Symposium on Computational Intelligence for Financial Engineering and Economics (IEEE), October 2024

Quantitative Market Situation Embeddings: Utilizing Doc2Vec Strategies for Stock Data

Frederic Voigt
University of the West of Scotland / Hamburg University of Applied Sciences
Hamburg, Germany
b01742821@studentmail.uws.ac.uk

Jose Alcarez Calero
University of the West of Scotland
Paisley, Scotland
jose.alcaraz-calero@uws.ac.uk

Keshav Dahal
University of the West of Scotland
Paisley, Scotland
keshav.dahal@uws.ac.uk

Qi Wang
University of the West of Scotland
Paisley, Scotland
qi.wang@uws.ac.uk

Kai von Luck
Hamburg University of Applied Sciences
Hamburg, Germany
kai.vonluck@haw-hamburg.de

Peer Stelldinger
Hamburg University of Applied Sciences
Hamburg, Germany
peer.stelldinger@haw-hamburg.de

***Abstract*—We introduce Quantitative Market Situation Embeddings (QMSEs), a pioneering artificial intelligence (AI)-driven methodology for encoding distinct temporal segments of stock markets into high-dimensional contextual embeddings exclusively leveraging quantitative stock data. Building upon prior research, we construe quantitative stock data analogously to Natural Language Processing (NLP) data, thereby adopting Doc2Vec methodologies to effectuate the embedding of stock data similar to document-level representations. We ascertain the efficacy of QMSEs in representing market dynamics by assessing their ability to discern various significant economic downturns post-2000, including but not limited to, the events of 9/11, the Subprime Crisis of 2008, and the Covid-induced market disruption. Moreover, we elucidate the practical utility of QMSEs through their application in employing distance metrics to gauge the rarity of market scenarios, serving as a regularizer in the training of quantitative stock AI models. Subsequently, we proceed to assess the algorithmic identification of analogous market conditions, aiming to elucidate their potential implications for future stock movements. Additionally, we demonstrate the efficacy of QMSEs in reducing data requirements for quantitative stock AI models by leveraging them as condensed representations of stock data.**

***Index Terms*—stock price prediction, stock movement prediction, quantitative analysis, stock embeddings**

I. Introduction

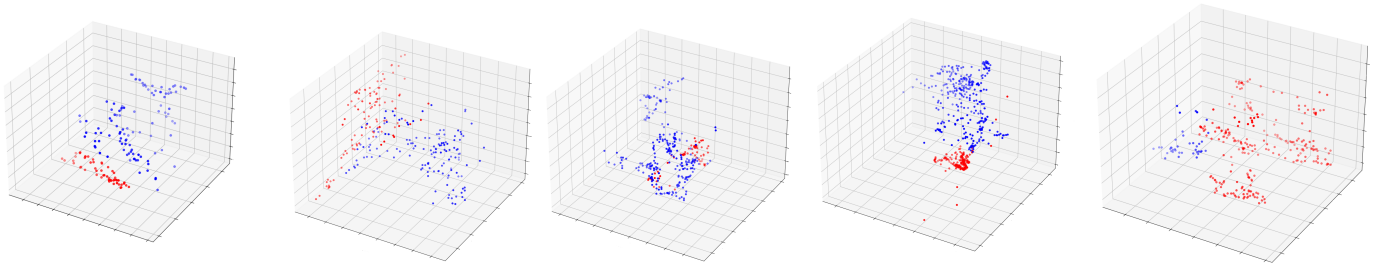
The endeavor to forecast stock prices, inclusive of both Stock Price Prediction (SPP) and Stock Movement Prediction (SMP), has emerged as a focal point within the domain of machine learning (ML) research. The pervasive notion of the inherent difficulty in these tasks [1] finds its roots, at least partially, in the non-stationary nature inherent to stock markets [25]. Efforts directed towards SPP/SMP typically fall into two methodological approaches, forming a dichotomy in their strategies. Quantitative analysis, as defined by Defusco et al. [32], aims to predict future price movements through historical market data analysis. In contrast, fundamental analysis relies on diverse data sources like annual reports, news, or analogous information streams to make predictions [38]. As established by Voigt et al. [36] quantitative analysis can be mathematically expressed as

$$P_\theta(X^{(t+1)}\big|\{X^{(t)},\dots,X^{(t-\Delta t)}\}) \tag{1}$$

for a ML model with parameters θ, with $X^{(t)}$ being the stock data at timestep t. Similarly, in the NLP domain, Unconditional Language Modeling (ULM) aims to ascertain the probability of a subsequent word token $w^{(l+1)}$ by considering all prior word tokens $\left\{w^{(i)}\right\}_{i=0}^{l}$. Building upon the observed parallels, Voigt el al. [36] examined the viability of implementing NLP methodologies for SPP/SMP.

In the realm of NLP, rather than employing ULM, the concept of Conditional Language Modeling (CLM) is often favored. CLM integrates an additional context Π, whereby the probability of the subsequent word token $w^{(l+1)}$ is articulated as $P_\theta(w^{(l+1)}|\Pi,\{w^{(l)},\dots,w^{(1)}\})$ [15]. This contextual information can take various forms, such as another text corpus [15] or a task description [39]. Fundamental stock analysis (if retaining consideration for historical stock data) can be defined as $P_\theta(X^{(t+1)}|\Pi,\{X^{(t)},\dots,X^{(t-\Delta t)}\})$, where Π encompasses fundamental data derived from sources such as those exemplified earlier. In the finance domain, studies such as [40] adhere to this, emphasizing the incorporation of contextual information. Moreover, there exists the capability to forecast price movements exclusively utilizing textual data, as demonstrated in [41], where the formulation simplifies to $P_\theta(X^{(t+1)}|\Pi)$.

One pertinent aspect for CLM in NLP involves specifying the document type as Π. Doc2Vec methodologies aim to encapsulate sentences, paragraphs, or entire documents (hereinafter referred to as the latter) as contextualized embeddings, mirroring the framework of Word2Vec. In Doc2Vec, akin to Word2Vec, the embedding vectors' similarities are intended to reflect the documents' similarities, positioning embedding vectors of similar documents near within the vector space. Consequently, these embedding vectors can be interpreted as representations of the document type. In the realm of



(a) 9/11, 2001 (b) Iraq War, 2003 (c) Subprime Crash, 2008 (d) Covid-19, 2020 (e) Ukraine War, 2022

Fig. 1: 3D-PCA visualization depicting the evolution of $\vec{e}$ across five pivotal events triggering substantial stock market fluctuations. Red markers indicate $\vec{e}$ representations pre-event, while blue markers signify post-event dynamics.

structuring text, documents serve as the conventional structure. However, when considering historical stock prices X as an equivalent to textual data [37], the inquiry emerges regarding the analogous construct to a document. Gabaix et al. [14] propose leveraging investors' portfolios, predominantly comprising funds and indices, as a framework for categorizing distinct companies c_i. While the creation of embeddings for c_i, as described in Section II, has been explored within the field of ML, we direct our attention towards an approach aligned with our conceptualization of interpreting stock trends akin to linguistic structures. Conceiving documents as a (finite) collection of textual data $\{w^{(i)}\}_{i=1}^{l<\infty}$, we define "Market Situations" as a fixed temporal window of size $\Delta\rho$ comprised of concatenated "Market Snapshots" [37], denoted as $X^{(t)} \in \mathbb{R}^{|C|\times 1}$. Each Market Snapshot encapsulates the pricing information of all companies $c_i \in C$ at a given time step t [37]. Drawing inspiration from Doc2Vec methodologies, we endeavor to construct contextualized embedding vectors for these "documents", thereby representing "Market Situations" to elucidate specific market conditions and facilitate comparative analyses. The embedding of market situations entails the generation of contextualized embeddings $\vec{e}$ for each timeframe, as illustrated in Figure 1.

We posit that the conceptualization of expressing "Situations" as distributed embeddings parallels the concept of Event Embedding models, exemplified by [7]. In these models, structured representations of events are derived from fundamental news data, facilitating SMP. In a broader sense, these event embeddings serve as embedded depictions of market situations, thereby constituting a foundational counterpart to our quantitative methodology.

In this work, we delineate several scenarios in the ensuing discourse wherein the Quantitative Market Situation Embeddings (QMSE) $\vec{e}$ can be deployed. Akin to the CLM, where the document type information can be incorporated into the model as Π, the QMSEs can be conveyed to a quantitative SPP/SMP model ($\Pi = \vec{e}$). Additionally, the inclusion of data pertaining to the current market conditions and its divergence from typical patterns can aid in stabilizing training. By being cognizant of exceptional circumstances, the model can adapt the learning process, as current data may not generalize well to conventional stock market dynamics. Given that $\vec{e}$ is designed to "summarize" the stock market over a time lag $\Delta\rho$, it follows that the volume of data required to represent extended stock time series may be diminished. This becomes particularly advantageous considering the increasing popularity of Transformer models [35] in quantitative SPP/SMP tasks. Notably, Transformer models exhibit quadratic time and space complexity [35], rendering them susceptible to challenges when handling lengthy input sequences, as evidenced in recent works such as [6] [37].

Analogous to event embedding models, which hypothesize that analogous events prompt comparable fluctuations and stock movements, one could infer that analogous market conditions result in akin stock movements. Consequently, time lags characterized by similar $\vec{e}$ representations are anticipated to exhibit parallel future stock movements. However, this notion contradicts the widely debated Random Walk Theory (RWT) [17] in economic discourse. According to this theory, stock market price variations are stochastic and unpredictable, thereby suggesting that past price movements lack reliability in forecasting future prices. Moreover, we propose future concepts for integrating QMSE techniques into (for SMP/SPP) Adapted Speech Models [37], as well as for applications in fraud detection, risk minimization / portfolio optimization, an "Emergency Switch" for automated training or flash crash detection.

Our study contributes by undertaking a comprehensive examination of various Doc2Vec inspired methodologies for generating QMSEs. We empirically illustrate the efficacy of these embeddings through their application to diverse historical stock market crashes (i.e. 9/11, the Iraq War, the Subprime Crash, the Covid-19 Crash and the Russia-Ukraine War). Additionally, we assess their utility across several applications, including their integration as supplementary inputs within quantitative SMP/SPP models, their role as regulatory mechanisms for training quantitative models, their application as data reduction techniques, and their utilization as a similarity-based approach for SMP.

We underscore that our primary objective is not centered on the development of superior-performing quantitative SMP/SPP models, but rather on the exploration of the efficacy of our conceptualizations concerning QMSEs within specific contexts.

Our aim is to evaluate the viability of these concepts within these exemplars, with the ultimate goal of their application to state-of-the-art (SOTA) SPP/SMP models. To achieve this, we conduct model training utilizing distinct baseline models on sixty-minute resolution quantitative intraday stock data encompassing 309 Standard & Poor's 500 (S&P-500) companies.

II. Related Work

In the ensuing discussion, we delve into related research categorized into the three relevant domains for the construction of the QMSEs: contextualized embeddings in finance, adaptable Doc2Vec methodologies, and quantitative SPP/SMP models as an orientation baseline.

a) Contextualized embeddings in Finance: Contextualized embeddings, such as GloVe [27], the BERT embeddings [5], or the Skip-Gram embeddings [23], represent prevalent strategies in NLP for embedding tokenized words into high-dimensional embedding spaces. The Word2Vec methodology employs either a Continuous Bag of Words (CBOW) approach or a Skip-Gram approach [23] to predict word tokens based on their surrounding words or vice versa, thereby training word embeddings. The resulting embeddings encode relationships of the word tokens based on their positions within the vector space. As discussed in Voigt et al. [37], the adaptation of this concept within the stock domain is actively addressed in contemporary research. Here it is customary to embed c_i entities to elucidate interrelationships among distinct companies. Such embeddings prove valuable in portfolio optimization and risk management strategies, facilitating investments in c_i entities characterized by minimal correlations [8]. In addition to the Stock2Vec models discussed by Voigt et al. [37], which directly adapt Word2Vec algorithms [37], various other c_i embedding methodologies are present in literature (also discussed in [37]). Dolphin et al. [8] endeavor to forecast companies with comparable returns ($x_i^{(t)} - x_i^{(t+1)}$), while Yoo et al. [42] incorporate embedding generation within their model pipeline. Sarmah et al.'s model [33] applies techniques inspired by Word2Vec strategies to sentence-like structures generated from stock correlations. In contrast, Gabaix et al.'s approach [14] employs portfolios of investors to facilitate the generation of embeddings for companies. The explicit embedding of entire market scenarios based on quantitative stock data to depict relationships across specific time intervals represents an approach, which, to the best of our knowledge, is entirely novel within the realm of quantitative data analysis.

b) Doc2Vec in NLP: Conceptually aligning $X^{(t)}$ with word tokens, as detailed in [37] and [36], and mapping the sequence $\left[X^{(t)}, X^{(t-1)}, \ldots, X^{(t-\Delta t)}\right]$ to documents, we draw inspiration for our embedding methodology from Doc2Vec models. Doc2Vec models, as expounded upon in Section I, serve within the NLP domain to encode entire documents into contextualized, high-dimensional vector representations. These representations are commonly employed in tasks like information retrieval, document classification, clustering, text generation and for recommender systems.

One of the pioneering models addressing the embedding of extended textual units, ranging from paragraphs to entire documents, in contrast to word embedding methodologies, was the Paragraph Vectors model proposed by Le and Mikolov [20]. The Skip-Thought model extends the Skip-Gram approach to the sentence level [19]. The advent of Transformer-based architectures in NLP has led to the emergence of models like Sentence-BERT, introduced by Reimers and Gurevych [31], which leverage large language models (LLMs) for the purpose of document embedding akin to Doc2Vec. Furthermore, there exist more lightweight methodologies utilizing Autoencoders, such as the one proposed by Bowman et al. [2], employing Long Short-Term Memory (LSTM) [16] networks for both encoding and decoding, aimed at reconstructing input sentences from latent representations.

c) Quantitative Stock Prediction: As previously acknowledged, there exists a notable preference for fundamental models over quantitative models within the domain of ML-based SPP/SMP, with the latter often struggling to attain comparable performance levels. We shall abbreviate this discussion, as our primary objective, as articulated in the introduction, does not entail surpassing SOTA models in SSP/SMP. This divergence in performance can be partially ascribed to theories such as the RWT and the Efficient Market Hypothesis (EMH) [12], which posit that asset prices encapsulate all available information. Consequently, some scholars advocate prioritizing fundamental data in light of these theories [22]. Noteworthy quantitative models have been previously delineated in [26] [36] [37]. However, for the sake of clarity and context within this discourse, we shall briefly enumerate a select few of these models.

While SSP remains less popular than SMP in the literature, efforts have been made to develop models tailored to this area. For instance, Qin et al. [28] proposed a model that achieved an root-mean-square error (RMSE) of 0.31 on NASDAQ intraday data at one-minute resolution, surpassing their baseline established using Recurrent Neural Networks (RNNs) [10], which achieved only a 0.96 RMSE. Feng et al. [13] integrate external data pertaining to industries and sectors. Except for this, they embrace a quantitative methodology, yielding outcomes, with reported RMSE values of 0.015 for NYSE data and 0.019 for interday NASDAQ data. Additionally, the LSTM baseline model also achieves an RMSE of 0.019 on NASDAQ and 0.015 on NYSE, demonstrating a performance nearly indistinguishable from that of the main model.

In the realm of SMP, notable quantitative models have emerged. For instance, Ding et al. [6] developed a Transformer-based model that achieved an accuracy of 57.3% on interday data sourced from NASDAQ stocks. Remarkably, the same model attained a higher accuracy of 58.7% when applied to intraday China A-shares data at a 15-minute resolution. In comparison, the baseline LSTM model achieved accuracies of 56.7% on the China A-shares data and 53.89% on the NASDAQ data. The model developed by Nguyen, Thu, and Yoon [25] demonstrates notable performance, achieving an average SMP accuracy of 60.7% across various interday

datasets of S&P 500 and KOSPI data and single c_i. The authors present baseline models, leveraging accuracies ranging from 51.49% to 57.36%.

III. Models

We define the whole stock data of $|C|$ companies $c_i \in C$ over the whole available time $\mathbb{T}$ as $X \in \mathbb{R}^{|C| \times \mathbb{T}}$. In each training step i we extract a sliding window of size ρ for the model introduced in Paragraph III-0a as $\mathcal{X}^{(i)}[j,v] = X[j,v]$ with $i \leq v \leq i+\rho$ and accordingly we define $\hat{\mathcal{X}}^{(i)}$ with $\rho := \Delta t$ for the baseline model in Paragraph III-0b. For simplicity we omit i in the following.

Following Yoo et al. [42] we employ a linear layer with parameters shared over all c_i with with ReLU activation function to convert the raw price features $\mathcal{X}$ into a latent feature representation $\bar{X}$ after passing $\mathcal{X}$ in a $\tanh(.)$ function.

a) Market Embedding Models: The Doc2Vec models discussed in Section II can be broadly classified into two overarching categories. The first encompasses the less prevalent models, such as Skip-Thought [19], which employ Word2Vec methodologies at the document level. The second category comprises Encoder-Decoder architectures, which are also comparatively more widespread in application. As the first category failed to yield usable outcomes in our preliminary (untabulated) experiments, we have directed our attention towards the latter ones.

Encoder-Decoder architectures operate on the principle of information transformation. Initially, an Encoder function $E(.)$ transforms the input into a latent representation $\vec{e}$, which encapsulates essential features of the input data. Subsequently, this latent representation is utilized by the Decoder function $D(.)$ to regenerate the input as $\hat{y}$, with the target variable $y = \mathcal{X}$ being the original input.

Particularly for textual data and stock price prediction tasks, the utilization of recurrent architectures for both $E(.)$ and $D(.)$, seems intuitive. This approach has been previously adopted in the models outlined in Section II. The most notable advantage of employing recurrent architectures is their ability to handle arbitrarily long input sequences during the encoding and decoding processes. However, our empirical findings reveal that while recurrent Encoder/Decoder models excel in faithfully reproducing input data, they fall short in generating abstracted representations. Consequently, we shall provide only a brief overview of this approach. For $E(.)$ and $D(.)$, we have the option to employ either Transformer based Encoder/Decoder architectures, as introduced by Vaswani et al. [35], or RNNs (and variants such as LSTM or Gated Recurrent Unit (GRU) networks). It is possible to employ complete Doc2Vec models by utilizing X as the embedding, thereby bypassing the typical embedding of $w^{(i)}$. Although we conducted several (untabulated) experiments employing Sentence-BERT [31], we refrained from pursuing this approach further for analogous reasons as those for the RNN/Transformer-based models delineated in Section IV .

Autoencoders are extensively employed models for training embeddings, adept at condensing large vectors of information into compact representations. Within the financial domain, for instance, Bao, Yue, and Rao [1] introduced an architecture built upon LSTM networks, leveraging Autoencoders to extract abstracted and generalized representations of $X^{(t)}$.

The Autoencoder model $A(.)$ is assembled from an encoder $E(.)$ and a decoder $D(.)$ and we define

$$A(\mathcal{X}) = \sigma(D(E(f(\mathcal{X})))) \tag{2}$$

where $D(.)$ and $E(.)$ are simple multi layer neural networks using the $\tanh(.)$ function between the N layers l_n. The function $f(.)$ is used to flat the input of dimensions $|C| \times \rho$. Here $\forall n : \dim(W_{l_n^D}) = \dim(W_{l_{N-n}^E})$ holds true with $\dim(W_{l_n^D})[0] < |C| \cdot \rho$. To initiate the model training, we establish the loss function as $\mathcal{L}_A = \text{MSE}(A(\mathcal{X}), \mathcal{X})$ (with MSE being the mean-squared-error) aiming to guide the model towards generating compressed vector representations. The QMSE $\vec{e} \in \mathbb{R}^\omega$ is defined as $\vec{e} = E(f(\mathcal{X}))$.

b) Baseline Models: We establish two baseline models for quantitative stock price prediction: the LSTM-based model M_L and the RNN-based model M_R. These models are constructed in accordance with the outlined previous literature, as exemplified by Feng et al. (for LSTMs) [28] (for RNNs), Ding et al. (for LSTMs), or [1] (for RNNs and LSTMs), as discussed in Section II. In our approach, we employ a unified vector z for all models, which is subsequently inputted into a linear prediction layer equipped with a Sigmoid function to generate the final prediction $\hat{y}$. We assign $z[i] = h_n[1,i]$ with h_n being the last hidden state [16] of either the RNN or the LSTM. In our training regimen, we employ either SPP or SMP. Regarding SPP we delineate the MSE loss function and the target $y = X^{(t+1)}$. For SMP we establish

$$y = \mathbb{I}^{(t)}(X^{(t)} > X^{(t+o)}) \tag{3}$$

drawn from the work of Yoo et al., to signify a binary label whether a given stock price has experienced an increase. We set $o = 1$ across all experiments. For SMP, we utilize the binary cross-entropy function as the loss metric (and transform y to be either 0 or 1).

c) Regulator Model: As delineated in Section I, we employ QMSEs as a regulatory mechanism during training. This serves the purpose of discerning whether the current training data, potentially derived from extraordinary events, may be unsuitable for transfer and generalization to other contexts. Our underlying assumption posits that atypical market situations arise due to exceptional occurrences, thereby necessitating a reduction in training progression. This adjustment might help the model training, as the existing data may insufficiently encapsulate the intricacies of "typical" and therefore to some extent predictable market dynamics.

To gauge the likelihood that the current training data originates from an exceptional circumstance, we quantify its deviation from other QMSEs. Through our conducted experiments, preliminary findings suggest that calculating the distance from all QMSEs may not be warranted. Instead, we compute the distance d to the preceding κ QMSEs. This approach enables the detection of significant deviations

from the prevailing market situations, potentially indicating exceptional situation. Given that d relies on the distances of $\vec{e}$ rather than the absolute values of X, our objective is to discern relationships and, particularly, movements that may elude capture by conventional metrics such as Exponential Moving Averages or volatility calculations (as visualized in Figure 3). We define the regulator model (returning d) as

$$R(\hat{\mathcal{X}^{(i)}}) = \left\| \left(\frac{1}{\kappa} \cdot \sum_{j=i-(1+\kappa)}^{i-1} E(f(\mathcal{X}^{(j)}) \right) - E(f(\mathcal{X}^{(i)})) \right\|_2 . \tag{4}$$

As $X^{(t)}$ is incorporated into the sliding window with each successive time step t, the system evaluates the congruence between this new data and the previously observed information. The magnitude of d directly correlates with the divergence from all other $\vec{e}$ vectors associated with the preceding κ market scenarios. In essence, an escalating d signifies an exceptionally rare market situation relative to the current time period. To integrate this concept into our training methodology, we posit that weight updates during backpropagation should be attenuated, or regulated, for unusual market scenarios. A regulation of training progress, akin to the introduction of noise, has previously been proposed by Yue et al. [30]. A straightforward approach to achieve this entails adjusting the loss term as $\mathcal{L} := \hat{\mathcal{L}} \cdot \left(1 + (1+d)^{-1}\right)$.

d) Neighbour Similarity for Stock Movement Prediction: Assessing contextualized embeddings presents a significant challenge, reflecting the broader effort within the research community to establish consistent quality metrics for evaluating word embeddings. One approach involves scrutinizing nearest neighbors (nearest neighbours approach (NNA) in the following) and assessing whether certain attributes of the nearest neighbor align with those of the respective data point, thereby discerning whether the embedding encapsulates meaningful relationships. While defining objective criteria in NLP poses smaller difficulty, this task becomes considerably more intricate within the financial domain [33]. In works such as [8] and [33], the proposition arises to ascertain whether the nearest neighbors of c_i represent similar companies or to cluster distinct c_i instances to ascertain if their respective industries correlate with the clusters.

We extend our investigation to examine the hypothesis that analogous market scenarios exhibit proximity within the vector space, with the expectation that discernible stock movements can be extrapolated from the prevailing market situations. To test this, we employ SMP, an idea which operates under the premise that similar market embeddings, indicative of analogous stock market conditions, are expected to correlate with comparable future stock movements. Accordingly, if $\| \vec{e}^{\,(i)} - \vec{e}^{\,(j)} \|_2$ is small, $\mathcal{X}^{(i)}$ and $\mathcal{X}^{(j)}$ are similar and $\mathbb{I}^{(i)}$ and $\mathbb{I}^{(j)}$ are expected to exhibit similarity across numerous instances $c_i \in C$. Conversely, if $\| \vec{e}^{\,(i)} - \vec{e}^{\,(j)} \|_2$ is large, it may suggest a limited similarity in directional movements, thus implying a relationship where $\mathbb{I}^{(i)} \approx \mathbb{I}^{(j)} \cdot (-1)$ holds true. Therefore we define the prediction of movements for $t+1$ as

$$\hat{y} = \text{sign}\left(\frac{1}{|K|} \cdot \sum_{\breve{e}_i \in K} (1+\breve{e}_i)^{-1} \cdot \mathbb{I}^{(i)} \right) \tag{5}$$

with $K = \text{topk}(\mathbb{E}, k)$ and $\mathbb{E} = \{\| \vec{e}^{\,(j)} - \vec{e}^{\,(t)} \|_2\}$. Vice versa $\mathbb{E} = \{- \| \vec{e}^{\,(j)} - \vec{e}^{\,(t)} \|_2\}$ can be used to predict $\mathbb{I} \cdot (-1)$.

e) Data reduction: Given that $\vec{e}$ serves as an unweighted representation derived from $\left[X^{(t)} \dots X^{(t-\rho)}\right]$, one might conjecture that a model could achieve comparable learning efficacy when trained on $\vec{e}$ instead of the raw stock data itself. As $\vec{e}$ encapsulates dense information spanning a timeframe ρ, we investigate whether the omission of a fraction of $X^{(t)}$ could yield comparable performance. In addition to models processing $\vec{e}$ directly, we explore a sliding window approach using ϕ^{-1} of the datapoints to ascertain the extent to which $\vec{e}$ accurately reflects the market dynamics within the timeframe Δt. For this we redefine the input U to the baseline models as $U^{(j)} = \hat{\mathcal{X}}^{(1+(j-1)\cdot\phi)}$.

IV. Experiments

Our findings will be presented in the subsequent sections, with the interpretation of the results deferred to Section V.

a) Setup: We leverage a dataset of stock data graciously provided by Alpha Vantage, a prominent stock data provider. Our initial set of companies, denoted as C, comprises 309 companies selected from the S&P-500 Index. The stock data spans from January 2000 to December 2023. To ensure data integrity and mitigate potential issues arising from missing or padded values, we include only those c_i for which data is available in the year 2000. Notably, missing values are more prevalent, particularly in intraday data, and this challenge becomes more pronounced with finer time granularities. To address missing data, we implement a padding procedure wherein we recursively assign $x^{(t)} := x^{(j)}$, with $\nexists x^{(i)} \in \mathbb{R} : \; j < i < t$. Additionally, we initialize $x^{(1)} := 0$ if the initial value is absent. For intervals with a granularity of sixty minutes, the proportion of padded data amounts to approximately 22%. While padded data yields satisfactory results for SPP, as demonstrated in [37], training $A(.)$ at higher granularities (e.g., minute or 15-minute resolutions) has proven challenging to stabilize. Furthermore, training for interday data presents its own set of challenges, notably the issue of too few data points (a well-documented concern in quantitative stock price prediction [25]) which frequently results in overfitting θ. Stock data is typically furnished in interval-based formats, encompassing features as Closing Price, Opening Price, Highest Price, Lowest Price, and Trading Volume (OHLCV). In our analysis, unless otherwise specified, we will primarily utilize the Closing Price following [37] [1] [13] [25]. We perform min-max normalization on all x_i individually for each c_i and exclude 0 values from the normalization process as [13]. This normalization procedure is essential as companies with high stock prices may disproportionately influence the range of values for other c_i. The dataset was partitioned into three subsets: a validation set comprising 9 % of the data,

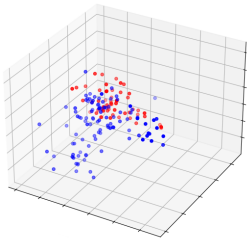


Fig. 2: 3D-PCA visualization of $\vec{e}$ for September 2001 for a Transformer based model.

a test set comprising 10% of the data, and a training set encompassing the remaining data. For the test set and validation set, six contiguous sequences were extracted from the time series. Only test set data was used for the visualizations in Paragraph IV-0b. Each experimental run was repeated 10 times, and the mean performance was computed across these repetitions. We ensured that every data point was included in the test set at least once. The model parameters were tuned using the training set, while the model hyperparameters were optimized using the validation set. The number of epochs for training each model was predetermined through multiple experiments with a maximum of 20 epochs and tailored to each model individually conducted prior to the main experiments. We employed the Adam optimizer with a weight decay of 0.002 [18].

b) Embedding Models: Following [33], embedding evaluation can be conducted using either "Extrinsic Evaluators" [33], wherein the embedding vector $\vec{e}$ is utilized in downstream tasks (as done in Paragraph IV-0c), allowing for comparison based on performance metrics, or "Intrinsic Evaluators" [33]. The latter includes methods such as the NNA suggested in Paragraph III-0d, which focuses on assessing intrinsic properties of the embeddings.

For training $A(.)$, we conducted a hyperparameter grid search and ended with a learning rate of $1{\times}10^{-4}$ and a relatively small batch size of 16. In addition to varying layers, we explored the utilization of different ω values. Notably, we observed that higher values of ω yielded superior outcomes in terms of $\mathcal{L}_A$. This observation can be rationalized by the notion that larger $\vec{e}$ dimensions facilitate a simpler reconstruction of the $|C| \times \rho$ input data points, as compression of knowledge becomes less stringent. However, our primary focus lies not solely in minimizing $\mathcal{L}_A$, as we consider this metric as auxiliary. Rather, our paramount objective is to identify analogous market situations. Hence, smaller values of ω carry greater significance in our analysis. Ultimately, we opt for employing a single-layer Autoencoder architecture for $A(.)$ with $\omega = 20$ independent of $\mathcal{L}_A$. This choice contrasts with the multiple-layer architectures commonly employed in comparable studies, such as those discussed in [1] and [3].

The QMSEs produced through the application of $A(.)$ exhibit proficiency in abstracting market situations. This proficiency is manifested in the discernment of rare occurrences i.e. crashes facilitated by the utilization of $\vec{e}$ or d. In Figure 1, we present a three-dimensional visualization of $\vec{e}$, wherein pre-event embeddings are distinguished by a distinct color from post-event embeddings. In Figure 3 we provide visualizations of d. The efficacy and utility of the QMSEs are prominently displayed, as evidenced by the presence of relatively clear clusters for most events (with exceptions discussed in Section V). This observation underscores the commonality and proximity of pre-event scenarios, contrasted sharply with the considerable divergence characterizing post-event scenarios, resulting in clusters exhibiting significantly greater inter-cluster distances. This implies that, for each trading interval, situations akin to the current one are notably absent. Henceforth, our focus will be directed solely towards $A(.)$ in all subsequent experiments.

The architectures utilizing Transformers failed to yield stable results for $\vec{e}$. This instability is evident in the considerably high standard deviation observed in (untabulated) training runs for the loss, particularly noticeable with small ω values. Notably, the distances between different Transformer-generated embeddings, exhibit a mean pairwise distance of 1.41 ($\omega = 64$), markedly lower than the average distances of 2.46 observed with $A(.)$ embeddings. Additionally, experiments utilizing cluster algorithms, such as DBSCAN [11], encountered challenges in organizing the Transformer-based QMSEs into meaningful structures. For instance, DBSCAN often failed to identify distinct clusters and frequently classified all $\vec{e}$ vectors as noise. Although RNN-based models demonstrated relatively stable outcomes in terms of loss values, the issue of small distances and the absence of discernible cluster structures persists.

We performed several preliminary experiments (untabulated) utilizing embeddings generated by both RNNs and Transformers for the SPP/SMP baseline models for producing representations U or d. The performance of these models was inferior compared with those generated by $A(.)$. Moreover, when employing Sentence-BERT embeddings, this issue was exacerbated, further diminishing the effectiveness of the models. In Figure 2, we juxtapose the $\vec{e}$ generated through a Transformer-based approach. Using the 9/11 Crash as an illustrative example, we observed the effectiveness of the visualization for $A(.)$-derived embeddings, a distinction that cannot be replicated with Transformer-generated $\vec{e}$.

c) Quantitative Baseline Models: The results for the best baseline for SPP/SMP models for each approach (U and $R(.)$) and each Δt are reported in Table I. We conducted experiments across a range of model sizes, layer configurations, and variations; however, we opted not to include tabulated results for all iterations due to their inability to match the performance levels achieved by the models listed (often below 50 % for SMP). In addition to varying model sizes, we conducted experimentation with different hyperparameters, including batch size and learning rates. Ultimately, for the SMP model, we settled on a learning rate of $2{\times}10^{-3}$ and a batch size of 16. The RMSE for SPP is reported for the normalized values. Comparatively, the baseline performance aligns with the results achieved by

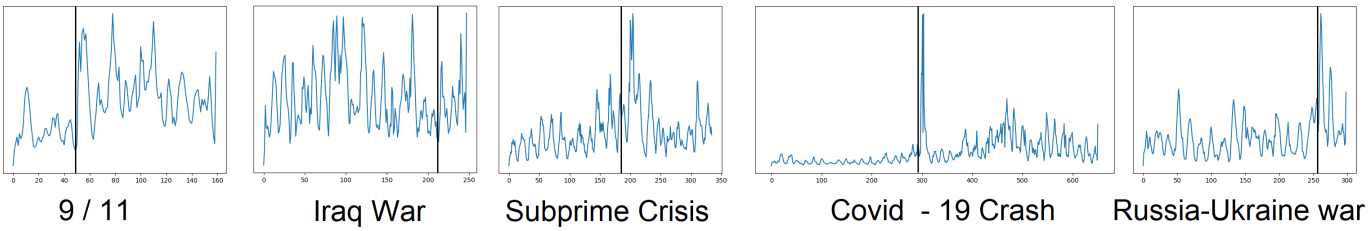


Fig. 3: Visualization of d during significant economic events over the past 24 years. Please acknowledge that the closure of the New York Stock Exchange for four days following the 9/11 events resulted in a limited dataset during that period. The X-axis represents the time steps in hours. The black line symbolizes the time step of the respective event. The Y-axis, which represents d, is not labelled, as the values vary greatly over time (for each event) and we prefer relative representations for visualization purposes.

the baseline models referenced in Section II or even the actual models.

In all tabulated experiments employing the U approach, we consistently set $\phi = 2$ as larger values invariably result in model performance plateauing at 50% accuracy for SMP or even lower, rendering the models untrainable. We established the value of $\rho = 6$ through empirical experimentation. Subsequently, to corroborate this determination, we test the best performing SMP baseline model (single-layer M_{L256}), and systematically explored various parameter configurations for ρ. The results of these investigations are illustrated in Figure 4.

For the respective best performing SMP and SPP model (single-layer M_{L256} and single-layer M_{R256}) we also experimented with using $\vec{e}$ as the input without using U or ϕ. The performance was 53.35 ± 0.12 for SMP and 0.083 ± 0.011 for SPP. To ensure that the model benefits specifically from the weighted loss function rather than solely from a diminished $\mathcal{L}$, we conducted three separate training iterations for each model. In these iterations, the $\mathcal{L}$ is scaled by the mean value of $1 + (1 + d)^{-1}$ computed across the entire training dataset (1.209), with worse performance in all cases.

Experiments involving the inclusion of $\Pi = \vec{e}$, inspired by a CLM approach ($P_\theta(X^{(t+1)}|\Pi, X^{(t)}, \ldots, X^{(t-\Delta t)})$), are not tabulated. We explored two methods: appending $\vec{e}$ to $\bar{X}$ or scaling the latent representation by assigning $\hat{\bar{X}}[i] := \bar{X}[i] + \vec{e}\,[i]$. Consequently, we experimented with reducing the sizes of the baseline models to accommodate small ω values. However, this adjustment often resulted in models achieving only 50% accuracy in SMP or performing even worse. Scaling $\bar{X}$ did not yield any improvement and typically extended the training duration until convergence.

d) Neighbour Similarity: The SMP methodologies have consistently yielded a stable yet relatively modest test set accuracy, averaging approximately 0.52 ± 0.023, across all price features except for trading volume. Furthermore, the correlation scores between accuracy and summed vector distances (or their inverse) persist at a consistent but diminutive scale, ranging between 0.01 and 0.04 The sole exception lies in the prediction of trading volume. Employing inverted distance vectors exhibits promise, resulting in test set accuracies of 0.63 ± 0.03, while utilizing nearest distances yields accuracies of 0.56 ± 0.02. In Figure 5, we visually depict the notable correlations with scores of 0.25 ± 0.02. This observations remain consistent across various values of κ, including $5, 20, 100, 1000$. We did not discern any correlation between κ and accuracy, nor did we observe any enhancement with different κ values. This approach has been validated across both interday data and the otherwise utilized sixty-minute intraday data.

V. Discussion and Future Work

Considering the inherent non-stationarity of stock data alongside established economic theories like the RWT and EMH, which cast doubt on the predictability of stock prices, there arises a pertinent inquiry regarding the characterization of **every** market scenario as potentially exceptional, while other works such as [34] acknowledge "standard market conditions". Therefore it becomes imperative to view every market condition in a time-dependent context, utilizing parameters such as κ, ρ, or Δt. Despite acknowledging the critical considerations inherent in the discussed concepts, we defer the resolution of these questions to the field of economics, focusing instead on the (at least partially) successful application of our findings.

As noted, for instance in [33], the absence of objective quality criteria for embeddings necessitates our reliance on the efficacy of QMSE visualizations, particularly during significant events post-2000. This approach allows us to identify and discard inadequately performing QMSE methods, primarily through clustering and distance measurements but also through the satisfying visualizations of $\vec{e}$ where we were able to clearly recognize major crash events. The visual representations of the Subprime Crisis and the Iraq war-induced market crash exhibit less clarity compared with other events. In the context of the Subprime Crisis, the ambiguity potentially arises from its pre-existing trajectory prior to the Lehman Brothers bankruptcy, which acted as a significant catalyst rather than an isolated event. Similarly, the complexities surrounding the Iraq war are attributable to heightened political tensions preceding its commencement [24].

In our pursuit of generating robust embeddings, the utilization of $f(.)$ and flattening seems crucial, as recurrent methods

failed to yield satisfactory results. Conversely, we speculate that the generated representations exhibit proficiency in reconstruction, as evidenced by the RMSE, yet lack sufficient abstraction of complex market scenarios. Additionally, given the importance of small ω values, we posit that a dense abstracted representation of the market situation is paramount.

The evaluation capabilities for QMSEs extend beyond mere illustration, with practical utility evident in downstream applications. As demonstrated in work such as [8], the potential for assessing portfolio management and risk mitigation is considerable. This can be realized through trading simulations akin to those presented in [9] or [13] (for SPP).

Future research endeavors will encompass the evaluation of QMSEs across diverse market contexts, particularly at smaller time resolutions or an expanded C. In this study, our models are trained on intraday data with 60-minute resolution. The primary rationale, as expounded in Section IV, pertains to challenges in achieving training stability at finer frequencies, likely attributable to data incompleteness. Utilizing QMSEs at shorter time intervals holds promise for discerning transient phenomena such as flash crashes, characterized by rapid and volatile price declines followed by swift recoveries [21].

The baseline models exhibit a performance on par with the baseline standards observed in other quantitative models discussed in Section II. As emphasized from the outset, the primary objective of this study is not to achieve exceptional SMP/SPP results but rather to leverage QMSEs, with the baseline models serving as a reference point. We have illustrated that the integration of $R(.)$ augments the training regimen across various scenarios, particularly evident in larger models and for SPP, albeit without consistent superiority over smaller counterparts. Notably, while the efficacy of $R(.)$ did not uniformly enhance performance, a discernible reduction in standard deviation was observed. This reduction suggests a potential successful stabilization of the training process, possibly achieved by mitigating the influence of highly uncommon training data. This discrepancy could stem from the capacity of larger or more intricate models to capture detailed patterns, which in turn may amplify the impact of exceptional situations or noise, as large ML models are shown to be able to fit random noise data [43]. Irrespective of the optimal performance achieved by the baseline models, it is imperative to acknowledge that more complex models might derive substantial benefits from QMSE-dependent loss regulation. This observation holds significance when juxtaposed with SOTA quantitative models discussed in Section II, which, along with fundamental models, exhibit significantly greater complexity compared with the baseline models discussed in Paragraph III-0b. Moreover, efforts will be directed towards devising strategies to incorporate $\Pi = \vec{e}$ for CLM inspired approaches, given our unsuccessful experimentation in this regard.

Replacing the data with U results in inferior performance, particularly noticeable with less complex models where the degradation is more pronounced. However, even with this substitution, the model maintains a degree of predictive capability, consistently surpassing the accuracy expected by chance. In forthcoming experiments, particularly those entailing the utilization of resource-intensive Transformers, there exists an opportunity to redefine U by imposing $0 < \phi < 1$ to effectively mitigate input dimensionality. One such sophisticated model, employing Transformer-based architectures, was delineated by Voigt at el. as "Adapted Speech Models" [37], a pioneering approach wherein we conceptualized the transformation of X into a structure reminiscent of NLP-like-sentences. This intricate process entails the flattening of $X^{(t)}$ and the incorporation of current stock prices to refine multidimensional contextualized embedding vectors for each c_i. Subsequently, the transformed representation is fed into adapted language models such as BERT [5], GPT-2 [29], or

Model	$R(.)$		Default		U	
	SPP	SMP	SPP	SMP	SPP	SMP
$\Delta = 8$						
1αL	8.7 ± 2	3.5 ± 3	8.5 ± 4	3.5 ± 2	8.8 ± 2	1.6 ± 1
2αL	8.6 ± 2	3.7 ± 6	8.7 ± 5	3.5 ± 4	9.1 ± 3	3.1 ± 4
1βL	9.1 ± 1	3.9 ± 3	8.6 ± 4	4.5 ± 3	9.0 ± 4	4.5 ± 3
2βL	8.6 ± 3	4.2 ± 3	8.8 ± 5	3.5 ± 4	9.0 ± 3	1.5 ± 3
1γL	9.0 ± 1	3.9 ± 4	8.7 ± 6	3.7 ± 3	8.7 ± 8	3.5 ± 5
2γL	8.6 ± 4	3.8 ± 3	8.3 ± 4	3.7 ± 2	9.0 ± 4	1.9 ± 4
1αR	8.8 ± 1	4.3 ± 3	8.6 ± 5	4.3 ± 9	8.8 ± 7	3.1 ± 3
2αR	8.6 ± 2	4.1 ± 3	8.9 ± 6	3.4 ± 3	9.4 ± 3	2.1 ± 3
1βR	8.7 ± 2	2.9 ± 2	8.9 ± 5	3.2 ± 5	9.1 ± 1	3.4 ± 6
2βR	9.1 ± 2	5.1 ± 3	9.2 ± 6	3.3 ± 3	9.8 ± 4	3.0 ± 4
1γR	8.9 ± 1	3.4 ± 1	9.2 ± 3	3.0 ± 3	9.9 ± 7	2.8 ± 4
2γR	9.2 ± 2	3.8 ± 2	8.8 ± 5	3.1 ± 3	9.2 ± 4	0.9 ± 3
$\Delta = 32$						
1αL	9.8 ± 2	4.5 ± 2	9.2 ± 5	$\mathbf{6.2} \pm 3$	8.6 ± 3	2.2 ± 4
2αL	9.8 ± 1	5.5 ± 1	9.0 ± 2	4.7 ± 2	9.2 ± 2	4.0 ± 5
1βL	9.5 ± 1	4.5 ± 3	9.7 ± 2	4.7 ± 5	8.9 ± 2	4.5 ± 6
2βL	3.2 ± 2	5.5 ± 3	9.2 ± 2	4.5 ± 4	9.2 ± 4	2.7 ± 3
1γL	10.9 ± 2	4.8 ± 2	9.9 ± 5	4.7 ± 6	8.4 ± 4	4.0 ± 8
2γL	8.9 ± 1	4.5 ± 3	8.6 ± 4	4.5 ± 5	9.2 ± 6	2.2 ± 5
1αR	$\mathbf{8.2} \pm 1$	3.1 ± 3	9.4 ± 4	3.2 ± 7	8.4 ± 1	2.0 ± 5
2αR	8.6 ± 1	5.0 ± 3	9.2 ± 5	4.1 ± 7	9.4 ± 5	2.1 ± 3
1βR	8.4 ± 3	8.3 ± 3	9.8 ± 4	3.9 ± 4	9.7 ± 9	3.8 ± 3
2βR	8.6 ± 2	4.9 ± 2	9.1 ± 3	4.2 ± 3	9.2 ± 3	4.0 ± 2
1γR	9.9 ± 2	4.1 ± 3	9.6 ± 2	4.1 ± 4	9.7 ± 5	3.7 ± 1
2γR	9.5 ± 3	4.0 ± 2	9.0 ± 3	3.8 ± 1	9.2 ± 4	2.5 ± 2
$\Delta = 64$						
1αL	9.3 ± 4	4.7 ± 2	9.7 ± 1	5.0 ± 6	9.9 ± 3	2.5 ± 9
2αL	9.7 ± 1	3.7 ± 1	8.6 ± 2	4.2 ± 7	8.9 ± 3	2.8 ± 2
1βL	9.3 ± 1	3.5 ± 2	9.7 ± 2	3.5 ± 9	9.8 ± 4	3.5 ± 5
2βL	9.7 ± 2	4.5 ± 1	9.9 ± 4	3.7 ± 6	10.3 ± 3	3.7 ± 3
1γL	8.3 ± 2	3.5 ± 2	9.8 ± 2	3.7 ± 8	9.8 ± 6	2.3 ± 9
2γL	10.2 ± 5	5.0 ± 2	9.7 ± 1	4.5 ± 7	9.8 ± 3	2.7 ± 6
1αR	9.0 ± 5	3.2 ± 2	8.3 ± 2	3.6 ± 3	9.0 ± 5	3.4 ± 9
2αR	8.3 ± 3	3.0 ± 1	8.3 ± 1	3.2 ± 5	8.8 ± 2	2.8 ± 6
1βR	8.4 ± 2	3.1 ± 3	8.8 ± 4	4.0 ± 9	8.8 ± 2	3.5 ± 4
2βR	8.5 ± 2	3.9 ± 3	8.8 ± 5	3.8 ± 9	9.4 ± 8	3.6 ± 7
1γR	8.5 ± 2	4.1 ± 2	9.4 ± 4	3.7 ± 8	9.9 ± 3	2.3 ± 6
2γR	8.4 ± 1	4.0 ± 1	9.0 ± 5	3.0 ± 8	9.5 ± 2	1.8 ± 6

TABLE I: Presentation of results from the most effective baseline models. The first number in the "Model" column denotes the layer number. The "L" and "R" encode M_L or M_R respectively. The models sizes are indicated by $\alpha = 256$, $\beta = 512$ and $\gamma = 768$. The deviations are scaled by 10^{-1} for SMP and 10^{-3} for SPP. SPP RMSE values are scaled by a factor of 10^{-2}. The SMP accuracies in % were subtracted with 50 for presentation reasons.

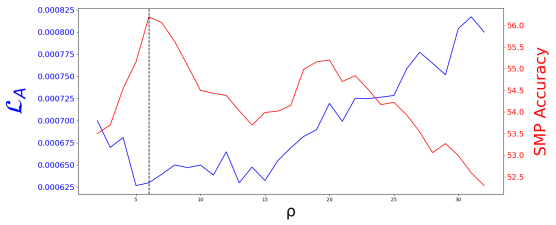


Fig. 4: Comparing $\mathcal{L}_A$ and the SMP accuracy for different ρ values using M_{L256} and $R(.)$.

TransformerXL [4]. Additionally, we will explore the possibility of enhancing the representational capacity of our model by feeding $\left[\vec{e}^{\,(t)}, \vec{e}^{\,(t-1)}, \ldots, \vec{e}^{\,(t-\Delta t)}\right]$ into the Adapted Speech Models. This approach allows for a departure from (stock) regression data (by using embeddings) while mitigating the computational burden associated with scaling the input length by $|C|$.

The NNA performs very weakly and shows that similar macro situations, at least from those learned by $A(.)$, are indicative of future price developments without learning complex rules through another ML model (as done in most event embedding model like [7]). Notably, its robust performance in forecasting trading volume warrants attention. We hypothesize that this may stem from its easier predictability, possibly owing to a stronger correlation with market conditions.

VI. Summary

In summarizing our findings, we have expanded upon the existing research trajectory, illustrating how quantitative stock price prediction shares similarities with language modeling, as demonstrated in prior studies such as [36] [37]. In this context, we have investigated the adaptation of Doc2Vec models to stock market data, culminating in the derivation of QMSEs through the modification of a Autoencoder model with flattened input. Subsequently, we conducted direct assessments of the QMSEs, particularly focusing on their performance during significant economic events, as illustrated in Figure 1. Moreover, we explored diverse applications of the QMSEs, including their utility in regulatory frameworks for training purposes and their potential for reducing input data in SMP /SPP tasks. Finally, we endeavored to ascertain whether close QMSE values exhibit correspondingly similar future stock movements, thus probing the predictive capabilities of our model.

Acknowledgment

The stock data used for the models presented in this research was collected courtesy of research access kindly provided by Alpha Vantage. [1]
We used the chatGPT AI [2] to improve the text in all sections of this work.

[1]https://www.alphavantage.co
[2]https://chat.openai.com/

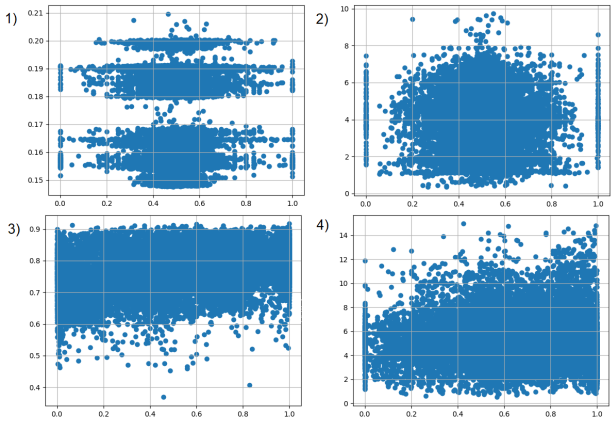


Fig. 5: Y-Axis: mean distance, X-Axis: accuracy. 1) Correlation of Closing Price Movement Accuracy and distance, 2) Correlation of Inverted Closing Price Movement Accuracy and distance, 3) Correlation of Volume Movement Accuracy and distance, Correlation of Inverted Volume Movement Accuracy and distance.

References

[1] Wei Bao, Jun Yue, and Yulei Rao. A deep learning framework for financial time series using stacked autoencoders and long-short term memory. *PLOS ONE*, 12(7):1–24, 07 2017.

[2] Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. Generating sentences from a continuous space. In *Conference on Computational Natural Language Learning*, 2015.

[3] Yushi Chen, Zhouhan Lin, Xing Zhao, Gang Wang, and Yanfeng Gu. Deep learning-based classification of hyperspectral data. *Selected Topics in Applied Earth Observations and Remote Sensing, IEEE Journal of*, 7:2094–2107, 06 2014.

[4] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G. Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *CoRR*, abs/1901.02860, 2019.

[5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[6] Qianggang Ding, Sifan Wu, Hao Sun, Jiadong Guo, and Jian Guo. Hierarchical multi-scale gaussian transformer for stock movement prediction. In Christian Bessiere, editor, *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 4640–4646. International Joint Conferences on Artificial Intelligence Organization, 7 2020. Special Track on AI in FinTech.

[7] Xiao Ding, Yue Zhang, Ting Liu, and Junwen Duan. Knowledge-driven event embedding for stock prediction. In Yuji Matsumoto and Rashmi Prasad, editors, *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2133–2142, Osaka, Japan, December 2016. The COLING 2016 Organizing Committee.

[8] Rian Dolphin, Barry Smyth, and Ruihai Dong. Stock embeddings: Learning distributed representations for financial assets, 2022.

[9] Xin Du and Kumiko Tanaka-Ishii. Stock embeddings acquired from news articles and price history, and an application to portfolio optimization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, July 2020. Association for Computational Linguistics.

[10] J. L. Elman. Finding structure in time. *Cognitive Science*, 14:179–211, 1990.

[11] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, page 226–231. AAAI Press, 1996.

[12] Eugene F. Fama. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 25(2):383–417, 1970.

[13] Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Trans. Inf. Syst.*, 37(2), mar 2019.

[14] Xavier Gabaix, Ralph Koijen, and Motohiro Yogo. Asset embeddings. *SSRN Electronic Journal*, 01 2023.

[15] Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-predict: Parallel decoding of conditional masked language models. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, November 2019. Association for Computational Linguistics.

[16] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997.

[17] M. G. Kendall and A. Bradford Hill. The analysis of economic time-series-part i: Prices. *Journal of the Royal Statistical Society. Series A (General)*, 116(1):11–34, 1953.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[19] Ryan Kiros, Yukun Zhu, Russ R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought vectors. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[20] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 1188–1196, Bejing, China, 22–24 Jun 2014. PMLR.

[21] Zachary S. Levine, Scott A. Hale, and Luciano Floridi. The october 2014 united states treasury bond flash crash and the contributory effect of mini flash crashes. *PLOS ONE*, 12(11):1–14, 11 2017.

[22] Jintao Liu, Hongfei Lin, Xikai Liu, Bo Xu, Yuqi Ren, Yufeng Diao, and Liang Yang. Transformer-based capsule network for stock movement prediction. In *Proceedings of the First Workshop on Financial Technology and Natural Language Processing*, pages 66–73, Macao, China, August 2019.

[23] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, 2013.

[24] Williamson Murray and Robert H. Scales. *The Iraq War: A Military History*. Harvard University Press, 2003.

[25] Thi-Thu Nguyen and Seokhoon Yoon. A novel approach to short-term stock price movement prediction using transfer learning. *Applied Sciences*, 9(22), 2019.

[26] Ramkrishna Patel, Vikas Choudhary, Deepika Saxena, and Ashutosh Kumar Singh. Review of stock prediction using machine learning techniques. In *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, pages 840–846. IEEE, 2021.

[27] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[28] Yao Qin, Dongjin Song, Haifeng Cheng, Wei Cheng, Guofei Jiang, and Garrison W. Cottrell. A dual-stage attention-based recurrent neural network for time series prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, IJCAI'17, page 2627–2633. AAAI Press, 2017.

[29] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[30] Scott E. Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015.

[31] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics.

[32] Jerald Pinto David Runkle Richard DeFusco, Dennis McLeavey. *Quantitative Investment Analysis*. 2015. John Wiley Sons. (Cited on pages 1 and 3).

[33] Bhaskarjit Sarmah, Nayana Nair, Dhagash Mehta, and Stefano Pasquali. Learning embedded representation of the stock correlation matrix using graph machine learning, 2022.

[34] Ramit Sawhney, Arnav Wadhwa, Shivam Agarwal, and Rajiv Ratn Shah. FAST: Financial news and tweet based time aware network for stock trading. In Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2164–2175, Online, April 2021. Association for Computational Linguistics.

[35] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[36] Frederic Voigt. Adapting natural language processing strategies for stock price prediction. DC@KI2023: Proceedings of Doctoral Consortium at KI 2023, 2023.

[37] Frederic Voigt, Kai Von Luck, and Peer Stelldinger. Assessment of the applicability of large language models for quantitative stock price prediction. In *Proceedings of the 17th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '24, page 293–302, New York, NY, USA, 2024. Association for Computing Machinery.

[38] Ahmed. S. Wafi, Hassan Hassan, and Adel Mabrouk. Fundamental analysis models in financial markets – review study. *Procedia Economics and Finance*, 30:939–947, 2015. IISES 3rd and 4th Economics and Finance Conference.

[39] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

[40] Yumo Xu and Shay B. Cohen. 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)*, pages 1970–1979, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[41] Linyi Yang, Zheng Zhang, Su Xiong, Lirui Wei, James Ng, Lina Xu, and Ruihai Dong. Explainable text-driven neural network for stock prediction. In *2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS)*, pages 441–445, 2018.

[42] Jaemin Yoo, Yejun Soun, Yong-chan Park, and U Kang. Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 2037–2045, New York, NY, USA, 2021. Association for Computing Machinery.

[43] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Commun. ACM*, 64(3):107–115, feb 2021.