Self-Supervised Prototype Representation Learning for Event-Based Corporate Profiling

Zixuan Yuan[1], Hao Liu[2*], Renjun Hu[3], Denghui Zhang[1], Hui Xiong[1*]

[1]Rutgers University
[2]Business Intelligence Lab, Baidu Research
[3]Alibaba Group
{zy101, denghui.zhang, hxiong}@rutgers.edu, liuhao30@baidu.com, renjun.hrj@alibaba-inc.com

Abstract

Event-based corporate profiling aims to assess the evolving operational status of the corresponding corporate from its event sequence. Existing studies on corporate profiling have partially addressed the problem via (i) case-by-case empirical analysis by leveraging traditional financial methods, or (ii) the automatic profile inference by reformulating the problem into a supervised learning task. However, both approaches heavily rely on domain knowledge and are labor-intensive. More importantly, the task-specific nature of both approaches prevents the obtained corporate profiles from being applied to diversified downstream applications. To this end, in this paper, we propose a *Self-Supervised Prototype Representation Learning* (SePaL) framework for dynamic corporate profiling. By exploiting the topological information of an event graph and exploring self-supervised learning techniques, SePaL can obtain unified corporate representations that are robust to event noises and can be easily fine-tuned to benefit various down-stream applications with only a few annotated data. Specifically, we first infer the initial cluster distribution of noise-resistant event prototypes based on latent representations of events. Then, we construct four permutation-invariant self-supervision signals to guide the representation learning of the event prototype. In terms of applications, we exploit the learned time-evolving corporate representations for both stock price spike prediction and corporate default risk evaluation. Experimental results on two real-world corporate event datasets demonstrate the effectiveness of SePaL for these two applications.

Introduction

Event-based corporate profiling aims to assess the evolving operational status of the concerned corporate from its event sequence. Instead of static corporate overview, it incorporates dynamic event information to capture the evolving business circumstances, which enables investors to exploit the price inefficiencies around events. Thus, event-based corporate profiling could be beneficial to a wide range of financial services, such as corporate budget planning, corporate bond credit ratings, and event-driven asset investing.
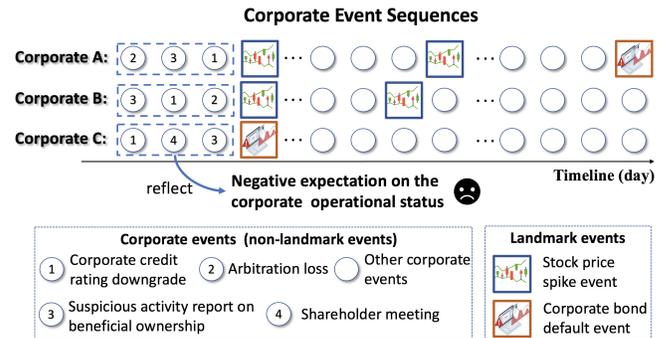
Figure 1: Illustrative example of corporate event sequences.

Prior studies on event-based corporate profiling can be roughly categorized into two classes: (i) *Traditional financial methods*, which implement case-by-case empirical analysis on substantial events (Richardson, Taylor, and Wright 2014), and (ii) *Learning-based methods*, which achieve automatic profile inference by formulating the problem into various supervised learning tasks such as stock price prediction (Ding et al. 2016) and corporate default risk evaluation (Yeh, Wang, and Tsai 2015). However, both types of approaches rely to a large extent on domain knowledge and are inevitably labor-intensive. More particularly, their task-specific nature prevents the obtained corporate profile from being applied to various application scenarios.

Indeed, the recent emergence of prototype learning and self-supervised learning provides great potentials for us to improve existing literature, in terms of both dependence of domain knowledge and generalization of profile. We detail our research insights from the following two perspectives.

First, real-life event sequences are usually characterized by their high degree of stochasticity and obscurity. Inspired by event trend analysis (Raj and Musgrave 2009), we find that the representative event subsequences can explicitly reflect the corporate status, but do not always strictly appear; instead, they may be delivered in various permutations and accompanied by noise events. For instance, in Figure 1, an event set $\{1, 2, 3\}$ is exactly repeated in different permutations of subsequences to express negative expectations on

the operational status for corporates A and B, while corporate C breaks such permutation-invariant property with noise event 4. Moreover, the support of these useful subsequence patterns decreases drastically with the growing cardinality of event subsequences. A promising solution to the above issues is to utilize the concept of prototype learning (Schmidt et al. 2001), which essentially obtains a refined representation of the original sequence by clustering single sequence items into abstract, interpretable prototypes. It allows us to learn a unified representation of event sequence with extracted event sets (i.e., prototypes) that are delivered in the form of meaningful subsequences, and highlight the evolving operational status for corporate profiling. While recent studies have successfully applied this concept into sequence modeling (Liu et al. 2016; Zhang et al. 2020b), most of them adopted simple regularization schemes to quantify the event correlations, which reduces the consistency of prototype selections from long event sequences.

Second, we notice that in lack of domain-expert guidance, the majority of non-landmark corporate events only reveal immaterial, neutral information w.r.t. corporate situation. According to the large-scale data analysis through Wind Data Service[1], the landmark events only account for less than 8% of all event data, which results in the absence of annotated data for effective supervised training. We therefore require a self-supervised learning paradigm to extract high-quality training signals from event sequences for dynamic corporate profiling.

Along these lines, in this paper, we propose a *Self-Supervised Prototype Representation Learning* (SePaL) framework for event-based corporate profiling. Our major contributions are summarized as follows:

- We first introduce the temporal skeletonization technique to embed event co-occurrence patterns into a graph structure. To reduce the stochasticity and obscurity of event sequences, we develop a graph-based co-occurrence smoothing operation to learn the event representations, which are further used to infer the initial cluster distribution of a set of noise-resistant event prototypes.

- We further construct four permutation-invariant self-supervision signals to guide the event prototype representation learning, in order to maintain the consistency of prototype selections.

- We finally utilize the learned prototypes to generate the time-evolving corporate representations and exploit them for various applications, including stock price spike prediction and corporate default risk evaluation. The corresponding experimental results demonstrate the effectiveness of proposed SePaL.

Related Work

Event-based Corporate Profiling. Unlike traditional profiling tasks such as user profiling (Wang et al. 2019; Gu et al. 2020), event-based corporate profiling receives little attention from prior studies in the machine learning field. Moreover, the majority of them put emphasis on profiling

corporates for specific downstream tasks (Ding et al. 2016; Li et al. 2020b; Xu et al. 2020; Lin et al. 2017). For instance, Li et al. (2020b) proposed an event-driven sequential neural network to aggregate multi-source information of market fundamentals and news articles for stock price prediction. However, we argue that the tight coupling with concrete tasks narrows down the applicable values of the learned event representations in dynamic corporate profiling.

Prototype Learning. The intuition of prototype learning is to cluster single items into abstract, interpretable prototypes for better pattern recognition (Liu et al. 2016; Zhang et al. 2020b). As an unsupervised learning exemplar, temporal skeletonization (Liu et al. 2016) proposes to form a series of abstract prototypes that cluster and represent the sequence items with similar co-occurrence patterns via an undirected sequential graph. Its major advantages include proactively reducing the sequence cardinality with extracted prototypes, and uncovering the significant high-hierarchy structures inside sequences. However, due to its low complexity of graph learning, such method has difficulty in fast convergence when dealing with long, noisy event sequences.

Self-Supervised Representation Learning. As an important subarea of unsupervised learning, self-supervised representation learning intends to learn the useful feature embeddings through self-supervisory signals that are directly constructed from input data. Such learning mechanism has been successfully applied in many areas, such as sequence modeling (Liu et al. 2016), translation embedding methods (Zhang et al. 2017), job title benchmarking (Zhang et al. 2019) and language modeling (Zhang et al. 2020a). To the best of our knowledge, this is the first study that adopts the self-supervised representation learning for event-based corporate profiling.

Preliminaries

Suppose there are M event sequences $\mathcal{S} = \{S_1, \cdots, S_M\}$ that entail a collection of N distinct corporate events $\mathcal{E}_{[1:N]} = \{e_1, \cdots, e_N\}$. We denote the m-th event sequence as $S_m = \{e_m^1, \cdots, e_m^{t_m}\}$, where t_m is the length of the m-th event sequence, and $m \in \{1, \cdots, M\}$.

Definition 1 *Event Prototype (EP) is defined as an event set that clusters individual events with similar co-occurrence. The purpose of EP is to form the meaningful event subsequences in different permutations, which can represent the evolving operational states of corporates, e.g., the event set $\{1, 2, 3\}$ in Figure 1 is considered as an EP exemplar that appears in different permutations of subsequences, delivering negative expectation on the corporate operational status.*

Problem Statement

This paper studies the problem of profiling companies with corporate event data. Specifically, we formulate the problem as a task of learning dynamic representation of corporate profile from event sequences. Since EP can encapsulate the hierarchical patterns of event sequences that reflect the corporates' real status, this problem is therefore decomposed to (i) group individual events into multiple useful EPs, and

---

[1]https://www.wind.com.cn/en/data.html
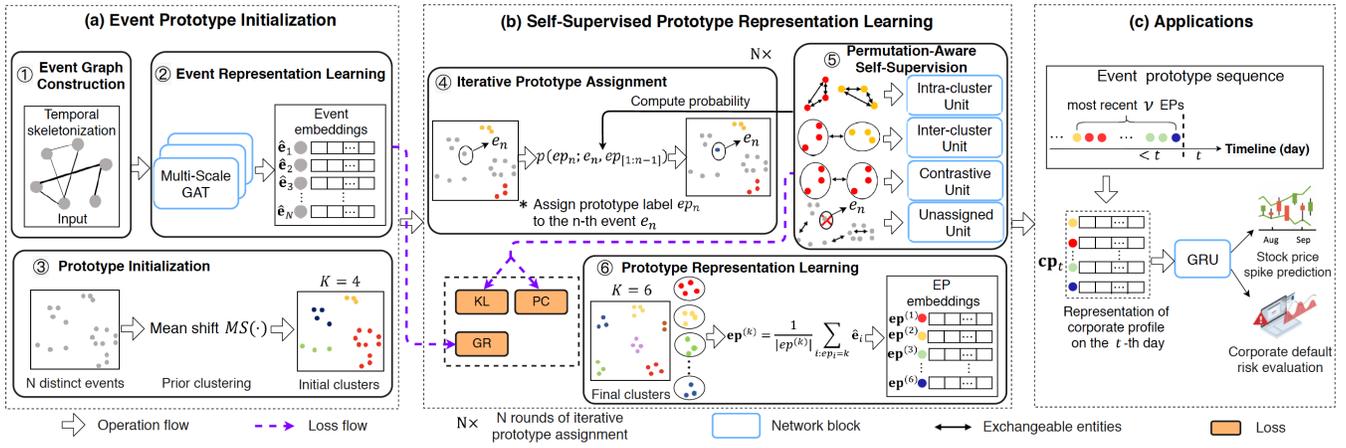
Figure 2: Framework of SePaL.

(ii) preserve the EP patterns in building meaningful event subsequences for dynamic corporate profiling.

Note that, throughout the paper, we use boldface capital and lowercase letters (e.g., $\mathbf{W}$ and $\mathbf{e}$) to denote matrices and vectors, and all $\mathbf{W}_s$ and $\mathbf{b}_s$ are regarded as the learnable weight matrices and bias terms, respectively.

Framework Overview

Figure 2 presents an overview of **SePaL** and its applications for financial services. It includes three major parts: (i) initializing EPs based on event graph learning, (ii) optimizing EP representation via self-supervised learning, and (iii) validating the learned time-evolving corporate representations in two representative financial applications.

In the first part, we propose the *Event prototype initialization* module, in which the *Event graph construction* block first embeds raw event sequences into an event graph through a temporal skeletonization technique, and the *Event representation learning* applies a graph-based smoothing operation to initialize the low-dimensional representations of each event. Then, the *Prototype initialization* block incorporates a clustering approach to derive the initial cluster distribution of EPs based on event representations.

In the second part, we develop the *Self-supervised prototype representation learning* module, where an N-round *Iterative prototype assignment* block is introduced to validate the cluster distribution of EPs. In each round, the *Permutation-aware self-supervision* block applies four types of neural units, i.e., *Intra-cluster unit*, *Inter-cluster unit*, *Contrastive unit*, and *Unassigned unit*, to produce self-supervised permutation-invariant signals for current EPs. Representations of EPs are finalized in the *Prototype representation learning* block, and we use three separate self-supervision losses to regularize the end-to-end training.

In the third part, we incorporate the learned EPs to refine the original event sequence for dynamic corporate profiling, which can be fine-tuned using the existing temporal model for various applications, such as stock price spike prediction and corporate default risk evaluation.

Event Prototype Initialization

We first present the event prototype initialization module to derive the initial distribution of EPs.

Event Graph Construction

Inspired by the temporal skeletonization (Liu et al. 2016), we first create an encoding scheme of event that corresponds to EP label: $ep = \mathcal{F}(e) \in \{1, \cdots, \tilde{K}\}$ such that the sequential variations of events is minimized for all given sequences:

$$\min_{ep \in \{1, \cdots, \tilde{K}\}} \frac{1}{M} \sum_{m=1}^{M} \sum_{\substack{1 \leq i, j \leq t_m \\ |i-j| \leq \delta}} (\mathcal{F}(e_m^i) - \mathcal{F}(e_m^j))^2. \quad (1)$$

Here δ denotes a pre-defined range of local sequence variations. The cardinality $\tilde{K}$ of generated EPs is typically much smaller than that of any original sequence. Furthermore, we assume that given an event sequence $S_m \in \mathcal{S}$, a pair of events s_m^i and s_m^j that are within a close interval are usually correlated with each other, which also applies to their encoded form $\mathcal{F}(s_m^i)$ and $\mathcal{F}(s_m^j)$. Finding the optimal $\mathcal{F}(\cdot)$ w.r.t. objective (1) belongs to an integer programming problem, which has shown to be NP-hard (Liu et al. 2016). We then construct an event graph to relax the integer constraint to real numbers, and encode the closeness of events $e_i, e_j \in \mathcal{E}_{[1:N]}$ into graph structure:

$$\mathbf{A}_{ij} = \frac{1}{M} \sum_{\substack{1 \leq m \leq M \\ e_i, e_j \in S_m}} \Phi_\lambda(|l(e_i, S_m) - l(e_j, S_m)|), \quad (2)$$

where $l(e, S)$ represents the exact location of event e in the sequence S. The range $|l(e_i, S_m) - l(e_j, S_m)|$ of local sequence is smaller than the pre-defined δ. Here, we use $\Phi_\lambda(x) = \exp(-\lambda x)$, a non-increasing function, to quantify the averaged closeness of event co-occurrences reflected by each element $\mathbf{A}_{ij}$ in the graph adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$. However, this simple exponential co-occurrence function still suffers from the scarcity problem of event co-occurrences, such that noise events can hardly be distinguished from useful ones.

Event Representation Learning

To further reduce the stochasticity and obscurity of event sequences, we develop a graph-based co-occurrence smoothing operation to substantiate the event correlations with its neighboring features. Specifically, we apply a multi-scale graph neural network (e.g., GAT (Veličković et al. 2017)) to convolve over the event graph for event encoding:

$$
\begin{cases}
\hat{\mathbf{e}}_i & = \mathsf{Softmax}(\phi^L(\cdots\phi^2(\phi^1(\mathbf{e}_i^1)))), \\
\mathbf{e}_i^{l+1} & = \phi^l(\mathbf{e}_i^l), \ l \in [1, L], \\
\phi^l(\mathbf{e}_i^l) & = \mathbf{W}_1^l \overset{H}{\underset{h=1}{||}} \mathsf{ReLU}(\sum_{e_j \in \mathcal{N}_i} \alpha_{ij}^h \mathbf{W}_2^l \mathbf{e}_j^l) + \mathbf{b}_1^l, \\
\alpha_{ij}^h & = \frac{\exp(\mathsf{ReLU}(\mathbf{W}[\mathbf{e}_i^l||\mathbf{e}_j^l]))}{\sum_{e_q \in \mathcal{N}_i} \exp(\mathsf{ReLU}(\mathbf{W}[\mathbf{e}_i^l||\mathbf{e}_q^l]))},
\end{cases}
\tag{3}
$$

in which $||$ is the concatenate operator. H is the number of attention heads in GAT. $e_q \in N_i$ represents the 1-hop neighbours of node event e_i. $\mathbf{W}_1^l \in \mathbb{R}^{d \times Hd}$, $\mathbf{W}_2^l \in \mathbb{R}^{d \times d}$, $\mathbf{W} \in \mathbb{R}^{2d}$, $\mathbf{e}_i^l, \hat{\mathbf{e}}_i \in \mathbb{R}^{d \times 1}$, where d is the dimension of event embeddings, and $l \in [1, L]$. $\mathbf{e}_i^1$, $\hat{\mathbf{e}}_i$ denote the randomly-initialized and the learned graph representations of event e_i, respectively. Note that $\mathbf{e}_i^l$ represents the l-th scale representation of event e_i. The time complexity of multi-scale GAT is $\mathcal{O}(LHNd^2 + LHN^2d)$.

GR Loss. To ensure that the learned event embeddings can capture the original event co-occurrence patterns stored in event graph adjacency matrix, we introduce the graph reconstruction loss to regularize the representations of events:

$$
\mathcal{L}_{GR} = -\frac{1}{N^2} \sum_{i=1}^N \sum_{j=1}^N (\hat{\mathbf{e}}_i^\top \hat{\mathbf{e}}_j - \mathbf{A}_{ij}).
\tag{4}
$$

Prototype Initialization

The concept of prototype learning has provided the inherent interpretability to highlight prototype patterns from long sequences (Zhang et al. 2020b). We therefore leverage the pre-learned event representations to derive the EPs, with the purpose of reflecting the corresponding corporate status. Specifically, we form the EPs $ep_{[1:N]}$ via the mean shift algorithm $MS(\cdot)$ (Cheng 1995) on events $\mathcal{E}_{[1:N]}$, parametrized by window radius γ and kennel function κ:

$$
ep_{[1:N]} \sim MS(\mathcal{E}_{[1:N]}; \gamma, \kappa),
\tag{5}
$$

in which $ep_i = \mathcal{F}(e_i)$ denotes the EP label which the event e_i is assigned to. Until now, we have established the initial cluster distribution of EP $p_{\gamma,\kappa}(ep_{[1:N]}; \mathcal{E}_{[1:N]})$ based on Eq. (5), as shown in Figure 2(a).

Self-Supervised Prototype Representation Learning

Then, we develop the self-supervised prototype representation learning module to learn the EP embeddings.

Iterative Prototype Assignment

During the early stage of training, the above event representations barely reflect the actual eventive semantics, ending up with the biased prototype distribution $p_{\gamma,\kappa}$. Motivated by

(Pakman et al. 2018), we develop an iterative prototype assignment process to maintain the consistency of prototype selection, via examining the permutation-invariant property possessed by desired EPs. Specifically, we first apply chain rule to reformulate $p_{\gamma,\kappa}$ in the log conditional probability of EP assignments onto events $e_n \in \mathcal{E}_{[1:N]}$, where $n \in \{1, \cdots, N\}$:

$$
p_{\gamma,\kappa}(ep_{[1:N]}; \mathcal{E}_{[1:N]}) = \sum_{n=2}^N \log p(ep_n; e_n, ep_{[1:n-1]}).
\tag{6}
$$

Note that the first event e_1 naturally creates the first EP ep_1, and each of the factor in Eq. (6) can be represented by a generic factor:

$$
p(ep_n; e_n, ep_{[1:n-1]}) = \frac{p(\mathcal{E}_{[1:n]}, ep_{[1:n]})}{\sum_{ep'_n=1}^{K+1} p(\mathcal{E}_{[1:n]}, ep'_{[1:n]})},
\tag{7}
$$

where K denotes the quantity of existing distinct EPs for events $\mathcal{E}_{[1:n-1]}$. The values of Eq. (7) decide on the event e_n of joining any of the K EPs, or forming a new EP cluster. Since the above EP configuration Eq. (6) is difficult to compute directly, we propose a neural representation of Eq. (7) to estimate these factors.

Permutation-Aware Self-Supervision

Critically, the neural representations of Eq. (7) should respect the symmetric structures inside the joint distribution of these EP assignments $p(\mathcal{E}_{[1:n]}, ep_{[1:n]})$. That is, we aim to identify the permutation-invariant properties of generated EPs from: (i) the within-EP events, (ii) the between-EP clusters, (iii) the within-EP events in consecutive assignment rounds, and (iv) the unassigned events. To construct such permutation-equivariant representations $\mathbf{ep}_{[1:n]}$ of EPs, we design four neural units to produce self-supervision signals:

- **Intra-cluster unit** is proposed to maintain the invariant permutations of Eq. (7) under the events $\mathcal{E}_{[1:n]}$ inside a specific EP. For the k-th EP sampled from K existing ones, we define the intra-cluster unit as:

$$
\mathbf{F}_n^{(k)} = \sum_{i:ep_i=k} f(\hat{\mathbf{e}}_i).
\tag{8}
$$

- **Inter-cluster unit** is proposed to maintain the invariant permutations of Eq. (7) under different EPs in the same round. For each of the between-EP invariant $\mathbf{H}_n^{(k)}$, we design the inter-cluster unit as:

$$
\mathbf{H}_n^{(k)} = \sum_{k=1}^K h(\mathbf{F}_n^{(k)}).
\tag{9}
$$

- **Contrastive unit** is proposed to maintain the invariant permutations of Eq. (7) under the same EP in the consecutive assignment rounds. In terms of the contrastive invariant $\mathbf{C}_n^{(k)}$ in the n-th assignment round, we design the contrastive unit as:

$$
\mathbf{C}_n^{(k)} = \sum_{k=1}^K c(\mathbf{F}_n^{(k)}).
\tag{10}
$$

- **Unassigned unit** is proposed to maintain the invariant permutations of Eq. (7) under $N - n$ unassigned events. In terms of the unassigned invariants $\mathbf{G}_n$, we design the unassigned unit as:

$$\mathbf{G}_n = \sum_{i=n+1}^{N} g(\hat{\mathbf{e}}_i). \qquad (11)$$

Note that $f(\cdot)$, $h(\cdot)$, $c(\cdot)$ and $g(\cdot)$ provide fixed-dimensional, permutation-invariant representations of the assigned and non-assigned events, respectively. Self-supervised signals generated by these forms yield arbitrarily accurate approximations for (partially) symmetric functions, which demonstrates at least two benefits: (i) the end-to-end self-supervision protocol allows assignment decision to adaptively improve the representation learning of event, and (ii) the permutation-invariant units justify the validity of cluster distribution $p_{\gamma,\kappa}$, such that the generated EPs will become more resistant to long-sequence stochastic noise. Note that the time complexity of all neural nets is $\mathcal{O}(d^2)$.

Prototype Representation Learning

Once the cluster distribution of EPs is anchored, we finalize the representation vectors for all $\tilde{K}$ EPs, as illustrated in Figure 2(b). Here, to downscale the parameter space, for the k-th EP representation $\mathbf{ep}^{(k)}$, where $k \in \{1, \cdots, \tilde{K}\}$, we simply average the embeddings of its included events as:

$$\mathbf{ep}^{(k)} = \frac{1}{|ep^{(k)}|} \sum_{i:ep_i=k} \hat{\mathbf{e}}_i. \qquad (12)$$

KL Loss. After assigning $n - 1$ corporate events to K existing EPs, where $K < \tilde{K}$, the possibility of allocating the n-th event e_n to each of $K + 1$ possible EPs in Eq. (7) is approximated by:

$$q_\theta(ep_n = k; e_n, ep_{[1:n-1]}) = \frac{\exp(u(\mathbf{R}_n^{(k)}))}{\sum_{k'=1}^{K+1} \exp(u(\mathbf{R}_n^{(k)}))}. \qquad (13)$$

In above, $\mathbf{R}_n^{(k)} = \mathbf{H}_n^{(k)}||\mathbf{C}_n^{(k)}||\mathbf{G}_n$, where $||$ is the concatenate operator, $k \in \{1, \cdots, K+1\}$. The single-layer linear mapping $u(\cdot)$ projects each pair of $(\mathbf{H}_n^{(k)}, \mathbf{C}_n^{(k)}, \mathbf{G}_n)$ into a real-number logit. Note that a new EP is formed with $\mathbf{F}_n^{(k)} = f(\hat{\mathbf{e}}_n)$ when the possibility of EP candidate to be assigned in Eq. (13) peaks at $k = K + 1$. To minimize the distribution divergence between the initial distribution $p_{\gamma,\kappa}$ and approximated one q_θ of EPs, we define the expected KL divergence loss as:

$$\mathcal{L}_{KL} = -\mathbb{E}_{p_{\gamma,\kappa}}[\sum_{n=2}^{N} \log q_\theta(ep_n; e_n, ep_{[1:n-1]})]. \qquad (14)$$

PC Loss. We then introduce the probabilistic contrastive loss (Oord, Li, and Vinyals 2018) to maximize a lower bound on mutual information shared by the same EP in consecutive EP-assignment rounds:

$$\mathcal{L}_{PC} = -\mathbb{E}_{p_{\gamma,\kappa}}[\sum_{k=1}^{\tilde{K}} \log \frac{I(\mathbf{C}_{n-1}^{(k)}, \mathbf{F}_n^{(k)})}{\sum_{k_1=1}^{\tilde{K}} \sum_{k_2=1}^{\tilde{K}} I(\mathbf{C}_{n-1}^{(k_1)}, \mathbf{F}_n^{(k_2)})}], \qquad (15)$$

Description (corporate events)	*Shanghai*	*Shenzhen*
# of events	104,072	161,246
# of corporates	262	287
Description (downstream applications)	*Stock price spike events*	*Corporate bond default events*
# of positive cases	6,060	1,465
# of total cases	153,480	18,415
# of corporates	549	549

Table 1: Statistics of datasets.

in which mutual information between two EPs is calculated as $I(\mathbf{x}, \mathbf{y}) = \exp(\mathbf{x}\mathbf{W}_1\mathbf{y} + \mathbf{b}_1)$. As can be seen, the minimal value for $\mathcal{L}_{PC}$ is negatively proportional to $I(\mathbf{C}_{n-1}^{(k_1)}, \mathbf{F}_n^{(k_2)})$, such that we can maximize the mutual information between same EPs in the consecutive assignment rounds.

Model Training

Overall, we define the end-to-end loss function as the weighted aggregation of the above self-supervised losses (i.e., $\mathcal{L}_{GR}$, $\mathcal{L}_{KL}$, and $\mathcal{L}_{PC}$) with hyper-parameters α, β:

$$\mathcal{L} = \mathcal{L}_{PC} + \alpha\mathcal{L}_{KL} + \beta\mathcal{L}_{GR}. \qquad (16)$$

Applications

In order to evaluate the effectiveness of the learned EPs, we re-encode the original event sequence by replacing each event with its assigned EP label, and obtain the dynamic corporate representations for diverse applications.

Definition 2 *Stock price spike prediction. Given an EP sequence in an openly traded company, the objective is to predict the existence of its day-frequent stock price spike in the next trading day:*

$$f_{PS}(\mathbf{cp}_t) \to \{0,1\}_t, \qquad (17)$$

where the label is set to be 1 if the percentage difference of daily stock price in two consecutive days is greater than 10% ($|p_{t-1} - p_t| / p_{t-1} \geq 0.1$) and 0 otherwise.

Definition 3 *Corporate default risk evaluation. Given an EP sequence in a publicly listed firm, the objective is to forecast the existence of its next corporate bond default event:*

$$f_{CD}(\mathbf{cp}_t) \to \{0,1\}_t, \qquad (18)$$

in which the label is set to be 1 if the corporate bond default event happens on the t-th day and 0 otherwise.

As shown in Figure 2(c), the corporate representation $\mathbf{cp}_t$ on the t-th day is represented by the most recent ν EPs. Furthermore, we introduce two separate temporal models (e.g., GRU (Chung et al. 2014)) as predictive functions $f_{PS}(\cdot)$, $f_{CD}(\cdot)$ to fine-tune $\mathbf{cp}_t$ for the above applications.

Experiments

Experimental Setup

Data Description. We collected three real-world datasets for corporate events, stock price spike events, and corporate

bond default events from Wind-Financial Terminal[2]. They all correspond to 549 publicly traded companies in Shanghai and Shenzhen Stock Exchanges. The first dataset is ranged from May 1, 2013, to June 1, 2020, while the rest are both ranged from May 1, 2017, to June 1, 2020. Since the number of positive cases for real corporate bond default event is very limited (≤ 170), according to (Altman 1998), we expand the original bond default event dataset with the monthly corporate default evaluation (i.e., 'improved', 'unchanged', 'deteriorated') provided by credit rating agency[3], where we treat the 'improved' and 'unchanged' as negative cases, and the 'deteriorated' as positive ones. We leverage LDA (Blei, Ng, and Jordan 2003) to extract event-based keywords, and categorize raw events into 176 distinct types of events as $\mathcal{E}_{[1:N]}$, where $N = 176$. We chronologically order the groundtruth data of stock price spike and corporate bond default events for performance evaluation, where we take the first 60% as the training set, the following 20% for validation, and the rest as the test set. The statistics of the datasets are summarized in Table 1.

Implementation Details. Our model[4] and all ten baselines are implemented with Pytorch, and optimized by the Adam algorithm (Kingma and Ba 2015). All models are trained by a NVIDIA GeForce RTX 2080 Ti. We set the learning rate to 0.0001, the batch size to 32, the dimension d of all representation vectors to 600, the decay weight λ of non-increasing function to 0.9, the scale L of graph convolutional operation to 5, the length of time window ν in EP sequence to 10, the number H of attention heads in GAT to 4, and hyper-parameters α, β to 0.5, 0.3. The kernel function κ and window radius γ of mean shift function $MS(\cdot)$ are set to be a Gaussian kernel and 2, respectively. $f(\cdot)$, $h(\cdot)$, $c(\cdot)$ and $g(\cdot)$ are four separate 3-layer MLPs with Parametric ReLUs, and the number of neurons in each layer is $[600, 400, 200]$, $[200, 200, 200]$, $[200, 200, 200]$, $[600, 400, 200]$, respectively. The range of local sequence variations δ is set to 23, i.e., the average number of events between two consecutive landmarks. Each temporal model consists of a single layer of GRU cell with $\tanh$ activation function. For fair comparison, we fine-tune the model parameters and set the number of training epochs to be 400 for all baselines.

Evaluation Metrics. We adopt $F1$ *score*, *Precision*, and *Recall*, three widely used metrics for evaluation.

Baselines. We compare our full approach with two prototype learning methods, three unsupervised representation learning methods, three rare event prediction methods, and two variants of SePaL:

- **TSM** (Liu et al. 2016) exploits the sequential structures of original sequences by encoding event graph topology into prototypes.
- **TapNet** (Zhang et al. 2020b) proposes a random dimension permutation method to form the attentional prototypes. In the experiments, we use the event sequence as input, and train separate models for two different tasks.

Algorithm	Stock price spike prediction			Corporate default risk evaluation		
	F1	*P.*	*R.*	*F1*	*P.*	*R.*
TSM	0.6034	0.5778	0.6314	0.6473	0.6519	0.6428
TapNet	0.5745	0.5249	0.6345	0.6126	0.5899	0.6371
Sqn2vec	0.5954	0.5771	0.6149	0.5472	0.5606	0.5344
Event2vec	0.4603	0.5065	0.4218	0.5017	0.4438	0.5770
FHVAE	0.5262	0.4938	0.5632	0.5745	0.5118	0.6547
AEDM	0.4037	0.4406	0.3725	0.4473	0.4920	0.4100
SSRDVis	0.5026	0.4533	0.5639	0.5527	0.5309	0.5764
VAE-sim	0.5341	0.4966	0.5777	0.5941	0.5562	0.6375
SePaL-G	0.6460	0.6125	0.6834	0.7051	0.6837	0.7279
SePaL-P	0.6242	0.5802	0.6864	0.6749	0.6629	0.6873
SePaL	**0.6894**	**0.6535**	**0.7295**	**0.7365**	**0.7026**	**0.7738**

Table 2: Overall performance.

- **Sqn2vec** (Nguyen et al. 2018) encodes the sequential patterns into low-dimensional vectors, and applies a gap constraint to obtain discriminative patterns. In the experiments, corporate event sequence is treated as model input.

- **Event2vec** (Hong et al. 2017) develops a sample generator to conduct probabilistic walks on event-based graphs for event representations. In the experiments, we adopt the identical operations in Sqn2vec.

- **FHVAE** (Hsu, Zhang, and Glass 2017) presents an unsupervised variational autoencoder that learns disentangled and interpretable representations from sequences. In the experiments, we use the event graph as input and apply the same representations for both applications.

- **AEDM** (Arora, Sun, and Wang 2019) introduces an embedding-based method to perform classification with data imbalanced over different classes using triple header hinge loss. In the experiments, we adopt the identical operations in FHVAE.

- **SSRDVis** (Li et al. 2020a) proposes a hybrid sequential pattern representation learning paradigm to summarize the common sequential patterns and detect the abnormal behaviors. In the experiments, we feed the original corporate event sequence as input, and leverage the sequence location of detected outliers for two applications.

- **VAE-sim** (Hamaguchi, Sakurada, and Nakamura 2019) learns disentangled representations from only low-cost negative samples using different variant and invariant factors. In the experiments, we generate the event sequences as input for both positive and negative labels in two application tasks.

- **SePaL-G** is a variant of SePaL that directly applies the eigenvectors of event graph as event embeddings, without using graph-based co-occurrence smoothing operation.

- **SePaL-P** is a variant of SePaL without applying self-supervised prototype representation learning module.

Overall Performance

Table 2 presents the overall performance comparisons of our model SePaL and all baselines. As can be seen, SePaL, together with its variants, outperforms all other baselines on both datasets using all metrics. Specifically, SePaL achieves

(a) Stock price spike prediction.

(b) Corporate default risk evaluation.
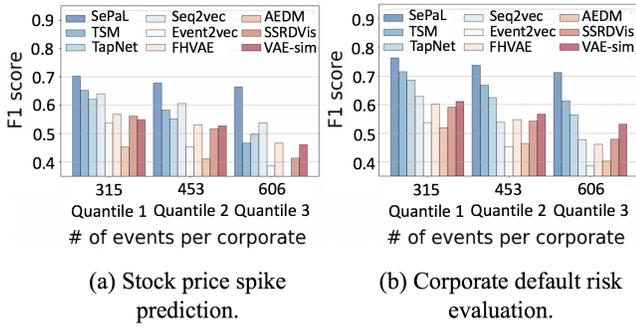
Figure 3: Robustness check.

(a) SePaL.

(b) TSM.
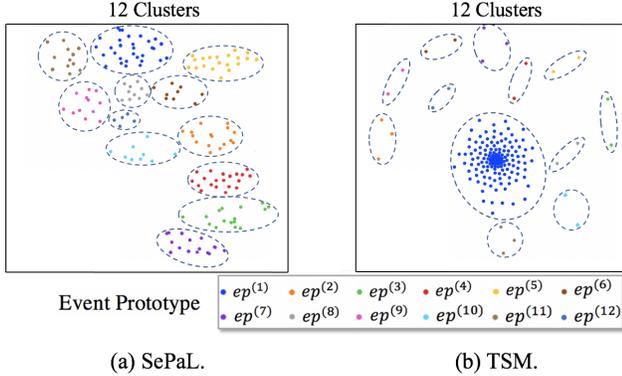
Figure 4: Visualizations of event prototypes.

EP	Corporate Event		Size
	Keywords@top-1	Keywords@top-2	
$ep^{(1)}$	Default Risk Event	Operational Abnormality	31
$ep^{(2)}$	Equity Incentive	Disclosure of Performance	16
$ep^{(3)}$	Key Project	Credibility Evaluation	16
$ep^{(4)}$	Disclosure of Performance	Shareholder Meeting	20
$ep^{(5)}$	Dividends	Seasoned Issuance	20
$ep^{(6)}$	Operating Event	Regulatory Compliance	9
$ep^{(7)}$	Fiscal Policy	Red Warning	16
$ep^{(8)}$	Disclosure of Performance	Amendment of Corporate Articles	10
$ep^{(9)}$	Red Warning	Financial Affair	12
$ep^{(10)}$	Shareholder Meeting	Operating Activity	9
$ep^{(11)}$	Asset Trading	Amendment of Corporate Articles	12
$ep^{(12)}$	Amendment of Corporate Articles	Equity Incentive	5

Table 3: The semantic annotation of EPs.

Path ID	EP Path	Stock Price Spike Exists?	Size (Positive / Negative)
P_1	$ep^{(4)} \rightarrow ep^{(10)} \rightarrow ep^{(8)}$		424 / 96
P_2	$ep^{(1)} \rightarrow ep^{(4)} \rightarrow ep^{(7)} \rightarrow ep^{(8)}$	Yes	267 / 65
P_3	$ep^{(8)} \rightarrow ep^{(9)}$		222 / 58
P_4	$ep^{(5)} \rightarrow ep^{(7)} \rightarrow ep^{(2)}$		2153 / 39
P_5	$ep^{(1)} \rightarrow ep^{(8)} \rightarrow ep^{(12)} \rightarrow ep^{(1)} \rightarrow ep^{(8)}$	No	3870 / 45
P_6	$ep^{(1)} \rightarrow ep^{(9)} \rightarrow ep^{(4)}$		1318 / 15

Table 4: Sequential EP paths in stock price spike prediction.

(8.60%, 7.57%, 9.81%) and (8.92%, 5.07%, 13.10%) improvements beyond the optimal prototype baseline, TSM on both datasets in terms of three metrics respectively. In addition, both prototype learning methods perform relatively better than unsupervised representation learning models and rare event prediction methods, which indicates that EPs may achieve better understandings in discovering event-based sequential patterns. Finally, our full approach SePaL outperforms all the baselines by (14.24%, 12.72%, 15.45%, 15.08%, 13.42%, 16.53%) in average on two datasets. Indeed, the introduction of graph-based co-occurrence smoothing and self-supervised prototype representation learning does generate meaningful EPs, imposing positive influences on dynamic corporate profiling.

Robustness Study

Figure 3 demonstrates the performances of SePaL and baselines on the firms that own different event sequence length. It is notable that fewer event sequence length leads to better performances for all models, possibly because the sequential pattern increases with the dwindling cardinality. Compared with all baselines, SePaL is more stable and achieves excellent performance with longer event sequences. More concretely, with the decrease in event sequence length, the $F1$ performances of SePaL can be improved from 0.6652 to 0.7034 in stock price spike prediction, and from 0.7123 to 0.7637 in corporate default risk evaluation, which indicates its effectiveness in extracting meaningful EPs.

Qualitative Study

Visualization of EP embeddings. In Figure 4, we use the t-SNE algorithm (Maaten and Hinton 2008) to visualize the EP embeddings learned by SePaL and TSM in the form of a two-dimensional map, respectively. Compared to TSM, SePaL can clearly demonstrate the distinctive characteristics possessed by all events, instead of aggregating the majority of them into one gigantic cluster. We also elaborate on the dominant semantic descriptions for each detected EP, as shown in Table 3. Note that the semantic information here is only used to summarize each EP to better interpret our results, but not for the purpose of grouping the events. We also find some interesting observations on the extracted EPs: (i) The nearby EPs tend to be conceptually relevant, e.g., $ep^{(1)}$ and $ep^{(5)}$, where the dividend policy of a corporate is closely related to its default risk event (Sun, Wang, and Zhang 2018); (ii) EPs with similar keywords may not be clustered in the same group. For instance, $ep^{(8)}$, $ep^{(11)}$, $ep^{(12)}$ are all marked with "Amendment of Corporate Articles", but they fall into three separate ESs. This could be explained by the fact that these three EPs are respectively distributed close to $ep^{(6)}$, $ep^{(1)}$, and $ep^{(9)}$, implying that they are characterized by different levels of warnings on corporate performance. Nevertheless, they are still distributed close to each other, mainly because of the same event content, i.e., "Amendment of Corporate Articles". That is to say, EPs exploit more fine-grained sequential correlations via SePaL, while still could be partially consistent with content-based clusters as well.

Analysis of EP Paths. After transforming the original event sequences into EP sequences via SePaL, we obtain corresponding EP paths to uncover the operational status of corporate. For better interpretation, we focus on frequently-appeared EP paths in the corporates with relatively more landmark events (i.e., stock price spike events). Specifically, given a landmark event with its most recent ν EPs, we merge all consecutive and identical EPs to generate the EP path. Some interesting findings could be observed from the frequently-appeared EP paths in Table 4. For instance, the EP path P_1 passes through three EPs, as $ep^{(4)} \rightarrow ep^{(10)} \rightarrow ep^{(8)}$. These corporates may release the latest status of corporate performance, then hold the shareholder meeting for detailed clarification and future planning. This may lead to the occurrence of stock price spike, which is triggered by investors' prompt reactions towards significant variations in enterprise status. All these paths can be grouped into the 'Yes' class, and used to discover the potential risks of the corporates. Moreover, the remaining three paths, P_4, P_5, P_6, are labeled as 'No' class, indicating that these EP sequences do not provide salient patterns for stock spikes.

Conclusion

In this paper, we proposed a self-supervised prototype representation learning (SePaL) framework for event-based corporate profiling. Specifically, we first generated the cluster distribution of event prototypes based on an event graph. Then, we proposed four permutation-invariant supervision signals to learn the noise-resistance prototype embeddings. After that, three self-supervision losses are introduced to guide the entire network update. Later, we incorporated the re-encoded prototype sequence as dynamic corporate profile, which can be fine-tuned with the temporal model for downstream applications. Finally, extensive experiments demonstrated that our SePaL can identify the evolving corporate status to benefit various financial applications.

Acknowledgements

References

Altman, E. I. 1998. The Importance and Subtlety of Credit Rating Migration. *Journal of Banking & Finance* 1231–1247.

Arora, V.; Sun, M.; and Wang, C. 2019. Deep Embeddings for Rare Audio Event Detection with Imbalanced Data. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3297–3301. IEEE.

Blei, D. M.; Ng, A. Y.; and Jordan, M. I. 2003. Latent Dirichlet Allocation. *Journal of machine Learning research* 993–1022.

Cheng, Y. 1995. Mean Shift, Mode Seeking, and Clustering. *IEEE transactions on pattern analysis and machine intelligence* 790–799.

Chung, J.; Gulcehre, C.; Cho, K.; and Bengio, Y. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *arXiv preprint arXiv:1412.3555* .

Ding, X.; Zhang, Y.; Liu, T.; and Duan, J. 2016. Knowledge-Driven Event Embedding for Stock Prediction. In *Proceedings of coling 2016, the 26th international conference on computational linguistics: Technical papers*, 2133–2142.

Gu, Y.; Ding, Z.; Wang, S.; and Yin, D. 2020. Hierarchical User Profiling for E-commerce Recommender Systems. In *Proceedings of the 13th International Conference on Web Search and Data Mining*, 223–231.

Hamaguchi, R.; Sakurada, K.; and Nakamura, R. 2019. Rare Event Detection Using Disentangled Representation Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9327–9335. IEEE.

Hong, S.; Wu, M.; Li, H.; and Wu, Z. 2017. Event2vec: Learning Representations of Events on Temporal Sequences. In *Asia-Pacific Web (APWeb) and Web-Age Information Management (WAIM) Joint Conference on Web and Big Data*, 33–47. Springer.

Hsu, W.-N.; Zhang, Y.; and Glass, J. 2017. Unsupervised Learning of Disentangled and Interpretable Representations from Sequential Data. In *Advances in neural information processing systems*, 1878–1889.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Li, C.; Dong, X.; Liu, W.; Sheng, S.; and Qian, A. 2020a. SSRDVis: Interactive Visualization for Event Sequences Summarization and Rare Detection. *Journal of Visualization* 171–184.

Li, Q.; Tan, J.; Wang, J.; and Chen, H. 2020b. A Multimodal Event-driven LSTM Model for Stock Prediction using Online News. *IEEE Transactions on Knowledge and Data Engineering* .

Lin, H.; Zhu, H.; Zuo, Y.; Zhu, C.; Wu, J.; and Xiong, H. 2017. Collaborative Company Profiling: Insights from an Employee's Perspective. *arXiv preprint arXiv:1712.02987* .

Liu, C.; Zhang, K.; Xiong, H.; Jiang, G.; and Yang, Q. 2016. Temporal Skeletonization on Sequential Data: Patterns, Categorization, and Visualization. *IEEE Transactions on Knowledge and Data Engineering* 211–223.

Maaten, L. v. d.; and Hinton, G. 2008. Visualizing Data Using t-SNE. *Journal of machine learning research* 2579–2605.

Nguyen, D.; Luo, W.; Nguyen, T. D.; Venkatesh, S.; and Phung, D. 2018. Sqn2Vec: Learning Sequence Representation via Sequential Patterns with a Gap Constraint. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 569–584. Springer.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748* .

Pakman, A.; Wang, Y.; Mitelut, C.; Lee, J.; and Paninski, L. 2018. Discrete Neural Processes. *arXiv preprint arXiv:1901.00409* .

Raj, R.; and Musgrave, J. 2009. *Event Management and Sustainability*. Cabi.

Richardson, G.; Taylor, G.; and Wright, C. S. 2014. Corporate Profiling of Tax-Malfeasance: A Theoretical and Empirical Assessment of Tax-Audited Australian Firms. *eJournal of Tax Research* 359.

Schmidt, R.; Montani, S.; Bellazzi, R.; Portinale, L.; and Gierl, L. 2001. Cased-based Reasoning for Medical Knowledge-based Systems. *International Journal of Medical Informatics* 355–367.

Sun, C.; Wang, S.; and Zhang, C. 2018. Corporate Payout Policy and Credit Risk: Evidence from CDS Markets. In *Asian Finance Association (AsianFA) 2018 Conference*.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph Attention Networks. *arXiv preprint arXiv:1710.10903* .

Wang, P.; Fu, Y.; Xiong, H.; and Li, X. 2019. Adversarial Substructured Representation Learning for Mobile User Profiling. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 130–138.

Xu, J.; Zhou, J.; Jia, Y.; Li, J.; and Hui, X. 2020. An Adaptive Master-Slave Regularized Model for Unexpected Revenue Prediction Enhanced with Alternative Data. In *ICDE*, 601–612. IEEE.

Yeh, S.-H.; Wang, C.-J.; and Tsai, M.-F. 2015. Deep Belief Networks for Predicting Corporate Defaults. In *2015 24th Wireless and Optical Communication Conference (WOCC)*, 159–163. IEEE.

Zhang, D.; Li, M.; Jia, Y.; Wang, Y.; and Cheng, X. 2017. Efficient Parallel Translating Embedding for Knowledge Graphs. In *Proceedings of the International Conference on Web Intelligence*, 460–468.

Zhang, D.; Liu, J.; Zhu, H.; Liu, Y.; Wang, L.; Wang, P.; and Xiong, H. 2019. Job2Vec: Job Title Benchmarking with Collective Multi-View Representation Learning. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, 2763–2771.

Zhang, D.; Yuan, Z.; Liu, Y.; Zhuang, F.; and Xiong, H. 2020a. E-BERT: A Phrase and Product Knowledge Enhanced Language Model for E-commerce. *arXiv preprint arXiv:2009.02835* .

Zhang, X.; Gao, Y.; Lin, J.; and Lu, C.-T. 2020b. TapNet: Multivariate Time Series Classification with Attentional Prototypical Network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 6845–6852.