Encoding Stock Returns Relationships via Latent Embeddings for Enhanced Portfolio Optimization

Rian Dolphin[1][0000−0002−5607−9948], Barry Smyth[1,2][0000−0003−0962−3362], and Ruihai Dong[1,2][0000−0002−2509−1370]

[1] School of Computer Science, University College Dublin, Dublin, Ireland
rian.dolphin@ucdconnect.ie
[2] Insight Centre for Data Analytics, University College Dublin, Dublin, Ireland
{barry.smyth, ruihai.dong}@ucd.ie

Abstract. Capturing the many, varied relationships between financial asset price movements is essential for several financial tasks. Conventional methods, which rely on pairwise similarity measurements, fail to capture the subtlety and nuance that define these relationships. Inspired by this need for more sophisticated approaches to modeling, we outline a task-agnostic, self-supervised framework to learn embedding representations for financial assets, by using only their historical returns. Rather than computing similarity directly on raw asset returns, this approach encodes common co-occurrence of similar return fluctuations in a latent embedding space. Using clustering, nearest neighbor, and sector classification experiments, we demonstrate that the learned embeddings are task-agnostic. We then propose new context selection strategies aimed at the downstream task of portfolio optimization and minimizing volatility. We evaluate our approach in a long-only hedging experiment on more than five years of out-of-sample data. A comparative analysis with traditional benchmarks reveals statistically significant performance benefits accruing to our embedding-based approach.

Keywords: Machine Learning · Latent Embeddings · Portfolio Optimization · Risk Management

1 Introduction

The financial markets have long been a fertile ground for the exploration of time series analysis, with its complexity and significance attracting the interest of researchers and industry practitioners. While recent time series methodologies have made significant strides in forecasting and understanding market dynamics [20], there remains an important gap in the literature: the frequent sidelining of relational dynamics among assets [21]. Capturing these inter-asset relationships is vital for tasks such as portfolio optimization [15], hedging risk, and industry classification [17]. Historically, Markowitz's groundbreaking work on portfolio theory established asset correlation as a foundational similarity technique [15].

Yet, as the financial landscape has evolved, so have the critiques of correlation as the sole determinant for financial similarity [14].

In parallel to this, the area of natural language processing (NLP) has experienced a renaissance, especially with the advent of embedding techniques that encode semantic inter-relationships [16]. While applications of NLP in the financial domain are popular, direct application of embedding-based techniques on primary financial data streams, like stock returns, is still in its nascent stage [6].

Against this backdrop, our paper outlines a methodology designed to mine and encode the relational nuances that exist in financial time series data into embeddings. In Section 3, we describe this technique in detail, and the utility of our derived representations is underscored through applications in industry sector classification and optimized portfolio formulation. In summary, our paper offers the following contributions:

- In Section 3, we present a way to extract relational information solely from financial time series data and encode this information as embeddings.
- Using several qualitative case studies we show that the embedding space encodes rich information about company fundamentals, such as industry sector. For example, in Section 5.3, the embedding space exposes the well-documented and undesirable subjectivity of current industry sector classification schemes.
- In Section 5.4, we use the embeddings to categorize assets into industry sectors, offering an alternative to traditional classification methods.
- Finally, in Section 4.3, we propose two novel variations of our embedding framework tailored to portfolio optimization, and in Section 5.5 demonstrate their utility by constructing portfolios with statistically significantly reduced out-of-sample volatility.

2 Related Work

Quantifying relationships between asset returns is essential for financial professionals tackling tasks like portfolio optimization, hedging, and sector classification [17]. Markowitz, in his foundational paper on modern portfolio theory, utilized the covariance of returns as a mechanism for risk management strategies [15]. Consequently, correlation became a widely accepted metric to measure the similarity in returns between two financial assets, though its merit has since been challenged [12,14]. Over time, alternative methods have been introduced, such as a geometric approach to asset similarity [4] and a refined correlation-based technique [8]. However, solutions rooted in ML for this specific problem are still in nascent stages.

While computing pairwise correlation values for asset returns is an actionable approach, exploring asset relationships through learned embedding representations allow non-pairwise trends to be explored. Distributed representations, especially in language modeling, have been extensively researched in recent years. Algorithms such as Word2Vec [16] have further demonstrated the value of neural embedding techniques across various sectors, from recommendation systems to medical applications [3].

In the financial sector, the primary application of embeddings has revolved around aggregating pre-trained word embeddings from language models, neglecting the potential of creating embeddings from non-textual financial datasets like historical returns [13,22]. However, in recent years, the literature has begun to explore learning embeddings from non-textual financial datasets. For example, there have been several papers suggesting ways of learning stock embeddings based on historical returns data [6,7,18,23][3] Aside from using returns data, Gabaix et al. [10] propose to learn asset embeddings from investors' holdings data, [5] use company co-occurrence in news articles, and [19] leverage co-occurring assets across funds to learn mutual fund embeddings.

3 Base Architecture

Drawing parallels with the principles of distributional semantics in natural language processing [9], the approach we outline uses the concept of *context assets* [7] to infer embeddings for a collection of assets. In linguistic theories, the distributional hypothesis serves as the backbone of several notable language models [16], emphasizing that words sharing common contexts often have related meanings. A similar parallel exists within the financial sector: assets (here we deal with a company's stock) with analogous attributes—like operating within the same industry segment—tend to showcase correlated price movements [11]. Accordingly, our approach focuses on tailoring the curation of these *context assets*, supplemented by noise filtering techniques, for specific downstream tasks. Consequently, the framework is adept at producing embeddings which encode intricate connections among financial assets solely on the foundation of historical price sequences. In Section 3.2, we delineate the base framework for learning asset embeddings from returns time series [7], followed by a discussion on noise reduction techniques in Section 3.3 and strategies for tailored context curation in Section 4.3.

3.1 Data & Context Assets

Consider a collection of assets represented as $U = a_1, \dots, a_n$. Each asset a_i associates with a chronological series $\mathbf{p}_{a_i} = p_0^{a_i}, \dots, p_T^{a_i}$, representing its price at predefined discrete points in time $t \in 0, 1, \dots, T$ (such as daily or weekly). From these price details, one can derive a series of *returns* $\mathbf{r}_{a_i} = r_1^{a_i}, \dots, r_T^{a_i}$, as depicted in Equation (1).

$$r_t^{a_i} = \frac{p_t^{a_i} - p_{t-1}^{a_i}}{p_{t-1}^{a_i}} \tag{1}$$

Grounded in these return series, we can identify groups we call *target:context sets*, and each set comprises a target asset along with its group of context assets.

[3] This paper is an extension of [7] with a particular focus on tailoring the embeddings for portfolio optimization tasks.

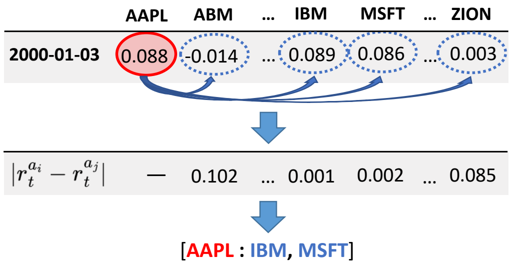

Fig. 1. Formulation of training data, denoting target:context asset groupings.

Specifically, given a context dimension of C (a modifiable parameter), the context assets for the target asset a_i at juncture t are the C assets that bear the closest return at that instance. This closeness is measured by the smallest absolute difference in return between a potential asset a_j and the target asset a_i, described as $|r_t^{a_i} - r_t^{a_j}|$. This context selection strategy is designed to capture the hypothesis mentioned previously: assets with similar attributes (business model etc.) tend to be related in terms of returns fluctuations [11].

An example is presented in Figure 1, with AAPL as the target asset for 3 January 2000. We assess the absolute difference between AAPL's return for that date against the return of every other asset on that same date. The C assets that exhibit the closest return are chosen as the context assets, with AAPL being excluded. In this instance, IBM and MSFT are selected due to their minimal return differences compared to AAPL. For training, we construct a target:context set for every asset at each discrete point in time, leading to a total of $|U| \times T$ sets, where $|U|$ denotes the count of assets in the dataset.

A sample of a target:context set for $C = 3$, is represented as $\mathcal{S}(a_1, t) = [a_1 : a_{270}, a_{359}, a_{410}]$, which translates to *[Apple : IBM, Microsoft, Oracle]*, with 270 as the index for IBM within the dataset, so that a_{270} corresponds to IBM. Thus, at time t, the three equities most closely mirroring Apple Inc. in returns were IBM, Microsoft, and Oracle.

3.2 Learning Embeddings

Classification problems can be broken down into two stages, the inference stage in which training data is used to learn a model for the posterior class probabilities, and the subsequent decision phase where the posterior probabilities are used to make class assignments [1]. To learn the asset embeddings we consider framing the problem as the inference stage of a classification task, since we are concerned only with learning embeddings rather than any downstream classification task.

To formulate our embedding strategy, we begin by observing a context set, as we previously defined. By design, this context set, denoted as $\mathcal{S}(a_T, t)$, comprises a set of assets that share close return characteristics with the corresponding target asset at a specific point in time t. Given this setup, our goal is to estimate the likelihood, or posterior probability, of any asset a_i from our collection U

being identified as this target asset a_T, based on the observed assets present in the context set. In essence, by considering the context set, we want to assess how probable it is for asset a_i to play the role of a_T, the true target asset, given the observed context assets. Thus, we are interested in estimating $\mathbb{P}(a_i|\mathcal{S}(a_T,t))$, which we can do by applying Bayes' theorem in the form

$$\mathbb{P}(a_i|\mathcal{S}(a_T,t)) = \frac{\mathbb{P}\left(a_i, \mathcal{S}(a_T,t)\right)}{\sum_{k=1}^{|U|} \mathbb{P}\left(a_k, \mathcal{S}(a_T,t)\right)} \tag{2}$$

To estimate these joint probabilities, we recall that we want to learn embeddings such that assets commonly experiencing similar returns patterns (and thus commonly appearing together in target:context sets) will have similar representations in the latent space. As a result, we estimate the joint probability using the dot product as

$$\mathbb{P}\left(a_i, \mathcal{S}(a_i,t)\right) \propto \mathbf{v}_{a_i}^{\mathbf{T}} \cdot \mathbf{h} \tag{3}$$

$$\mathbf{h} = \frac{1}{C} \cdot \sum_{a_j \in \mathcal{S}(a_i,t)} \mathbf{v}_{a_j} \tag{4}$$

where $\mathbf{v}_{a_i}$ is the embedding representation for asset a_i, and $\mathbf{h}$ is the aggregate representation of the context set $\mathcal{S}(a_i,t)$, which in its most simple form is computed as an average of the embedding representations for all assets in the context see (Equation 4). We note that both $\mathbf{v}_{a_i}$ and $\mathbf{h}$ are N dimensional vectors, where N is a hyperparameter representing the embedding dimensionality, and at the outset, each embedding vector is initialized as a multivariate normal distribution with mean vector $\mathbf{0}$ of dimension N and identity covariance matrix $\mathbf{I}_N$.

By using the softmax function to ensure adherence to the properties of a probability distribution[4], namely that each value lies in the $[0,1]$ range and that the probabilities sum to 1 across all assets, we can rewrite Equation 2 as

$$\mathbb{P}(a_i|\mathcal{S}(a_T,t)) = \frac{\exp(\mathbf{v}_{a_i}^{\mathbf{T}} \cdot \mathbf{h})}{\sum_{k=1}^{|U|} \exp(\mathbf{v}_{a_k}^{\mathbf{T}} \cdot \mathbf{h})} \tag{5}$$

In this way, we can learn the set of weight vectors $\mathbf{v}_{a_i}$ for $i \in 1,2,...,|U|$ that maximize the likelihood of the observed target asset $\mathbb{P}(a_i = a_T|\mathcal{S}(a_T,t))$ for all context:target pairs. While we could seek to maximize the posterior probability directly, in practice, it is more convenient to minimize the negative log of the posterior probability. Because the logarithm is monotonically increasing, minimization of the log of a function is equivalent to minimization of the function itself, and taking the log not only simplifies the mathematical analysis, but it also helps in terms of numerical stability [1]. As a result, the loss function that we seek to minimize is shown in Equation 6.

[4] We note that the softmax outputs from our model, while allowing a probabilistic interpretation, are based on model assumptions and might not reflect calibrated probabilistic confidence.

$$
\begin{aligned}
\mathcal{L} &= -\log \mathbb{P}(a_T | \mathcal{S}(a_T, t)) \\
&= -\log \frac{\exp(\mathbf{v}_{a_T}^{\mathbf{T}} \cdot \mathbf{h})}{\sum_{k=1}^{|U|} \exp(\mathbf{v}_{a_k}^{\mathbf{T}} \cdot \mathbf{h})} \\
&= -\mathbf{v}_{a_T}^{\mathbf{T}} \cdot \mathbf{h} + \log \sum_{k=1}^{|U|} \exp(\mathbf{v}_{a_k}^{\mathbf{T}} \cdot \mathbf{h})
\end{aligned}
\tag{6}
$$

To simplify the loss function and increase interpretability, let us make the substitution $u_{a_k} = \mathbf{v}_{a_k}^{\mathbf{T}} \cdot \mathbf{h}$. We can interpret u_{a_k} as a *score* for asset a_k, whereby u_{a_k} will have a higher value if the embedding representation of a_k is similar (in terms of dot product similarity) to the aggregate context embedding represented by $\mathbf{h}$. The updated loss function is given in Equation 7.

$$
\mathcal{L} = -u_{a_T} + \log \sum_{k=1}^{|U|} \exp(u_{a_k}) \tag{7}
$$

We can more clearly see that minimizing this loss function will optimize the embedding vectors, such that the score of the target asset u_{a_T} will be high when in the correct context. As a result, assets that commonly occur in each other's context sets will have similar representations in the latent space.

3.3 Noise Reduction Techniques

Financial return datasets are recognized for their inherent noisiness. Recognizing this, our approach incorporates two techniques aimed at preventing this noise from degrading target:context sets.

Initially, we introduce a weighting mechanism based on overall distributional co-occurrence, whereby the hidden layer representation $\mathbf{h}$ is no longer a simple average; instead, it is computed as a weighted mean. Each individual asset embedding, $\mathbf{v}_{a_j}$, gets scaled by a weighting factor. This factor captures the frequency with which a given context asset, a_j, co-occurs in context with the target asset, a_i, across all time periods in the dataset. This is formalised in Equation (8):

$$
w_{i,j,t} \propto \beta_{i,j} = \frac{1}{T} \sum_{t=1}^{T} \mathbb{1}\Big(a_j \in \mathcal{S}(a_i, t)\Big) \tag{8}
$$

Here, $\beta_{i,j}$ signifies the co-occurrence rate and $\mathbb{1}$ is the indicator function. The scaling constant $k_{i,t}$ ensures that when summed across all context assets, the weights equal one.

$$
w_{i,j,t} = k_{i,t} \cdot \beta_{i,j} \quad : \quad k_{i,t} = \left(\sum_{j: a_j \in \mathcal{S}(a_i, t)} \beta_{i,j} \right)^{-1} \tag{9}
$$

Next, to filter out instances of little informational value, we introduce a criterion based on the variability of daily asset returns. Notably, a significant portion of daily returns are clustered around zero, suggesting minimal stock price fluctuations. To focus on instances where meaningful price variations occur, we discard any target:context pair from our training data where the target asset's return, is within the interquartile range (IQR) of returns for that day. This ensures our model trains primarily on cases where the target asset's movement is notably different from the market's typical behavior, thus emphasizing more informative data points.

4 Experimental Design & Tailored Context Curation

In this section, we introduce a series of experiments to evaluate our asset embedding approach. Our first step involved an initial qualitative assessment to ascertain that relevant information was indeed being captured by the embeddings. We visualize the embedding space in a two-dimensional representation, scrutinizing assets with high embedding similarity, and observing instances where assets, despite differing in their GICS (Global Industry Classification Scheme) labels, displayed pronounced similarity in the latent space. This was further substantiated by implementing an industry sector categorization task and comparing our results with standard benchmarks. In the final stage, we apply the learned embeddings to inform a diversification strategy which achieves lower out-of-sample portfolio volatility than traditional approaches. Additionally, we propose two additional strategies for curating context assets, specifically aimed at the downstream task of hedging.

4.1 Data

All of the experiments use a publicly available dataset of daily pricing data for 611 US stocks during the period 2000–2018. In addition to daily returns, each stock is also associated with a *sector* and *industry* classification label from the GICS. The sector label captures the primary business domain of the respective company—with categories like Finance, Health Care, and Technology, among others, while the industry label provides a finer-grained classification. For example, a stock in the Technology sector may have Computer Software as its industry label, to contrast it with another Technology stock in the Electronic Components industry. Unless otherwise stated, the embeddings used in all experiments were generated using a context size of 3 and an embedding dimension of 20. Results are reported both with and without noise reduction techniques.

4.2 Sector Classification

Investing in the stock market is inherently laden with complexities, given the plethora of concealed variables and unpredictable incidents influencing stock prices. When allocating capital, investors are confronting both market-related risks (systematic) and specific asset risks (idiosyncratic). The popularity of exchange-traded funds (ETFs) has grown because of their ability to help investors dilute idiosyncratic risks, via diversification, in a cost-effective manner.

However, determining portfolio constituents is a major challenge for ETF and index providers. The choice is rife with subjectivity, especially when the ETFs are designed to reflect specific market sectors. Take Amazon for example: despite its classification as consumer discretionary under the GICS, arguments can be made for its presence in consumer discretionary, technology, or even consumer staples ETFs. Beyond the application of ETFs and indices, categorizing stocks into distinct sectors is vital for a range of financial and economic endeavors: measuring economic activity, identifying peers and competitors, quantifying market share and bench-marking company performance [17].

The first number of experimental results we present in Section 5 are motivated by the sector classification problem. We initially present a low-dimensional visualization of the embeddings space, showing clustering of asset embeddings that tends to align with their GICS sector label. Next, we explore the top-k nearest neighbours of a sample selection of assets, as well as discussing interesting cases involving asset pairs with high embedding similarity which have differing GICS labels. Finally, we quantify our findings with the results of a classification task that takes embeddings as inputs and attempts to predict sector membership for each asset.

4.3 Volatility Reduction: Tailored Context Curation

Computational research in financial markets has traditionally been skewed towards predicting asset returns. However, Markowitz emphasized that there is more to capital allocation: it's not just about forecasting returns and risks for individual assets but also about determining the right capital distribution considering their relationships [15]. This latter aspect, vital for many in the industry, hasn't been as extensively explored. Algorithmic traders, for instance, must navigate a complex terrain. Their challenge isn't limited to identifying high-returning assets; they also need to adeptly group these assets in a way that neutralizes each asset's unique risk. We propose that leveraging the learned embedding space can offer a more nuanced understanding of asset relationships beyond conventional metrics, and thus allow for better capital allocation and out-of-sample portfolio performance.

We set up a simplified long-only scenario where an investor, holding a stock, seeks another complementary stock to best offset risk. While conventional wisdom might lean towards the least correlated asset as a hedge, we compare a strategy of choosing assets that are maximally dissimilar in the embedding space. With this specific use case in mind, we propose two new approaches to context stock curation. The first leverages raw returns in a similar way to before and the second strategy transforms returns into rolling volatility before computing similarity. We note that both the proposed tailored context selection strategies retain the desirable self-supervised characteristic by also not requiring any labelled data.

Returns Offset Context Curation Hedging stands as a cornerstone in portfolio management, where the goal is to mitigate potential losses that might arise from undesirable price movements of an asset, by also investing in an asset with negatively correlated returns. To facilitate this, the ideal hedge asset should exhibit returns that effectively negate or offset those of the target asset.

By reformulating the context selection criterion, we focus on capturing this characteristic. Instead of identifying context assets that mimic the returns of the target, our revamped approach seeks assets whose returns, when combined with the target asset's returns, tend to cancel each other out. Formally, given a target asset a_i at time t, the context assets are chosen such that the sum of returns $r_t^{a_i} + r_t^{a_j}$ are as close to zero as possible. Mathematically, we capture this by choosing context assets a_j that minimize the absolute value of their combined returns, $\min_{j \neq i} |r_t^{a_i} + r_t^{a_j}|$.

In the resulting embedding space, an asset that emerges as *highly similar* to the target likely serves as an effective hedge. This alternative context selection process, thus, caters explicitly to the requirements of hedging, ensuring the embeddings are not only information-rich but also purpose-driven. Additionally, for this strategy, both previously proposed noise reduction techniques (Section 3.3) are equally applicable.

Volatility Co-Occurrence Context Curation As touched on at the beginning of this section, understanding and managing volatility is paramount in capital allocation to manage risk. Thus, we investigate whether embedding representations directly derived from volatility co-occurrences can provide an edge in hedging strategies. To capture this, we choose context stocks based on the similarity in volatility. The volatility of an asset a_i at time t over the previous n time points can be represented as $\sigma_{t,n}^{a_i}$. Building upon the original context selection strategy, we aim to find assets with concurrent volatility patterns. Specifically, for a given asset a_i at time t, the context assets are selected based on the similarity in their volatility, quantified as the difference, $|\sigma_{t,n}^{a_i} - \sigma_{t,n}^{a_j}|$. This selection strategy is identical to the approach outlined in Section 3.1 except the series of returns $\mathbf{r}_{a_i}$ is swapped for a rolling series of volatility $\boldsymbol{\sigma}_{a_i,n} = \sigma_{n,n}^{a_i}, \ldots, \sigma_{T-n,n}^{a_i}$. To be more precise, given a time series of returns $\mathbf{r}_{a_i}$, the rolling volatility at time t for an asset a_i over a window of n periods is computed as the standard deviation of the last n returns up to time t:

$$\sigma_t^{a_i} = \sqrt{\frac{1}{n} \sum_{k=t-n+1}^{t} \left(r_k^{a_i} - \mu_{t,n}^{a_i}\right)^2} \tag{10}$$

where $\mu_{t,n}^{a_i}$ is the mean return of asset a_i over the same n periods.

While this selection may not directly offset the target's returns, as we previously engineered, once the learning algorithm has converged, assets with *high dissimilarity* in the embedding space will tend not to experience periods of high volatility during the same periods. This is desirable behaviour for a risk-averse investor.

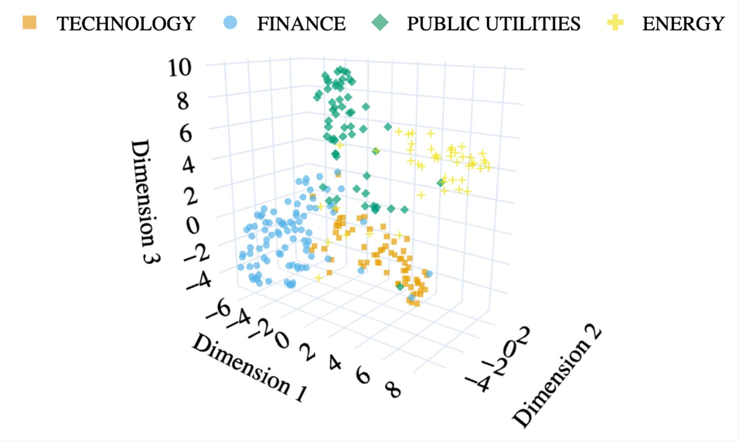

Fig. 2. 3-Dimensional t-SNE Visualization of Asset Embeddings Colored by Business Sector

5 Experimental Results

5.1 Visualizing the Embedding Space

Visualizing an embedding space in two or three dimensions is an intuitive way to verify that the learned representations are encoding useful information. One of the notable approaches to visualize such data in a low-dimensional space is t-SNE (t-distributed Stochastic Neighbor Embedding). Figure 2 showcases a 3D t-SNE visualization of the embeddings for assets across four of the most populated business sectors in the dataset: Energy, Finance, Public Utilities, and Technology. Each point in the plot symbolizes an asset (a company in this case), and they are coloured by industry sector. By design of t-SNE, the proximity of the data points is indicative of the similarity in the higher dimensional space.

A clear clustering of assets, delineated by the sectors, is apparent in Figure 2. Assets within the same business sectors tend to be grouped together, implying similarity between their embeddings. This gives credence to the fact that the embeddings derived from our proposed model architecture, context selection hypothesis and training procedure effectively encode sectoral relationships between assets even though the training data was exclusively composed of historical returns data with no accompanying labels. This highlights the potential of these embeddings to encode nuanced relationships and generalise to other tasks in the same way.

5.2 Top-k Nearest Neighbours

When leveraging the power of the embeddings, a natural expectation is for assets with shared characteristics to be ‘close’ in the embedded space. Such proximity can be gauged through various similarity metrics, with cosine similarity being

Table 1. Examples of Top-3 Nearest neighbors for Given Query Stocks

Query Stock **Sector - Industry**	**3 Nearest Neighbors - Sector - Industry**	**Similarity**
JP Morgan Chase	Bank of America Corp - Finance - Major Bank	0.88
Finance	State Street Corp - Finance - Major Bank	0.82
Major Bank	Wells Fargo & Company - Finance - Major Bank	0.81
Analog Devices	Maxim Integrated - Technology - Semiconductors	0.93
Technology	Texas Instruments - Technology - Semiconductors	0.91
Semiconductors	Xilinx, Inc. - Technology - Semiconductors	0.90
Chevron Corporation	Exxon Mobil - Energy - Oil & Gas	0.89
Energy	BP P.L.C. - Energy - Oil & Gas	0.82
Oil & Gas	Occidental Petroleum - Energy - Oil & Gas	0.78

used in this experiment. Nevertheless, other metrics like Euclidean distance yield similar results and demonstrate a robust embedding space.

Illustrating this, Table 1 highlights the top-3 most similar assets for three diverse companies: JP Morgan Chase, Analog Devices, and Exxon Mobil Corp. The neighbours reveal that the closest assets for each example align with ones intuition and tend to agree in sector and industry label. For instance, the neighbours of JP Morgan, a leading banking institution, is populated with other large US banks. This clustering is particularly validating given that our embeddings solely processed daily returns data without any explicit sectoral or industrial categorization.

In addition to the tangible implications of sector classification for investors discussed in Section 4.2, by zoning in on an assets' nearest neighbors, one could craft rudimentary stock recommendation frameworks. Given an investor's target asset, such a system could curate a list of assets resonating with the historical return patterns of the target. Furthermore, the capability to pinpoint sets of assets that starkly contrast each other augments portfolio diversity, arming investors with the means to shield against abrupt market fluctuations and sector-specific tremors—this will be investigated more in Section 5.5.

5.3 High Similarity Mismatch

In our exploration, as depicted in Figure 2, while the majority of assets from the same sector are clustered closely, certain anomalies emerge. These anomalies, termed "high similarity mismatches," are instances where asset pairs, though originating from diverse sectors, demonstrate a profound similarity in their embeddings.

A case in point is the Lennar Corporation, predominantly classified in the Basic Industries sector with its core in home construction. In Table 2, we identify KB home which exhibits very high embedding similarity with Lennar Corporation and is operationally tied with Lennar through their shared emphasis on Homebuilding. However, notably KB Home is classified under the Capital Goods

Table 2. Examples of stocks with very high similarity that have different sector labels

Stock A	**Stock B**	**Similarity**
Sector - Industry	**Sector - Industry**	
Lennar Corporation	KB Home	0.98
Basic Industries	Capital Goods	
Homebuilding	Homebuilding	
Bristow Group Inc.	Unit Corporation	0.92
Transportation	Energy	
Transportation Services	Oil & Gas Production	
Cirrus Logic, Inc.	Xcerra Corporation	0.93
Technology	Capital Goods	
Semiconductors	Electrical Products	

sector rather than Basic Industries like Lennar Corp. Such observations suggest potential overlaps or ambiguities in the traditional sector labels.

Upon closer examination of the learned embeddings, we encounter the pairing of Cirrus Logic and Xcerra Corporation. Both companies, though designated under separate sectors by the GICS classification, centralize their operations around semiconductors. Similarly, an intersection is visible between Unit Corporations and Bristow Group; both are intertwined with the oil and gas industry, the former specializing in production and the latter in transportation.

These mismatches underscore two crucial insights: first, the potential shortcomings and subjectivities in established sector classifications, and second, the nuanced capabilities of the learned embeddings to discern relationships often overlooked by conventional categorizations. Leveraging these insights, as elaborated on in Section 5.5, opens up the possibility for a nuanced approach to diversification stemming directly from market data.

5.4 Industry Sector Classification

This section aims to quantify the findings from the previous experimental sections—that the embedding space captures the notion of industry sector. To do this, we use the learned embeddings as the features in a supervised multi-class classification problem and compare the output against the established GICS sector classifications to determine accuracy. The level of agreement we can expect between the model and the GICS labels is limited by unpredictable factors inherent in historical returns data as well as inconsistencies in current subjective approaches to stock labelling, as discussed in Section 5.3. We also note that the data exhibits an imbalanced sector distribution, which we counteract by applying upsampling to the training dataset.

Table 3 outlines the performance of the proposed embedding methodology, as well as several time series classification models as baselines, on the sector classification task. The embeddings with both noise reduction techniques achieve the highest accuracy of 60%. In addition, our method provides the added benefit

Table 3. Sector Classification Results

Model	Precision	Recall	F1	Accuracy
Catch22	0.31	0.35	0.31	35%
RBOSS	0.57	0.42	0.45	42%
Shapelet Transform	0.39	0.46	0.40	46%
Time Series Forest Classifier	0.55	0.55	0.53	55%
Canonical Interval Forest	0.57	0.56	0.52	56%
Arsenal	**0.64**	0.58	0.53	58%
Embedding	0.57	0.54	0.55	54%
Embedding + IQR	0.59	0.56	0.56	56%
Embedding + Weight	0.59	0.57	0.57	57%
Embedding + Weight + IQR	0.62	**0.60**	**0.60**	**60%**

of producing learned representations that can serve as features in other asset-related tasks.

Given that there are 11 distinct sector classes[5], achieving this degree of accuracy is commendable. Diving deeper into the results, there is an observable variance in classification accuracy among sectors. The most populated sectors record notably higher performance (F1 scores surpassing 0.9 in some instances), while the less-represented sectors tend to score lower. This variance underscores the potential for enhanced performance with a larger dataset.

The sector segmentation strategy proposed in this study presents a promising avenue for addressing the challenge of inconsistent company categorization—an issue well documented in the literature [2]—in real-world scenarios.

5.5 Out-of-Sample Portfolio Volatility

We now present the results of the hedging experiment, where the goal is to find the best stock to pair with a query stock to minimize risk, which we measure as volatility. To do this, we create a hedged two-asset long-only portfolio for each stock in the dataset, resulting in 611 portfolios, each containing a query stock and optimal hedge stock according to the various baseline and proposed similarity methodologies. For example, in the case of Pearson correlation, the lowest correlated stock would be the optimal hedge. We then simulate each portfolio's out-of-sample performance, recording realized volatility values, and repeat for each baseline and proposed method. This process is detailed in Algorithm 1. The similarity/embeddings are computed/trained on the initial 70% of the data, spanning 2000-2012, and the resulting portfolios are simulated from 2013-2018.

Figure 3 displays the volatility distribution over 611 portfolios for the Pearson baseline, the initial embeddings with IQR noise reduction, the volatility-based

[5] Basic Industries, Capital Goods, Consumer Durables, Consumer Non-Durables, Consumer Services, Energy, Finance, Health Care, Public Utilities, Technology, Transportation

Algorithm 1 Finding Hedged Portfolios and Simulating Realized Volatility

```
Input: List of stocks, similarity function
Output: List of hedged portfolios and their volatilities
for target_stock in stocks do
    min_similarity $\leftarrow \infty$
    selected_hedge $\leftarrow$ None
    for candidate_hedge in stocks do
        if similarity(target_stock, candidate_hedge) $<$ min_similarity then
            min_similarity $\leftarrow$ similarity(target_stock, candidate_hedge)
            selected_hedge $\leftarrow$ candidate_hedge
        end
    end
    # Simulate the performance of the hedged portfolio
end
```

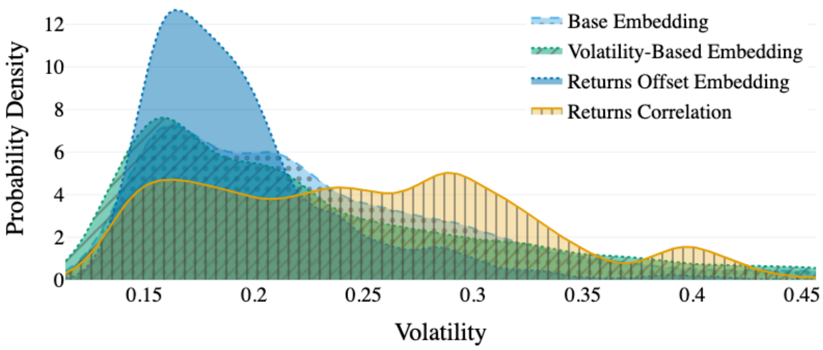

Fig. 3. Volatility Distribution for Different Hedge Approaches

embeddings and returns offset embeddings with IQR noise reduction[6]. The plot shows that the embedding approaches have a higher probability of lower volatility portfolios than the Pearson baseline, which has a longer tail, indicating a greater tendency towards high volatility portfolios. This means that embedding similarity yields better hedge stocks. In particular, the volatility distribution of the returns offset curation with IQR noise reduction leads to a much larger probability density in low-volatility regions.

To ensure the robustness of results, we reran the experiment 100 times where, instead of choosing the single most dissimilar (or most similar in the case of the returns offset embeddings) stock as the hedge stock, we randomly chose one of the 25 most dissimilar for each target stock on each iteration. Table 4 displays the average volatility results over these $100 \times 611 = 61,100$ out-of-sample portfolios. Overall, the proposed hedge embedding approach with IQR noise reduction results in portfolios with the lowest average volatility at 19.5%.

[6] For visual clarity, not all variations and baselines are included in Figure 3. See Table 4 for further details.

Table 4. Portfolio hedging experiment results along with Tukey HSD test indicating significantly lower volatility than Pearson baseline at $\alpha = 0.01$.

Method	**Avg Volatility**	**Significant**
Pearson	23.8%	-
Spearman	24.0%	✗
Geometric	23.9%	✗
Embedding	22.9%	✓
Embedding + Weight	22.8%	✓
Embedding + IQR	21.3%	✓
Embedding + Weight + IQR	21.9%	✓
Offset Context	23.3%	✓
Offset Context + Weight	23.2%	✓
Offset Context + IQR	**19.5%**	✓
Offset Context + Weight + IQR	21.5%	✓
Volatility Context	23.1%	✓

Post-hoc Tukey HSD tests indicate that the volatility in all of the embedding based methods is statistically significantly lower than the Pearson baseline at $\alpha = 0.01$; none of the other baseline approaches generate significantly lower mean volatility over the Pearson approach.

Thus, we have demonstrated that our embedding methodology can enhance hedging strategies compared to standard baselines in a basic two-stock portfolio setting. While real-world portfolios are more complex, this success indicates potential applicability in larger-scale settings. However, further research is needed to understand the nuances of these embeddings, such as the effects of varying context size or embedding dimension.

6 Conclusion and Future Work

In this paper, we have presented a methodology for quantifying the relationships between financial assets by learning embedding representations derived solely from historical returns. The effectiveness of the approach is demonstrated through several qualitative case studies and benchmark comparisons in two key financial tasks: (1) accurately classifying stocks into their respective industry sectors, and (2) constructing portfolios that exhibit statistically significant reductions in volatility compared to traditional baseline methods.

Moving forward, we aim to further assess the potential of this methodology by examining more intricate portfolio management scenarios and incorporating additional datasets. The proposed technique is versatile and applicable to any group of financial assets with accessible pricing information, allowing us to extend our analysis to encompass multiple asset classes beyond equities. Furthermore, we plan to conduct a thorough exploration of the model's parameter space used for learning the embeddings, as well as investigate alternative approaches for generating context stocks.

Acknowledgments: This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant numbers 18/CRT/6183 and 12/RC/2289P2.

References

1. Bishop, C.M., Nasrabadi, N.M.: Pattern recognition and machine learning, vol. 4. Springer (2006)
2. Chan, L.K., Lakonishok, J., Swaminathan, B.: Industry classifications and return comovement. Financial Analysts Journal **63**(6), 56–70 (2007)
3. Choi, E., Bahadori, M.T., Searles, E., Coffey, C., Thompson, M., Bost, J., Tejedor-Sojo, J., Sun, J.: Multi-layer representation learning for medical concepts. In: proceedings of the 22nd ACM SIGKDD. pp. 1495–1504 (2016)
4. Chun, S.H., Ko, Y.W.: Geometric case based reasoning for stock market prediction. Sustainability **12**(17), 7124 (2020)
5. Dolphin, R., Smyth, B., Dong, R.: A machine learning approach to industry classification in financial markets. In: Irish Conference on Artificial Intelligence and Cognitive Science. pp. 81–94. Springer (2022)
6. Dolphin, R., Smyth, B., Dong, R.: A case-based reasoning approach to company sector classification using a novel time-series case representation. In: International Conference on Case-Based Reasoning. pp. 375–390. Springer (2023)
7. Dolphin, R., Smyth, B., Dong, R.: Stock embeddings: Representation learning for financial time series. Engineering Proceedings **39**(1), 30 (2023)
8. Dolphin, R., Smyth, B., Xu, Y., Dong, R.: Measuring financial time series similarity with a view to identifying profitable stock market opportunities. In: ICCBR. pp. 64–78. Springer (2021)
9. Firth, J.R.: A synopsis of linguistic theory. Studies in linguistic analysis (1957)
10. Gabaix, X., Koijen, R.S., Yogo, M.: Asset embeddings. Available at SSRN (2023)
11. Gopikrishnan, P., Rosenow, B., Plerou, V., Stanley, H.E.: Identifying business sectors from stock price fluctuations. arXiv preprint cond-mat/0011145 (2000)
12. Hagerman, R.L.: More evidence on the distribution of security returns. The Journal of Finance **33**(4), 1213–1221 (1978)
13. Ito, T., Camacho Collados, J., Sakaji, H., Schockaert, S.: Learning company embeddings from annual reports for fine-grained industry characterization (2020)
14. Lhabitant, F.S.: Correlation vs. trends: a common misinterpretation (2020)
15. Markowitz, H.: Portfolio selection (1952)
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781 (2013)
17. Phillips, R.L., Ormsby, R.: Industry classification schemes: An analysis and review. Journal of Business & Finance Librarianship **21**(1), 1–25 (2016)
18. Sarmah, B., Nair, N., Mehta, D., Pasquali, S.: Learning embedded representation of the stock correlation matrix using graph machine learning. arXiv preprint arXiv:2207.07183 (2022)
19. Satone, V., Desai, D., Mehta, D.: Fund2vec: Mutual funds similarity using graph learning. In: Proceedings of the Second ACM International Conference on AI in Finance. pp. 1–8 (2021)

20. Sezer, O.B., Gudelek, M.U., Ozbayoglu, A.M.: Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. Applied Soft Computing **90**, 106181 (2020)
21. Sharma, A., Bhuriya, D., Singh, U.: Survey of stock market prediction using machine learning approach. In: 2017 International conference of electronics, communication and aerospace technology (ICECA). vol. 2, pp. 506–509. IEEE (2017)
22. Vamvourellis, D., Toth, M., Bhagat, S., Desai, D., Mehta, D., Pasquali, S.: Company similarity using large language models. preprint arXiv:2308.08031 (2023)
23. Yi, Z., Xiao, T., Ijeoma, K.O., Cheran, R., Baweja, Y., Nelson, P.: Stock2vec: An embedding to improve predictive models for companies. arXiv preprint arXiv:2201.11290 (2022)