\ell, \mathbf{c}_{\ell-1}^{txt}, \overrightarrow{\mathbf{h}}_{\ell-1}^{txt}) \\ &\mathbf{h}_\ell^{txt} = [\overrightarrow{\boldsymbol{h}}_\ell^{txt}, \overleftarrow{\mathbf{h}}_\ell^{txt}] \quad \text{for} \quad \ell = 1, \ldots, K \end{aligned} \tag{3}$$

The concatenated vector $\mathbf{h}_\ell^{txt}$ leverages the context surrounding the ℓ-th word and hence better characterizes its semantic as compared to the embedding vector $\mathbf{e}_\ell$ which ignores the local context in the input sequence. Second, we extend the

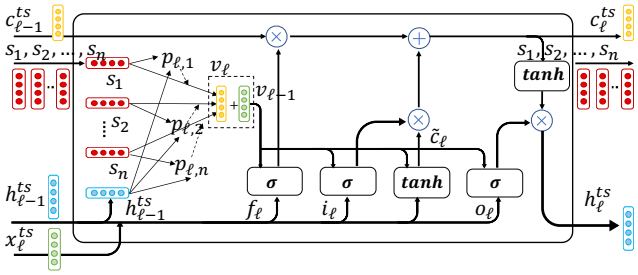

Figure 2: Novel memory cell of the recurrent multi-step data interrelation network (MSIN) with two input sequences from text ($\{\mathbf{s}_1, \ldots, \mathbf{s}_n\}$) and time series ($\{\mathbf{x}^{ts}_{t-m}, \ldots, \mathbf{x}^{ts}_{t-1}\}$) data modalities.

model by exploiting the weighted mean pooling from all vectors $\{\mathbf{h}^{txt}_1, \mathbf{h}^{txt}_2, \ldots, \mathbf{h}^{txt}_K\}$ to form the overall representation $\mathbf{s}_j$ of the entire j-th document doc_j:

$$\mathbf{s}_j = K^{-1} \sum\nolimits_{\ell} \beta_\ell * \mathbf{h}^{txt}_\ell \quad \text{for} \quad \ell = 1, \ldots, K$$

$$\text{where} \quad \beta_\ell = \frac{\exp(\mathbf{u}^\top \tanh(\mathbf{W}_\beta * \mathbf{h}^{txt}_\ell + \mathbf{b}_\ell))}{\sum_\ell \exp(\mathbf{u}^\top \tanh(\mathbf{W}_\beta * \mathbf{h}^{txt}_\ell + \mathbf{b}_\ell))} \tag{4}$$

in which $\mathbf{u} \in \mathbb{R}^{2d_h}$, $\mathbf{W}_\beta \in \mathbb{R}^{2d_h \times 2d_h}$ are respectively referred to as the parameterized context vector and matrix whose values are jointly learnt with the BiLSTM. This pooling technique resembles the successful one presented in [Conneau *et al.*, 2017] that learns multiple views over each input sequence. Our implementation yet simplifies them by adopting only a single view ($\mathbf{u}$ vector) with the assumption that each document (e.g., a news article) contains only one topic relevant to the time series. Note that, similar to CNNs [Kim, 2014], a max pooling can also be used in replacement for the mean pooling in defining $\mathbf{s}_j$. We, however, attempt to keep the module simple enough since the max function is non-smooth and generally requires more training data to learn. We apply our text module BiLSTM to every article collected at time period t and its output is a sequence of representative vectors $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$. Each $\mathbf{s}_j$ corresponds to one text document doc_j at input.

3 Multi-Step Interrelation Of Data Modalities

Our next task is to model the time series taking immediately into account the information learnt from the textual data modality in order to discover a small set of relevant text articles well aligning with the current state of the time series. An approach of using a *single* alignment between the two data modalities (empirically evaluated in Section 4) generally does not work effectively since the text articles are highly general, contain noise, and especially are not step-wise synchronized with the time series. The number of look-back time points m in the time series can be much different from the number of text documents n collected at the time moment, i.e., n varies from time to time. To tackle these challenges, we propose the novel `MSIN` network that broadens the scope of recurrent neural networks so that it can handle concurrently *two* input sequences of different lengths. More profoundly, we develop in `MSIN` a neural mechanism that integrates information captured in the representative textual vectors learnt in the previous textual module to every step reasoning in the time series sequence. Doing so allows `MSIN` leverage the mutual information between the two data modalities through multi-steps of data interrelation. Consequently, it gradually filters out irrelevant text articles while focuses only on those that correlate with the current patterns learnt from the time series, as it advances in the time series sequence. The chosen text articles are hence captured in the form of a probability mass attended on the input sequence of textual representative vectors.

In particular, inputs to `MSIN` at each timestamp t are two sequences: (1) a sequence of last m time steps in the time series modality $\{\mathbf{x}^{ts}_{t-m}, \mathbf{x}^{ts}_{t-m-1}, \ldots, \mathbf{x}^{ts}_{t-1}\}$ (superscript ts denotes the time series modality)[1]; (2) a sequence of n text representative vectors learnt by the above textual module $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_n\}$. Its outputs are the set of hidden state vectors (described below) and the probability mass vector p_m outputted at the last state of the series sequence. The number of entries in p_m equal to the number of text vectors at input. Its values are non-negative encoding which text documents relevant to the time series sequence. A large value at j-th entry of p_m reveals a high relevance of doc_j to the time series.

The memory cell of our `MSIN` is illustrated in Fig.2. As observed, we *augment* the information flow within the cell by the information learnt in the text modality. To be concrete, `MSIN` starts with the initialization of the initial cell state $\mathbf{c}^{ts}_0$ and hidden state $\mathbf{h}^{ts}_0$ by using two separate single-layer neural networks applied on the average state of the text sequence:

$$\mathbf{c}^{ts}_0 = \tanh(\mathbf{U}_{c_0} * \bar{\mathbf{s}} + \mathbf{b}_{c_0}) \tag{5}$$

$$\mathbf{h}^{ts}_0 = \tanh(\mathbf{U}_{h_0} * \bar{\mathbf{s}} + \mathbf{b}_{h_0}) \tag{6}$$

where $\bar{\mathbf{s}} = 1/n \sum_j \mathbf{s}_j$; and $\mathbf{U}_{c_0}, \mathbf{U}_{h_0} \in \mathbb{R}^{2d_h \times d_s}$, $\mathbf{b}_{c_0}, \mathbf{b}_{h_0} \in \mathbb{R}^{d_s}$, with d_s as the number of neural units. These are parameters jointly trained with our entire model.

`MSIN` then incorporates information learnt from the text articles into every step it performs modelling on the time series in a selective manner. Specifically, at each timestep ℓ in the time series sequence[2], `MSIN` searches through the text representative vectors to assign higher probability mass to those that better align with the signal it discovers so far in the time series sequence, captured in the state vector $\mathbf{h}^{ts}_{\ell-1}$. In particular, the attention mass associated with each text representative vector $\mathbf{s}_j$ is computed at ℓ-th timestep as follows:

$$a_{\ell,j} = \tanh(\mathbf{W}_a * \mathbf{h}^{ts}_{\ell-1} + \mathbf{U}_a * \mathbf{s}_j + \mathbf{b}_a) \tag{7}$$

$$p_\ell = softmax(\mathbf{v}_a^T [a_{\ell,1}, a_{\ell,2}, \ldots, a_{\ell,n}]) \tag{8}$$

where $\mathbf{W}_a \in \mathbb{R}^{d_s \times d_s}$, $\mathbf{U}_a \in \mathbb{R}^{2d_h \times d_s}$, $\mathbf{b}_a \in \mathbb{R}^{d_s}$ and $\mathbf{v}_a \in \mathbb{R}^n$. The parametric vector $\mathbf{v}_a$ is learnt to transform each alignment vector $a_{\ell,j}$ to a scalar and hence, by passing through the softmax function, p_ℓ is the probability mass distribution over the text representative sequence. We would like the information from these vectors, scaled proportionally by their probability mass, to immediately impact the learning process over the time series. This is made possible through generating a context vector $\mathbf{v}_\ell$:

$$\mathbf{v}_\ell = \frac{1}{2}(\sum_j p_{\ell,j} * \mathbf{s}_j + \mathbf{v}_{\ell-1}) \tag{9}$$

[1] We denote $\mathbf{x}^{ts}$ as vectors since an input time series can be multivariate in general (as illustrated in Fig. 1).

[2] We re-use ℓ to index timestep in a time series sequence with similar meaning in Eq.(2)-(4). Yet, note here that: $\ell = 1, \ldots, m$.

in which $\mathbf{v}_0$ is initialized as a zero vector. As observed, MSIN constructs the latest context vector $\mathbf{v}_\ell$ as the average information between the current representation of relevant text article (1st term on the RHS of Eq.(9)) and the previous context vector $\mathbf{v}_{\ell-1}$. By induction, influence of context vectors in the early time steps is fading out as MSIN advances in the time series sequence. MSIN uses this aggregated vector to regulate the information flow to all its input, forget and output gates:

$$\mathbf{i}_\ell = \sigma(\mathbf{U}_{ix} * \mathbf{x}_\ell^{ts} + \mathbf{U}_{ih} * \mathbf{h}_{\ell-1}^{ts} + \mathbf{U}_{iv} * \mathbf{v}_\ell + \mathbf{b}_i) \quad (10)$$

$$\mathbf{f}_\ell = \sigma(\mathbf{U}_{fx} * \mathbf{x}_\ell^{ts} + \mathbf{U}_{fh} * \mathbf{h}_{\ell-1}^{ts} + \mathbf{U}_{fv} * \mathbf{v}_\ell + \mathbf{b}_f) \quad (11)$$

$$\mathbf{o}_\ell = \sigma(\mathbf{U}_{ox} * \mathbf{x}_\ell^{ts} + \mathbf{U}_{oh} * \mathbf{h}_{\ell-1}^{ts} + \mathbf{U}_{ov} * \mathbf{v}_\ell + \mathbf{b}_o) \quad (12)$$

and the candidate cell state:

$$\tilde{\mathbf{c}}_\ell^{ts} = \tanh(\mathbf{U}_{cx} * \mathbf{x}_\ell^{ts} + \mathbf{U}_{ch} * \mathbf{h}_{\ell-1}^{ts} + \mathbf{U}_{cv} * \mathbf{v}_\ell + \mathbf{b}_c)$$

where $\mathbf{U}_{\bullet x} \in \mathbb{R}^{D \times d_s}, \mathbf{U}_{\bullet h} \in \mathbb{R}^{2d_h \times d_s}, \mathbf{U}_{\bullet v} \in \mathbb{R}^{n \times d_s}$ and $\mathbf{b}_\bullet \in \mathbb{R}^{d_s}$. Let $\odot$ denote the Hadamard product, the current cell and hidden states are then updated in the following order:

$$\mathbf{c}_\ell^{ts} = \mathbf{f}_\ell \odot \mathbf{c}_{\ell-1}^{ts} + \mathbf{i}_\ell \odot \tilde{\mathbf{c}}^{ts}, \quad \mathbf{h}_\ell^{ts} = \mathbf{o}_\ell^{ts} \odot \tanh(\mathbf{c}_\ell^{ts})$$

By tightly integrating the information learnt in the textual modality to every step in modeling the time series, our network distributes burden of work in discovering relevant text articles throughout the course of the time series sequence. The selected relevant documents are also immediately exploited to better learn behaviors in the time series.

Concatenation and Output layer: Given representative vectors learnt from the text domain and the hidden state vectors from the time series, we use a concatenation layer to aggregate and pass them through an output dense layer. The entire model is trained with the output as the next value in the time series. As ablation study, we consider a variant of our model in which we exclude the multistep of interrelation between the two data modalities. A conventional LSTM is used to model the time series and subsequently align its last state with the representative textual vectors to find the information clues. This simplified model has fewer parameters yet the interaction between two data modalities is limited to only the last state of the time series sequence. We name it LSTMw/o abbreviated for LSTM without interaction.

4 Experiment

Datasets: We analyze dataset consisting of news headlines (text articles) daily collected from Thomson Reuters in seven consecutive years from 2006 to 2013 [Ding *et al.*, 2014], and daily stock-price series of companies collected from Yahoo! Finance for the same time period. We construct each data sample from two data modalities as follows: (i) stock values in the last $m = 5$ days (one week trading) to form a time series sequence; (ii) all n news headlines (ordered by their released time) in the last 24 hours to form a sequence of text documents. A trained model hence aims at discovering top relevant news articles associated with the time series in a daily basis[3].

[3]To identify "top" relevant articles, we set 0.5 as a threshold of accumulated probability mass from p_m, which is explained shortly.

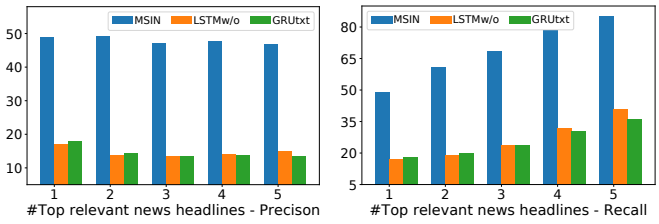

Figure 3: (a) Precision and (b) Recall computed w.r.t. ground truth headlines annotated by Reuters for "Apple" stock time series.

Methods: We evaluate our model with each of the two company-specific stock time series: (1) Apple (AAPL), and (2) Google (GOOG). Each dataset (text and time series) is split into training, validation and test sets respectively to the year-periods of 2006-2011, 2012, and 2013. We use the validation set to tune model's parameters, while utilize the independent test set for a fair evaluation. As mentioned at the beginning, neither time series identity nor pre-specified keywords have been used to train the model. For baselines, we implement LSTMw/o as described above, the GRUtxt deep learning network based on [Yang *et al.*, 2016]. In both models, their conventional attention mechanisms can reveal attentive documents. To emphasize the key contributions, we mainly discuss the performance of our model and these techniques on the important task of discovering relevant news articles associated to a given time series. For other predictive models, we further implemented LSTMpar [Akita *et al.*, 2016] (analyzing two data modalities in parallel), CNNtxt [Kim, 2014] (using text modality), GRUts (using time series modality), SVM [Weng *et al.*, 2017] (taking in both time series and unigrams of text news). A general observation is that models exploiting both data modalities tend to perform better than those utilizing a single data modality. For all models, we use 50-dim GloVe word embedding [Pennington *et al.*, 2014], while other parameters are tuned from: neural units $d_s, d_h \in \{16, 32, 64, 128\}$, regularization $L_1, L_2 \in \{0.1, \ldots, 0.001\}$, dropout rate $\in \{0.1, 0.2, 0.4\}$.

4.1 Relevant Text Articles Discovery

The Thomson Reuters corpus provides meta-data indicating whether a news article is reported about a specific company and we use such information as the ground truth relevant news, denoted by GTn. Just to clarify, such information is never used to train our model. Instead, we let MSIN learn itself the association between the textual news and the stock series via jointly analyzing the two data modalities concurrently and through multiple steps of data alignment. MSIN is hence *completely data-driven* and is straightforward to be applied to other corpus such as Bloomberg news source (or other applications like cloud business combined with client support tickets) where ground-truth articles are not available.

The set of GTn headlines allows us to compute the rate of discovering relevant news stories in association with a stock series through the precision and recall metrics. Higher ranking (based on attention mass vector p_m as in MSIN's Eq.(8)) of these GTn headlines on top of each day signifies better performance of an examined model. Fig.3(a-b) and Fig.4(a-b) plot

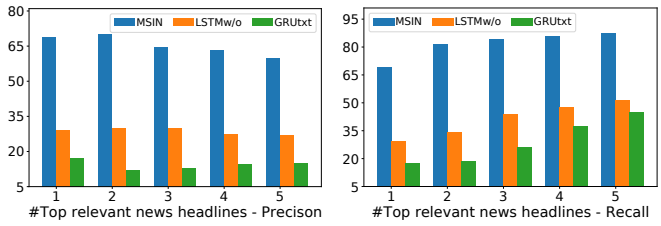

Figure 4: (a) Precision and (b) Recall computed w.r.t. ground truth headlines annotated by Reuters for "Google".

these concrete evaluations over the AAPL and GOOG time series respectively, when we vary the number of returned daily top relevant headlines k between 1 and 5 (shown in x-axis). For example, at $k = 5$, `MSIN` achieves up to 84.9% and 87.2% in recall while retains the precision at 46.8% and 59.6% respectively to the GTn sets of AAPL and GOOG. Other settings of k also show that `MSIN`'s performance is far better than those of two competitive models `GRUtxt` and `LSTMw/o`. Our novel neural technique of fusing two data modalities through multiple-step data interrelation allows `MSIN` to effectively discover the complex hidden correlation between the time-varying patterns in the time series and a small set of corresponding information clues in the categorical text data. Its precision and recall values significantly outperform those of `LSTMw/o` that utilizes only a single step of alignment between the two data modalities, and of `GRUtxt` which solely explores the text data with the attention mechanism in deep learning.

4.2 Explanation On Discovered Textual News

As concrete examples for qualitative evaluation, we show in Table 1 all news headlines from three specific examined days when our `MSIN` model is trained with the AAPL stock time series. We highlight the top relevant news headlines discovered by our `MSIN` model in green by setting the cumulative probability mass of attention $\geq 50\%$[4]. As clearly seen on date 2013-01-22, `MSIN` assigns the highest attention mass to *"steve jobs threatened patent suit to enforce no-hire policy"* though none of the words mentioned the Apple company. Likewise, on 2013-09-06, *"china unicom, telecom to sell latest iphone shortly after u.s. launch"* received the 2nd highest attention mass (21%), in addition to the 1st one *"apple hit with u.s. injunction in e-books antitrust case"* (37%). These news headlines obviously demonstrate the success of `MSIN` in discovering relevant news stories associated with the stock time series, as our model is capable of unearthing the news contents that never explicitly mention the specific company name. For comparison, we show the performance of two other competing methods, `GRUtxt` and `LSTMw/o` using single-step attention mechanism, in the 1st and 2nd columns of Table 1 (respectively annotated in the purple and orange colors). It is seen that the top relevant news articles outputted by `MSIN` is more specific and far better than those highlighted by `GRUtxt` and `LSTMw/o`, which consolidates our quantitative evaluation reported in Fig.3 and 4.

In Table 2, we report the results of all three models when

[4]That means the cumulative summation from largest to smallest on the MSIN's column (p_m) in Table 1 is greater than 0.5.

GRUtxt	LSTM$_{wo}$	MSIN	News headlines
			Date: 2013-01-22
0.06	0.00	0.00	01. analysis no respite euro zone long rebalancing slog
0.12	0.12	0.15	02. japan government welcome boj ease step towards percent inflation
0.06	0.04	0.03	03. japan government panel need achieve budget surplus
0.07	0.10	0.21	04. instant view existing home sale fall december
0.14	0.12	0.09	05. german exporter fear devaluation round boj move
0.07	0.05	0.03	06. bank japan yet revive economy
0.07	0.06	0.04	07. home resale fall housing recovery still track
0.08	0.12	0.02	08. instant view google put up well than expect quarterly number
0.08	0.11	0.03	09. bank japan buy asset sp set new five year high
0.12	0.06	0.01	10. google fourth quarter result shine ad rate decline slows
0.12	0.03	0.00	11. banks commodity stock lift sp five year high
0.07	0.02	0.01	12. google fourth quarter result shine ad rate decline slows
0.07	0.14	0.40	13. steve jobs threaten patent suit no hire policy filing
			Date: 2013-08-14
0.11	0.03	0.00	01. france exit recession beat second quarter gdp forecast
0.10	0.14	0.00	02. euro zone performance suggests recovery sight european rehn
0.14	0.03	0.00	03. germany france haul euro zone recession
0.11	0.07	0.00	04. yellen see likely next fed chair despite summers chatter reuters poll
0.11	0.12	0.00	05. us modest recovery fed cut back qe next month reuters poll
0.04	0.06	0.03	06. j.c. penney share spike report sale improve august
0.10	0.06	0.00	07. wallstreet end down fed uncertainty data boost europe
0.06	0.02	0.02	08. analysis balloon google experiment web access
0.05	0.11	0.01	09. wallstreet fall uncertainty fed bond buying
0.06	0.09	0.76	10. apple face possible may trial e book damage
0.04	0.03	0.17	11. japan government spokesman no pm abe corporate tax cut
			Date: 2013-09-06
0.03	0.04	0.21	01. china unicom telecom sell late iphone shortly us launch
0.03	0.05	0.01	02. spain industrial output fall month july
0.05	0.05	0.00	03. french consumer confidence trade point improve outlook
0.08	0.02	0.00	04. uk industrial output flat july trade deficit widens sharply
0.04	0.02	0.00	05. boj kuroda room policy response tax hike hurt economy minute
0.05	0.12	0.05	06. wind down market street funding amid regulatory pressure
0.09	0.03	0.01	07. china able cope fed policy taper central bank head
0.04	0.05	0.02	08. instant view us august nonfarm payroll rise
0.05	0.04	0.04	09. analysis fed shift syria crisis trading strategy
0.03	0.04	0.01	10. factbox three thing learn us job report
0.05	0.04	0.09	11. g20 say economy recover but no end crisis yet
0.03	0.04	0.04	12. us regulator talk european energy price probe
0.04	0.04	0.01	13. wallstreet flat job data syria worry spur caution
0.04	0.03	0.00	14. job growth disappoints offer note fed
0.04	0.05	0.02	15. bond yield dollar fall us job data
0.05	0.05	0.37	16. apple hit us injunction e books antitrust case
0.06	0.12	0.01	17. wallstreet week ahead markets could turn choppy fed syria risk mount
0.04	0.04	0.01	18. china buy giant kazakh oilfield billion
0.05	0.04	0.01	19. italy won't block foreign takeover economy minister
0.05	0.03	0.00	20. china august export beat forecast point stabilization
0.12	0.06	0.01	21. wallstreet week ahead markets could turn choppy fed syria risk mount
0.04	0.03	0.02	22. india inc policymakers shape up ship
0.02	0.02	0.04	23. cost lack indonesia economy
0.03	0.02	0.01	24. mexico proposes new tax regime pemex

Table 1: News headlines from 3 examined days in 2013 (test set). Relevant ones to AAPL stock series discovered by MSIN are highlighted in blue by choosing accumulated probability mass $\geq 50\%$. Attention vectors in GRUtxt and LSTMw/o are shown in the 1st and 2nd columns respectively.

they are trained with the GOOG stock time series (input of the text corpus remains the same). One can observe similar results in which the quality of top news articles highlighted by our `MSIN` outperforms those discovered by `GRUtxt` and `LSTMw/o` using conventional attention mechanisms. Note that news articles on date 2013-08-14 are deliberately shown in both Tables 1 and 2 to demonstrate the reality that the set of relevant news stories are dependent on which time series has been used

GRUtxt	LSTM$_{wo}$	MSIN	News headlines
			Date: 2013-01-09
0.23	0.34	0.00	01. short sellers circle stock confidence waver
0.26	0.52	0.87	02. google drop key patent claim microsoft
0.21	0.10	0.00	03. alcoa result lift share dollar up vs. yen
0.10	0.03	0.00	04. wallstreet rise alcoa report earnings
			Date: 2013-08-14
0.11	0.10	0.00	01. france exit recession beat second quarter gdp forecast
0.10	0.05	0.00	02. euro zone performance suggests recovery sight european rehn
0.11	0.07	0.00	03. germany france haul euro zone recession
0.10	0.11	0.00	04. yellen see likely next fed chair despite summers chatter reuters poll
0.12	0.08	0.00	05. us modest recovery fed cut back qe next month reuters poll
0.05	0.04	0.02	06. j.c. penney share spike report sale improve august
0.10	0.05	0.00	07. wallstreet end down fed uncertainty data boost europe
0.08	0.18	0.79	08. analysis balloon google experiment web access
0.09	0.08	0.00	09. wallstreet fall uncertainty fed bond buying
0.06	0.16	0.05	10. apple face possible may trial e book damage
0.04	0.08	0.02	11. japan government spokesman no pm abe corporate tax cut
			Date: 2013-08-29
0.08	0.09	0.01	01. india pm likely make statement economy friday
0.07	0.11	0.01	02. boj warns emerge market may see outflow
0.06	0.09	0.03	03. european rehn say lender step up assessment greece next month
0.03	0.06	0.20	04. china environment min suspends approval cnpc
0.05	0.12	0.08	05. italy still meet target scrap property tax say rehn
0.04	0.05	0.01	06. spain economic slump longer than thought but ease
0.06	0.04	0.02	07. india central bank consider gold trade minister
0.06	0.02	0.02	08. rupee fall front slow indian economy
0.05	0.02	0.02	09. india rupee bounce record low pm may address economy
0.04	0.05	0.01	10. exclusive india might buy gold ease rupee crisis
0.06	0.05	0.02	11. spain recession longer than thought but close end
0.05	0.05	0.01	12. india finance minister asks bank ensure credit flow industry
0.06	0.08	0.01	13. easing stimulus weigh oil next year reuters poll
0.03	0.05	0.01	14. exclusive india might buy gold ease rupee crisis
0.04	0.02	0.02	15. india rupee bounce record low government seek solution
0.03	0.02	0.37	16. china google power global drive
0.09	0.03	0.00	17. gdp growth beat forecast may boost case fed move
0.08	0.01	0.00	18. wallstreet rise economy but syria concern limit gain
0.06	0.01	0.03	19. oil dip syria action uncertain dollar rise data
0.02	0.00	0.01	20. boe carney say uncertainty rbs future end

Table 2: News headlines from 3 testing days in 2013. Relevant ones to GOOG stock series are highlighted in blue by MSIN.

to train the model. As observed, they are different between AAPL and GOOG stock time series, yet their relevance to each of the stock is intuitively interpretable. The automate discovery of these relevant news articles w.r.t. each time series is desirable, and `MSIN` performs comparably well on both.

5 Related Work

A large number of single-modality studies analyzing either time series data or unstructured text documents have been proposed in the literature. Some of these works are based on classical statistical methods [Wu *et al.*, 2016; Kristjanpoller and Michell, 2018] while others relying on neural networks [Qiu *et al.*, 2016; Zhong and Enke, 2017]. Recent studies from the financial domain explore both time series of asset prices and the news articles [Schumaker *et al.*, 2012; Weng *et al.*, 2017; Akita *et al.*, 2016] which are related to our work. Typically, these studies attempt to convert financial news into various numerical forms including news sentiment, subjective polarity, n-grams and then attach them as extra features of the numerical stock data. These handcrafted features require extensive pre-processing while being extracted independently from the time series data. [Akita *et al.*, 2016] relies on recurrent neural networks that enable it to model stock series in their sequential form, which are then merged with the vector representation of the text news prior to making the market prediction. The text news must be chosen manually to ensure their relevancy to the given stock series. This technique and the aforementioned ones hence lack the capability of automatically discovering news articles relevant to a given time series and are limited in explanation.

Our work is also related to multi-modal deep learning studies [Baltrušaitis *et al.*, 2019] which generally can be classified into three categories: early, late, and hybrid, depending on how and at which level the data from multiple modalities are fused together. In early fusion [Valada *et al.*, 2016; Zadeh *et al.*, 2016], multi-modal data sources are concatenated into a single feature vector prior to being used as the input to a learning model, while in late fusion, data are aggregated from the outputs of multiple models, each trained on a separate modality. The data can be fused based on aggregation rules such as, averaged-fusion [Nojavanasghari *et al.*, 2016], tensor products [Zadeh *et al.*, 2017], or a meta-model like gated memory [Zadeh *et al.*, 2018]. The hybrid (in-between) fusion is the trade-off between the early and late paradigms, that allows the data to be aggregated at different scales, yet often requiring the strict synchronization among involved data modalities, such as in the gesture recognition [Neverova *et al.*, 2015; Rajagopalan *et al.*, 2016]. Our model is related to the third category; yet, we deal with asynchronous multimodals of numerical time series and unstructured text and relax the constraints on the time-step synchronization between modalities. More significantly, we perform data fusion through multiple steps and at the low-level features which strengthens our model in learning associated patterns across data modalities.

6 Conclusion

Jointly learning both numerical time series and unstructured text data is an important research endeavor to enhance our understanding of time series performance. In this work, we presented a novel neural model that is capable of discovering the top relevant textual information associated with a time series. In dealing with the complexity of relationship between the two data modalities, we develop `MSIN` that allows the direct incorporation of information learnt in the text modality to every time step modeling on the behavior of time series, considerably leveraging their mutual association through time. Through multi-steps of data interrelation, our model can learn to focus on a small subset of textual documents that best align with the time series. `MSIN` is completely data-driven and can act as an automated news filtering system for time series. We evaluate the performance of our model in the financial domain using stock time series, which are trained along with the common news headlines collected from Thompson Reuters. The empirical results well demonstrate the capability of `MSIN` in discovering the association between the two data modalities by highlighting top news articles relevant to the given stock series. Its performance is superior to state-of-the-art models relying on the conventional attention mechanisms.

References

[Akita *et al.*, 2016] Ryo Akita, Akira Yoshihara, Takashi Matsubara, and Kuniaki Uehara. Deep learning for stock prediction using numerical and textual information. In *IEEE/ACIS*, 2016.

[Ardia *et al.*, 2019] Davir Ardia, Keven Bluteau, and Kris Boudt. Measuring the impact of financial news and social media on stock market modeling using time series mining techniques. *International Journal of Forecasting*, 2019.

[Baltrušaitis *et al.*, 2019] Tadas Baltrušaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2), 2019.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *EMNLP*, 2014.

[Chung *et al.*, 2014] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv:1412.3555*, 2014.

[Conneau *et al.*, 2017] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. *arXiv preprint arXiv:1705.02364*, 2017.

[Ding *et al.*, 2014] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Using structured events to predict stock price movement: An empirical investigation. In *EMNLP*, 2014.

[Fraiberger *et al.*, 2018] Samuel P. Fraiberger, Do Lee, Damien Puy, and Romain Ranciere. Media sentiment and international asset prices. *International Monetary Fund (IMF) Working Paper, Working Paper No. 18/274*, 2018.

[Hochreiter and Schmidhuber, 1997] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 1997.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[Kristjanpoller and Michell, 2018] Werner Kristjanpoller and Kevin Michell. A stock market risk forecasting model through integration of switching regime, anfis and garch techniques. *Applied Soft Computing*, 67:106–116, 2018.

[Neverova *et al.*, 2015] Natalia Neverova, Christian Wolf, Graham Taylor, and Florian Nebout. Moddrop: adaptive multi-modal gesture recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8), 2015.

[Nojavanasghari *et al.*, 2016] Behnaz Nojavanasghari, Deepak Gopinath, Jayanth Koushik, Tadas Baltrušaitis, and Louis-Philippe Morency. Deep multimodal fusion for persuasiveness prediction. In *ACM International Conference on Multimodal Interaction*, 2016.

[Pennington *et al.*, 2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[Qiu *et al.*, 2016] Mingyue Qiu, Yu Song, and Fumio Akagi. Application of artificial neural network for the prediction of stock market returns. *Chaos, Solitons & Fractals*, 85, 2016.

[Rajagopalan *et al.*, 2016] Shyam Sundar Rajagopalan, Louis-Philippe Morency, Tadas Baltrusaitis, and Roland Goecke. Extending long short-term memory for multi-view structured learning. In *ECCV*, 2016.

[Schumaker *et al.*, 2012] Robert P Schumaker, Yulei Zhang, Chun-Neng Huang, and Hsinchun Chen. Evaluating sentiment in financial news articles. *Decision Support Systems*, 53(3):458–464, 2012.

[Valada *et al.*, 2016] Abhinav Valada, Gabriel L Oliveira, Thomas Brox, and Wolfram Burgard. Deep multispectral semantic scene understanding of forested environments using multimodal fusion. In *International Symposium on Experimental Robotics*. Springer, 2016.

[Weng *et al.*, 2017] Bin Weng, Mohamed A Ahmed, and Fadel M Megahed. Stock market one-day ahead movement prediction using disparate data sources. *Expert Systems with Applications*, 79:153–163, 2017.

[Wigglesworth, 2018] Robin Wigglesworth. The rise of quantamental investing: Where man and machine meet. *Financial Times*, 2018.

[Wu *et al.*, 2016] Lifeng Wu, Sifeng Liu, and Yingjie Yang. Grey double exponential smoothing model and its application on pig price forecasting in china. *Applied Soft Computing*, 39, 2016.

[Yang *et al.*, 2016] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *NAACL-HLT*, 2016.

[Zadeh *et al.*, 2016] Amir Zadeh, Rowan Zellers, Eli Pincus, and Louis-Philippe Morency. Multimodal sentiment intensity analysis in videos: Facial gestures and verbal messages. *IEEE Intelligent Systems*, 31(6):82–88, 2016.

[Zadeh *et al.*, 2017] Amir Zadeh, Minghai Chen, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Tensor fusion network for multimodal sentiment analysis. *arXiv preprint arXiv:1707.07250*, 2017.

[Zadeh *et al.*, 2018] Amir Zadeh, Paul Pu Liang, Navonil Mazumder, Soujanya Poria, Erik Cambria, and Louis-Philippe Morency. Memory fusion network for multi-view sequential learning. In *AAAI*, 2018.

[Zaremba *et al.*, 2014] Wojciech Zaremba, Ilya Sutskever, and Oriol Vinyals. Recurrent neural network regularization. *arXiv preprint arXiv:1409.2329*, 2014.

[Zhong and Enke, 2017] Xiao Zhong and David Enke. Forecasting daily stock market return using dimensionality reduction. *Expert Systems with Applications*, 67:126–139, 2017.