A Novel Document Embedding Approach to Industry Classification

Xiaohang Zhao[1], Xiao Fang[2,*], Jing He[2], Olivia Sheng[3]

[1] School of Information Management & Engineering, Shanghai University of Finance and Economics, Shanghai, China

[2] Lerner College of Business and Economics, University of Delaware, Newark, DE, USA

[3] W. P. Carey School of Business, Arizona State University, Tempe, AZ, USA

∗ Corresponding Author: xfang@udel.edu

Abstract: Industry classification systems (ICSs), which aim to identify economically related firms, play a crucial role in business research and practice. Traditional expert-driven approaches manually design ICSs, leading to limitations such as high maintenance costs and coarse granularity of the identified firm relatedness. To circumvent these limitations, recent research takes an algorithm-driven approach, employing the bag-of-words method to represent firms' 10-K documents and leveraging these representations for identifying economically related firms. While firms' 10-K documents are highly informative for identifying economically related firms, the bag-of-words method is inadequate for representing these documents, as it ignores rich semantic information encoded in word contexts and order, resulting in a less effective ICS. Recent developments in deep learning-based document embedding provide powerful tools for document representation. However, existing document embedding models (DEMs) are not well suited to capture the semantics of 10-K documents, as they fail to account for two key characteristics of 10-K documents: concept switching within a 10-K document and the evolution of a firm's 10-K documents over time. To address these limitations, we propose a novel DEM that introduces two methodological innovations: adaptive gating mechanism and dynamic embedding mechanism, each designed to model one of the key characteristics of 10-K documents. We further develop a new ICS by employing the proposed DEM to capture the semantics of firms' 10-K documents and identifying economically related firms based on the similarities in their document representations. Through extensive empirical evaluations, we demonstrate the superior performance of our proposed ICS in identifying economically related firms, compared to representative existing ICSs as well as ICSs constructed using state-of-the-art DEMs.

Keywords: industry classification system, document embedding, deep learning, financial technology

1. Introduction

Business researchers and practitioners often need to identify peer firms for numerous purposes, such as benchmarking firm performance (Lee et al. 2015), measuring competition faced by a firm (Li et al. 2013), determining executive compensation (Bizjak et al. 2011), understanding a firm's strategic behavior (Cao et al. 2019), and examining the relationship between information technology and firm performance (Bharadwaj 2000). To these ends, various industry classification systems (ICSs) have been developed. An ICS aims to identify economically related firms as peer firms (Lee et al. 2015). Specifically, the economical relatedness among firms can be quantitatively manifested as their contemporaneous correlation in stock returns and a variety of other financial ratios, such as valuation multiples (Bhojraj et al. 2003, Lee et al. 2015).

The central role of ICSs in business research has been widely documented (Weiner 2005, Lee et al. 2015, Phillips and Ormsby 2016, Hoberg and Phillips 2016). For example, Cao et al. (2019) employ an ICS to investigate a firm's strategic behavior in response to its peers' adoption of corporate social responsibility policy. Information Systems (IS) research uses ICSs for a number of purposes, including restricting samples to particular industries (Chung et al. 2019), identifying industry peers (Bharadwaj 2000), constructing industry-level variables (Pant and Sheng 2015, Liu et al. 2020), and controlling for industry effects (Bharadwaj et al. 1999, Tanriverdi 2005). For practitioners, ICSs serve as the foundation for a wide and expanding range of critical applications. For instance, it is common to determine a firm's executive compensation packages based on those of its peers identified by an ICS (Bizjak et al. 2011). Given their importance in business research and practice, substantial efforts have been devoted to developing various ICSs.[1] Existing ICS developments follow either an expert-driven or an algorithm-driven approach, depending on whether an ICS is manually developed by domain experts or automatically constructed by computer algorithms.

An expert-driven ICS relies on a team of specialists to design industry hierarchies, with firms placed at the lowest level in each hierarchy. The Standard Industrial Classification (SIC) system, established by the U.S. Census Bureau, is the oldest and widely used expert-driven ICS (Bhojraj et al. 2003). Other major expert-driven ICSs include the North American Industry Classification System (NAICS), developed through a collaboration between the Canadian, Mexican, and U.S. governments, and the Global Industry Classification Standard (GICS), jointly developed by Standard and Poor's and Morgan Stanley Capital International (Bhojraj et al. 2003, Phillips and Ormsby 2016). Notwithstanding their popularity, expert-driven ICSs have limitations. First, due

[1] Two streams of ICS research exist. One stream, including our study, develops novel and more effective ICSs. The other stream aims at assigning new firms to an existing ICS (e.g., Wood et al. 2017, Zhao et al. 2023). Since these two streams solve different problems—designing a new ICS versus mapping firms to an existing ICS—we concentrate on reviewing the research stream to which our study belongs.

to their reliance on intensive human labor, expert-driven ICSs are costly to develop and maintain, leading to infrequent updates (Hoberg and Phillips 2016). For instance, SIC has not been updated since year 1987, and NAICS is updated every five years (Phillips and Ormsby 2016). Second, expert-driven ICSs do not provide a measure of pairwise firm similarity (Hoberg and Phillips 2016). Consequently, they treat all firms within the same industry of a focal firm as being equally similar to the focal firm, and therefore cannot rank the peers of the focal firm. However, business research and practice could benefit from peer ranking information because it provides more granular control in selecting comparable firms (Li et al. 2013, Hoberg and Phillips 2016, Shi et al. 2016).

In contrast, algorithm-driven ICSs do not suffer from the aforementioned issues. First, algorithm-driven ICSs can be largely automated once they are specified, which lowers the maintenance cost and enables timely updates (Hoberg and Phillips 2016). Second, algorithm-driven ICSs measure similarity at the firm-pair level and rank peers based on their similarities to the focal firm. Therefore, algorithm-driven ICSs allow more granular control in selecting comparable firms and investigating interfirm relationships. Despite their potential, algorithm-driven ICSs have not received as much attention as their expert-driven alternatives. An algorithm-driven ICS widely used in business research and practice is the Text-based Network Industry Classifications (TNIC) developed by Hoberg and Phillips (2016). It examines the business description portion of a firm's annual 10-K report (i.e., Items 1 and 1A of the report) and represents the firm as a bag-of-words vector derived from the textual content in that portion.[2] Pairwise similarities between firms are then computed by applying the cosine similarity function to their corresponding vectors. Hereafter, we refer to Items 1 and 1A in a 10-K report as a *10-K document*. Please see Appendix A for an example of a 10-K document.

According to Hoberg and Phillips (2016), product similarity is core to identifying economically related firms. Public firms are legally obligated to accurately document their up-to-date product-related information in their 10-K documents (Hoberg and Phillips 2016). These documents provide rich product-related information, ranging from product descriptions to market segmentation and risk factors. Given that 10-K documents contain updated, accurate, and rich product-related information, text in these documents is an ideal input for an algorithm-driven ICS.[3] Therefore, the effectiveness of an algorithm-driven ICS hinges on the construction of a firm's representation that

[2] The bag-of-words vector for a firm is a vocabulary-size vector with each element populated by 1 if the business description portion of the firm's 10-K report uses the word represented by that element and 0 otherwise, where the vocabulary consists of all unique nouns and proper nouns in a corpus of business description portions of investigated 10-K reports (Hoberg and Phillips 2016).

[3] Firms offering similar products, operating in similar market segments, and exposing to similar risks, respond similarly to common market shocks. Therefore, text in 10-K documents is an ideal input for discovering the economical relatedness among firms.

can effectively capture the semantics of text in the firm's 10-K document. TNIC proposed by Hoberg and Phillips (2016) employs the bag-of-words method to represent text. However, this method is inadequate for financial textual analysis, because it only considers word occurrences but ignores the rich semantic information encoded in word contexts and orders (Loughran and Mcdonald 2016). For example, consider the following two sentences extracted from the 10-K documents of Walmart and Target:[4]

> *Through innovation, we are striving to create a customer-centric experience that seamlessly integrates our eCommerce and retail stores in an omni-channel offering that saves time for our customers.* (Walmart)
>
> *We operate as a single segment designed to enable guests to purchase products seamlessly in stores or through our digital channels.* (Target)

It is obvious that both sentences stress dual-channel operations, and therefore their semantic similarity is high. However, because of different word choices in referring to the digital and physical channels in these sentences, the bag-of-words method could significantly underestimate their semantic similarity.

Recent developments in deep learning and natural language processing (NLP) provide powerful tools for document representation. The general idea is to embed documents as vectors that preserve their linguistic regularities (e.g., semantics), namely, a document embedding model (DEM) (Le and Mikolov 2014, Li et al. 2015, Sun et al. 2022, Wang et al. 2024). However, existing DEMs are not well suited to represent text in 10-K documents because of two challenges. First, a 10-K document covers a wide variety of concepts, ranging from market competition to risk factors. Not all concepts are emphasized in any given part of a 10-K document. That is, one part may focus on one concept while another part may switch to a different one. Please see Appendix B for an example of concept switching in a 10-K document. None of the existing DEMs can capture the phenomenon of concept switching. Second, a firm files a sequence of annual 10-K documents over time. The evolution of these documents exhibits two key characteristics: variations among these documents resulting from the firm's evolving business activities in response to market changes and time-persistent content shared across the documents reflecting the firm's enduring traits (Andreou et al. 2020). Existing DEMs cannot model the evolution of 10-K documents over time.

To address these challenges, we develop a novel DEM that effectively captures the semantics of 10-K documents. Methodologically, the main differences between our DEM and existing DEMs lie

[4] The first sentence is from Walmart's 10-K report retrieved from https://www.sec.gov/Archives/edgar/data/104169/000010416918000028/wmtform10-kx1312018.htm, and the second is from Target's 10-K report retrieved from https://www.sec.gov/Archives/edgar/data/27419/000002741918000010/tgt-20180203x10k.htm.

in our proposed adaptive gating mechanism and dynamic embedding mechanism. These methodological novelties allow our DEM to effectively model concept switching in a 10-K document and track the evolution of 10-K documents over years. We further develop a new ICS based on our proposed DEM, and we demonstrate through extensive empirical studies that our ICS substantially outperforms representative existing ICSs as well as ICSs constructed using state-of-the-art DEMs in identifying economically related firms.

2. Related Work

Our paper is related to two streams of research: the development of existing ICSs and deep learning-based document embedding models. In the following subsections, we review representative studies from each stream and discuss the main novelties of our study.

2.1. Existing Industry Classification Systems

Existing ICSs can be categorized as expert-driven or algorithm-driven. To construct an expert-driven ICS, domain experts design industry hierarchies, with firms placed at the lowest level in each hierarchy. For example, Figure 1 illustrates a hierarchy from the sector Consumer Discretionary to the industry Automobile Manufacturers and firms in this industry. The SIC system developed by the U.S. Census Bureau was the first major expert-driven ICS (Bhojraj et al. 2003). Since its inception, SIC has been widely used by government agencies as well as business researchers and practitioners. Building on SIC, Fama and French (1997) created the Fama-French Industry Classification System (FF) by aggregating the SIC industries into 48 industry groups. Partly due to its creation by highly influential academics, FF has been widely used in academic research (Phillips and Ormsby 2016). With the objective of replacing SIC, NAICS was constructed through a collaboration between the Canadian, Mexican, and U.S. governments (Bhojraj et al. 2003). In addition to government agencies and academics, business practitioners have also produced a number of ICSs. The most prominent of these is the GICS developed by by Standard and Poor's and Morgan Stanley Capital International (Bhojraj et al. 2003, Phillips and Ormsby 2016). As discussed in Section 1, expert-driven ICSs suffer from issues such as high development and maintenance costs, infrequent updates, and lack of a pairwise firm similarity measure, which can be overcome by algorithm-driven ICSs.

One approach to algorithm-driven ICSs leverages co-occurrence patterns between firms. Sprenger and Welpe (2011) define the similarity between two firms as how frequently they are co-mentioned in twitter messages discussing stocks. This measure reflects investors' perceptions of strategic peers. Motivated by a similar recognition that investors' perceptions are useful in identifying a firm's economically related peers, Lee et al. (2015) utilize the search log data of the SEC's Electronic Data-Gathering, Analysis, and Retrieval (EDGAR) website to measure the similarity between two

Figure 1 An Industry Hierarchy in an Expert-Driven ICS

firms by how frequently they are co-searched on the website. Another approach to algorithm-driven ICSs is to develop a representation for each firm and measure pairwise similarities between firms by applying a suitable similarity function to their corresponding representations. In this vein, Jaffe (1986) employs the patent feature of firms (i.e., patents filed by firms) and represents a firm as a vector of its distribution over patent classes. He then computes pairwise similarities between firms by applying the cosine similarity function to their corresponding vectors. Hoberg and Phillips (2016) target the product feature of firms (i.e., products offered by firms) and develop TNIC by representing a firm as a bag-of-words vector derived from text in the business description portion of its annual 10-K report. They also compute the cosine similarity between firm vectors. Yang et al. (2019) extract financial accounting taxonomies from firms' balance sheets written in the eXtensible Business Reporting Language (XBRL) and represent a firm as a tree in which each node is an XBRL label denoting an accounting concept. Firms' pairwise similarity scores can be computed using the editing distance between their tree representations. However, as recognized by the authors, the information captured by balance sheets is not as rich as 10-K documents (Yang et al. 2019). Additionally, the XBRL approach suffers from limited historical coverage, because its mandatory reporting by the SEC only began in 2009 and the number of errors in early filings was significant (Debreceny et al. 2010). While text in firms' 10-K documents is an ideal input for an algorithm-driven ICS, the bag-of-words method employed in TNIC is inadequate for financial textual analysis because it ignores the rich semantic information encoded in word contexts and order (Loughran and Mcdonald 2016). Recent developments in deep learning and NLP provide tools for representing the semantics of a document, which we review next.

2.2. Deep learning-based Document Embedding Models

A document embedding model (DEM) represents a document as a low dimensional dense vector, with the aim of capturing the semantics of the document (Le and Mikolov 2014, Devlin et al. 2019, Wang et al. 2024). Existing DEMs can be broadly distinguished along two dimensions: 1) supervised vs. unsupervised, depending on whether training documents are labeled, and 2) compositional vs. holistic, depending on whether a document vector is derived from its constituent word vectors. A supervised DEM relies on labels associated with training documents to learn and

optimize its model parameters for predicting these labels (e.g., sentiment labels). Consequently, supervised DEMs generate task-specific document representations intended for their respective tasks, such as sentiment analysis, document classification, and information retrieval (Joulin et al. 2017). In contrast, training documents for an unsupervised DEM are not labeled, and text in these documents is the major source for learning signals. Frequently, a statistical language model is the objective. That is, for every word in each document, an unsupervised DEM is trained to predict that particular word in its context (Le and Mikolov 2014, Pagliardini et al. 2018). As a result, unsupervised DEMs produce task-agnostic and general-purpose document representations that can be used for many downstream tasks. Because the objective of an ICS is to serve as the foundation for a variety of business applications and research settings (Phillips and Ormsby 2016), supervised DEMs are not suitable for representing unlabeled 10-K documents for industry classification. In the following, we focus on reviewing unsupervised DEMs.

Unsupervised compositional DEMs treat the embedding process of a document as a mapping function from its constituent words to the document. A mapping function can be learned, which is done by *learning-based unsupervised compositional DEMs*. Traditionally, it is learned as an encoder in an autoencoder structure (Li et al. 2015, Bowman et al. 2016). Given the word sequence in a document, an encoder is used to derive the document representation by sequentially composing its constituent word vectors via a Recurrent Neural Network, which is then fed into a decoder to reconstruct the original word sequence. Modern approaches predominantly leverage Transformer-based architectures due to their superior capabilities of capturing long-range dependencies and contextual relationships inherent in document semantics (Vaswani et al. 2017, Devlin et al. 2019, Petukhova et al. 2024). Transformer-based DEMs are closely related to Large Language Models (LLMs), such as the GPT series (OpenAI 2023) and the LLaMA series (Touvron et al. 2023). Specifically, these DEMs take a pre-trained LLM as the backbone and fine-tune its parameters for the document embedding task with signals indicating which document pairs are more similar than others (Reimers and Gurevych 2019, Nie et al. 2024, Wang et al. 2024).

Alternatively, the mapping function can be specified directly, leading to a family of *aggregation-based unsupervised compositional DEMs*. In this vein, a common approach is to compute a document vector as the weighted average of its compositional word vectors, which are given by pre-trained models such as GloVe (Pennington et al. 2014) and BERT (Devlin et al. 2019, Sun et al. 2022). To enhance the quality of resulting document vectors, one way is to design more sophisticated weighting scheme. For example, Arora et al. (2017) propose a closed form solution of word vector weights that is optimal under certain assumptions of the text generation process. Another way is to conduct word vector average from multiple perspectives, obtain multiple aggregated vectors, and finally concatenate these vectors as the document representation (Mekala et al. 2017, Ionescu

and Butnaru 2019). Additionally, rather than treating word vectors as fixed inputs for aggregation, it is possible to make them more compatible with the equally-weighted scheme by refining their values with proper learning objectives (Miao et al. 2016, Chen 2017, Pagliardini et al. 2018). Simply averaging word vectors with equal or unequal weights significantly reduces the computational cost of document embedding; however, the resulting document representation might be a coarse and lossy summary of the original document. As a result, document semantics are not well captured by this approach, especially for documents featuring complicated semantic and temporal structures.

Unsupervised holistic DEMs bypass the mapping step and directly learn document vectors as model parameters. A representative unsupervised holistic DEM is the paragraph2vec model (Le and Mikolov 2014), which has two variants: the distributed memory model (PV-DM) and the distributed bag-of-words model (PV-DBOW). For each word in a document, PV-DM predicts its occurrence using its local context (i.e., its neighborhood words) as well as its global context (i.e., the document). In contrast, PV-DBOW only exploits the global context to predict the occurrence of each word in a document. More recent holistic DEMs represent documents in spherical space rather than Euclidean space (Meng et al. 2019, Banerjee et al. 2022). As a result, the document embeddings generated by these DEMs are better suited for computing document similarities. Compared to compositional DEMs, unsupervised holistic DEMs do not need to learn a mapping function, making them more flexible in capturing document semantics.

2.3. Key Novelties of Our Study

Two salient characteristics are featured in 10-K documents. First, a 10-K document encompasses a large number of concepts, and the emphasized concept changes from one part to another in the document. Second, a firm files a series of annual 10-K documents over years. The evolution of these documents is driven by both the intrinsic characteristics of the filing firm and external factors that impact the firm, such as market shocks and changes. The semantics of a 10-K document are determined by the meanings of its constituent concepts and the degree of emphasis of each concept in the document. Therefore, to accurately capture the semantics of 10-K documents, a DEM must effectively model the dynamics of concepts both within a single document (i.e., concept switching) and across multiple documents (i.e., evolution). Our review of existing DEMs shows that none of them consider the notion of concept, nor can they capture concept switching in a document. Furthermore, none of the existing DEMs can model the evolution of 10-K documents filed by a firm.

We propose a novel DEM to address the aforementioned limitations of existing DEMs. Specifically, we develop an adaptive gating mechanism that enables our DEM to effectively model concept switching within a document. We also design a dynamic embedding mechanism that allows our DEM to track the evolution of 10-K documents filed by a firm over time. Built upon our proposed DEM, we develop a new ICS that can effectively identify economically related firms.

3. Method

In this section, we present a novel document representation method tailored to the need of industry classification, namely the Adaptive and Dynamic Document Embedding (ADDE) model. We first introduce notation and a general strategy for unsupervised document embedding in Section 3.1. Next, we propose the two building blocks of ADDE in Sections 3.2, which constitute the main methodological novelties of our study. Section 3.3 summarizes the ADDE model and presents the ICS built on it.

3.1. Preliminaries

A corpus $\mathcal{C}$ is a set of documents $\{\mathcal{D}_1, \mathcal{D}_2, ..., \mathcal{D}_{|\mathcal{C}|}\}$ indexed by $d = 1, 2, \cdots, |\mathcal{C}|$, where $\mathcal{D}_d$ is the dth document and $|\mathcal{C}|$ is the number of documents in the corpus. Because a document can be uniquely identified by its index, we use d and $\mathcal{D}_d$ interchangeably to denote a document. A document d is represented as a sequence of words, written as $< w_1, w_2, \cdots, w_{|d|} >$, where $|d|$ is the length (number of words) of the document and w_t is the word at position t. The local context of w_t, denoted as c_t, is defined as the sequence of words surrounding w_t, i.e., $c_t =< w_{t-n_{\text{ws}}}, \cdots, w_{t-1}, w_{t+1}, \ldots w_{t+n_{\text{ws}}} >$, where n_{ws} is the window size for each side of the context. The vocabulary of the corpus, $\mathcal{V}$, is the set of unique words indexed by $w = 1, 2, \cdots, n_w$ in the corpus, where n_w is the number of unique words. We use the notation $v \in R^n$ for a real-valued vector of length n and $W \in R^{m \times n}$ for a real-valued matrix of m rows and n columns. Let $W_{i,:}$ and $W_{:,j}$ respectively represent the ith row and the jth column of matrix W. We use the superscript T to denote the transpose of a vector or matrix. Important notation and acronyms are summarized in Tables 1 and 2, respectively.

Table 1 Notation

Notation	Description
$\mathcal{C}$	A corpus
d, $\mathcal{D}_d$	A document in a corpus
w	A word in a document
c	The local context of word w
$P(w\|c, d)$	The probability of observing word w given its local context c and global context d
v_c	The vector representation of local context c
v_d	The global vector representation of document d
f_j	The vector representation of factor (concept) j
l_d	The global factor loading vector for document d
$l_d^{(c)}$	The locally adapted factor loading vector for document d; see Equation 7
$g(\cdot)$	The gating function that transforms l_d to $l_d^{(c)}$; see Equations 7, 13
$v_d^{(c)}$	The locally adapted vector representation of document d; see Equation 9
$\mathbb{L}(.)$	The factor loading function; see Equation 16

Table 2 Acronyms

Acronym	Full Name
ICS	Industry Classification System
DEM	Document Embedding Model
ADDE	Adaptive and Dynamic Document Embedding

A general strategy for unsupervised document embedding is to derive word and document vectors such that the negative log likelihood objective

$$-\sum_{d\in\mathcal{C}}\sum_{w_t\in d}\log P(w_t|c_t,d) \tag{1}$$

is minimized (Le and Mikolov 2014), where $P(w_t|c_t,d)$ is the likelihood of word w_t given its local context c_t and global context d that contains w_t. The optimization problem is solved by seeking proper vector representations of the target event w and the conditioning event (c,d), denoted by $v_w^{(o)}$ and $v_{c,d}$, respectively, given that the probability mass function $P(w|c,d)$ has the following softmax formulation:

$$P(w|c,d)=\text{Softmax}_w(W^{(o)}v_{c,d})=\frac{\exp(v_{c,d}^T v_w^{(o)})}{\sum_{j=1}^{n_w}\exp(v_{c,d}^T v_j^{(o)})} \tag{2}$$

where $W^{(o)}\in R^{n_w\times n_e}$ is the output word embedding matrix with ${v_w^{(o)}}^T=W_{w,:}^{(o)}$ and ${v_j^{(o)}}^T=W_{j,:}^{(o)}$, and $\text{Softmax}_w(.)$ denotes the w-th entry of the vector returned by the softmax function.[5] Because the softmax formulation is mainly characterized by the numerator, we simplify the notation with the "proportional to" relationship. We have,

$$P(w|c,d)\propto\exp(v_{c,d}^T v_w^{(o)}). \tag{3}$$

On the one hand, we aim to embed d as a vector v_d, which should capture the meaning of document d. On the other hand, what can be directly learned is $v_{c,d}$, which is jointly determined by the local context c and the global context d. Therefore, it is critical to specify a proper relationship between v_d and $v_{c,d}$ such that v_d is promoted to store the semantics of document d. To this end, we propose a novel adaptive gating mechanism in Section 3.2.1, which is motivated by the phenomenon of concept switching in a 10-K document. Moreover, it is important to model the evolution of 10-K documents filed by a firm over years. Accordingly, we develop a novel dynamic embedding mechanism in Section 3.2.2.

[5] The softmax function transforms an arbitrary real-valued vector into a probability distribution. Let $s\in R^n$ be the input vector and $p\in R^n$ be the output vector. Formally, the softmax function is defined as $p=\text{Softmax}(s)$ with $p_i=\exp(s_i)/\sum_{j=1}^{n}\exp(s_j)$, where $\exp(.)$ is the exponential function, and p_i and s_i denote the ith element of p and s, respectively.

3.2. Adaptive and Dynamic Document Embedding

Let $\mathcal{C}_i$ be a set of annual 10-K documents filed by firm i:

$$\mathcal{C}_i = \{\mathcal{D}_d^{(i)} | d = 1, 2, \ldots, m_i\}, \tag{4}$$

where $\mathcal{D}_d^{(i)}$ is the dth 10-K document filed by firm i at time $t_d^{(i)}$ and m_i is the total number of 10-K documents filed by the firm within an investigated time window. The 10-K documents in $\mathcal{C}_i$ are ordered chronologically, i.e., $t_1^{(i)} < t_2^{(i)} < ... < t_{m_i}^{(i)}$. The full corpus is constructed by pooling all 10-K documents of all firms; that is, $\mathcal{C} = \cup_{i \in \mathcal{I}} \mathcal{C}_i$, where $\mathcal{I}$ is the set of investigated firms. In this subsection, we describe how to embed annual 10-K documents filed by a firm and therefore drop the superscript (i) to simplify notation. Accordingly, we denote a set of m annual 10-K documents filed by a firm as $\mathcal{C}$:

$$\mathcal{C} = \{\mathcal{D}_d | d = 1, 2, \ldots, m\}, \tag{5}$$

where 10-K document $\mathcal{D}_d$ is filed at time t_d and $t_1 < t_2 < ... < t_m$. As stated previously, we use $\mathcal{D}_d$ and d interchangeably to denote a document.

A 10-K document features heterogeneous concepts such as product description, market segmentation, and risk factor. We represent these concepts as a set of latent factors shared across the corpus. Let $W^{(f)} \in R^{n_e \times n_f}$ be the learnable factor embedding matrix, where n_f is the number of factors. Let f_j be the n_e-dimensional embedding vector for factor (concept) j, $j = 1, 2, \ldots, n_f$. By definition, we have $f_j = W_{:,j}^{(f)}$. We model the embedding vector v_d of document d as a convex combination of the factor vectors:

$$v_d = \sum_{j=1}^{n_f} l_{d,j} f_j = W^{(f)} l_d \tag{6}$$

where $l_{d,j}$ is the loading for factor j in document d, which satisfies $\sum_{j=1}^{n_f} l_{d,j} = 1$ and $0 \leq l_{d,j} \leq 1$ for $j = 1, 2, \ldots, n_f$, and $l_d = [l_{d,1}, \ldots, l_{d,n_f}]^T$ is the factor loading vector for document d.[6] We are now ready to present the two building blocks of our proposed ADDE model: the adaptive gating mechanism for representing a 10-K document from a single year and the dynamic embedding mechanism for learning the evolution of 10-K documents filed by a firm over multiple years.

3.2.1. Adaptive Gating Mechanism Different parts of a 10-K document focus on varying subsets of concepts. For example, one part might focus on concepts concerning market segmentation and market position, whereas another part might focus on concepts related to different risk factors. Therefore, it is beneficial to endow ADDE with the ability to emphasize a particular subset of

[6] Using convex combination to aggregate representations of composing units is a common practice in NLP (Bahdanau et al. 2015, Vaswani et al. 2017, Devlin et al. 2019).

concepts by increasing their corresponding loadings while decreasing the rest, according to the meaning of the local context. Specifically, the local context provides useful guidance for pinpointing the focused-upon concepts. We therefore lay out the framework for transforming a global factor loading vector l_d to its locally adapted version $l_d^{(c)}$ based on the local context c:

$$l_d^{(c)} = g(v_c) \odot l_d = [g_1(v_c) l_{d,1}, g_2(v_c) l_{d,2}, \cdots, g_{n_f}(v_c) l_{d,n_f}]^T. \tag{7}$$

In Equation 7, $\odot$ stands for element-wise multiplication and v_c denotes the vector representation of the local context c, which is computed using a pretrained Transformer-based architecture specified in Appendix C.

The gating function $g(\cdot)$ takes v_c as input and produces a rescaling vector $g(v_c)$ of length n_f, of which each element is denoted as $g_j(v_c) \geq 0$ for $j = 1, 2, \ldots, n_f$. The rescaling vector provides a plan that increases the loadings of factors more relevant to the local context c while decreasing the loadings of less relevant factors. The design of the gating function will be discussed later in this subsection. An element of $l_d^{(c)}$, denoted as $l_{d,j}^{(c)}$, is the locally adapted loading of factor j in document d. By Equation 7, we have

$$l_{d,j}^{(c)} = g_j(v_c) l_{d,j} \qquad \text{for } j = 1, 2, \cdots, n_f. \tag{8}$$

With $l_d^{(c)}$ defined, the locally adapted document representation $v_d^{(c)}$ can be formulated as

$$v_d^{(c)} = \sum_{j=1}^{n_f} l_{d,j}^{(c)} f_j = \sum_{j=1}^{n_f} g_j(v_c) l_{d,j} f_j = W^{(f)} \big[g(v_c) \odot l_d \big]. \tag{9}$$

Like v_d, $v_d^{(c)}$ is also defined as a convex combination of the factor vectors and $\sum_{j=1}^{n_f} l_{d,j}^{(c)} = 1$. We refer to $v_d^{(c)}$ as the local document representation and v_d, defined in Equation 6, as the global document representation. By Equation 9, the local document representation $v_d^{(c)}$ adapts to different local contexts and thus varies from one local context to another within a document. Equipped with $v_d^{(c)}$, $P(w|c,d)$ for ADDE can be defined as

$$P(w|c,d) \propto \exp\big({v_d^{(c)}}^T v_w^{(o)} \big). \tag{10}$$

Modeling both v_d and $v_d^{(c)}$ as a convex combination of the factor vectors allows us to take a probabilistic approach to realizing the local adaption of the factor loadings, which in turn motivates our design of the gating function. Namely, we treat $l_{d,j}$ as $P(j|d)$, the probability of factor j given document d, and $l_{d,j}^{(c)}$ as $P(j|c,d)$, the probability of factor j given local context c and document d. By Bayes' theorem, we have

$$P(j|c,d) = \frac{P(c|j,d)}{\sum_{j'=1}^{n_f} P(c|j',d) P(j'|d)} P(j|d). \tag{11}$$

Recall that c is a sequence of words before and after a target word w. In Equation 11, $P(c|j,d)$ is the probability of word sequence c given that the word sequence is about concept j and appears in document d. According to Logeswaran and Lee (2018), the meaning of a language unit (e.g., semantics of a word sequence) is more important than the surface form of the unit (e.g., a word sequence). They further argue that a language model should focus on the probability of the semantic representation of a language unit rather than the probability of the unit itself. Therefore, we model the probability of the vector representation of a local context rather than that of the word sequence comprising the local context by replacing $P(c|j,d)$ with $P(v_c|j,d)$. Furthermore, substituting $P(j|c,d)$ and $P(j|d)$ in Equation 11 with $l_{d,j}^{(c)}$ and $l_{d,j}$, respectively, Equation 11 becomes

$$l_{d,j}^{(c)} = \frac{P(v_c|j,d)}{\sum_{j'=1}^{n_f} P(v_c|j',d) l_{d,j'}} l_{d,j}. \tag{12}$$

By comparing Equations 8 and 12, the gating function $g(\cdot)$ is given by

$$g_j(v_c) = \frac{P(v_c|j,d)}{\sum_{j'=1}^{n_f} P(v_c|j',d) l_{d,j'}}. \tag{13}$$

To instantiate the gating function given in Equation 13, we need to specify $P(v_c|j,d)$ for $j = 1, 2, \ldots, n_f$. Recall that v_c is the context embedding vector that tracks the meaning of the local context c. Given that the local context is about concept j and is part of document d, v_c reflects the document-specific semantics of concept j. Miao et al. (2016) model a language unit as a multivariate Gaussian distribution and regard a vector drawn from the distribution as reflecting the semantics of the language unit. Motivated by Miao et al. (2016), we represent concept j in document d (i.e., a language unit) as a multivariate Gaussian distribution and treat v_c as a sample drawn from the distribution, given that the local context c is about concept j and is part of document d. The multivariate Gaussian distribution is characterized by the document-concept specific mean vector $\mu_{j,d} \in R^{n_e}$ and diagonal covariance matrix $\Sigma_{j,d} \in R^{n_e \times n_e}$:

$$P(v_c|j,d) = \frac{\exp\left(-\frac{1}{2}(v_c - \mu_{j,d})^T \Sigma_{j,d}^{-1} (v_c - \mu_{j,d})\right)}{\sqrt{(2\pi)^{n_e} |\Sigma_{j,d}|}}. \tag{14}$$

Next, we need to estimate the parameters $\mu_{j,d}$ and $\Sigma_{j,d}$. Recall that concept j is represented by the factor vector f_j and document d is represented by the document vector v_d. Because both $(\mu_{j,d}, \Sigma_{j,d})$ and (f_j, v_d) represent the same concept —specifically, concept j in document d— it is reasonable to anticipate a relationship between them. Therefore, we derive $(\mu_{j,d}, \Sigma_{j,d})$ from (f_j, v_d) using a neural network by following a common practice in the machine learning literature (Bishop 1994, Bowman et al. 2016, Doersch 2016). We refer to the designed neural network as the Gaussian Embedding Neural Network (GENN) due to the fact that it transforms the document-concept representation from its vector form to its Gaussian distribution form. As shown in Figure 2, GENN

takes the concatenation of f_j and v_d as input and produces a vector of length $2n_e$. The first n_e entries of the output vector constitute the mean vector $\mu_{j,d}$. The last n_e entries of the output vector are the logarithm of the diagonal elements of the covariance matrix $\Sigma_{j,d}$. We use the logarithm to ensure that $\Sigma_{j,d}$ is positive definite. Formally, we have

$$\left(\mu_{j,d,1},\ldots,\mu_{j,d,n_e},\log\Sigma_{j,d,(1,1)},\ldots,\log\Sigma_{j,d,(n_e,n_e)}\right)^T=\mathrm{GENN}(f_j\oplus v_d), \tag{15}$$

where $\oplus$ is the vector concatenation operator and $\Sigma_{j,d,(x,y)}$ is the element at row x, column y in matrix $\Sigma_{j,d}$. There is literally no constraint on the structure of GENN except for the dimension of the input and output vectors. For simplicity, GENN is instantiated as a standard multilayer perceptron (MLP) in Figure 2.

In summary, the adaptive gating mechanism enables ADDE to model concept switching by capturing shifts in the concepts emphasized from one local context to another within a document. This is accomplished through the introduction of a novel gating framework, as specified in Equation 7, alongside a new gating function defined in Equation 13.

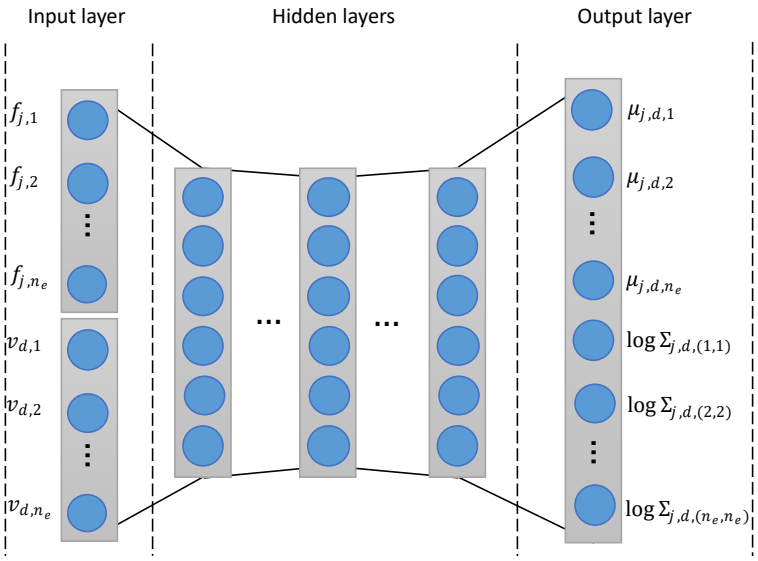


Figure 2 The Gaussian Embedding Neural Network

3.2.2. Dynamic Embedding Mechanism According to Equation 6, the vector representation v_d of a firm's 10-K document filed at time t_d is given by $v_d = W^{(f)}l_d$, $d = 1, 2, \ldots, m$. Intuitively, within-document concept weights, denoted by l_d, evolve over time to reflect changes in business activities conducted by a firm across years. Therefore, to learn the evolution of v_d over time, it is essential to learn the evolution of factor loading vector l_d. To this end, we introduce the factor

loading function $\mathbb{L}(.)$, which takes a scalar t as input and produces a firm's factor loading vector at time t. For example, we have

$$l_d = \mathbb{L}(t_d) \text{ and } l_{d,j} = \mathbb{L}_j(t_d),$$

where $\mathbb{L}_j(t_d)$ represents the j-th element of the vector produced by $\mathbb{L}(t_d)$. Formally, we define factor loading function $\mathbb{L}(t)$ as

$$\mathbb{L}(t) = [\mathbb{L}_1(t), \mathbb{L}_2(t), \ldots, \mathbb{L}_{n_f}(t)]^T = \text{Softmax}\big(W^{(l)}\big[\theta \odot \Gamma(t)\big]\big), \tag{16}$$

where $\Gamma(t) \in R^{2n_t}$ is the firm-specific embedding vector of scalar time t, with n_t being the time embedding dimension, $\theta \in R^{2n_t}$ is the firm-specific coefficient vector, $\odot$ is the element-wise multiplication operator, and $W^{(l)} \in R^{n_f \times 2n_t}$ is the projection matrix that converts a $2n_t$-dimensional vector to a n_f-dimensional vector, which is then normalized to a factor loading vector via the Softmax function. Both θ and $W^{(l)}$ are learnable parameters, and $\Gamma(t)$ will be defined later.

Equation 16 models the evolution of a firm's 10-K documents over years by capturing essential characteristics of the evolution. First, in response to market or industry shocks and changes, a firm adjusts its business activities, leading to variations in its 10-K documents over years. Such variations are captured by the dynamic time embedding vector $\Gamma(t)$. Second, a firm also has corporate traits and activities that are persistent over time. For example, Andreou et al. (2020) find that the market orientation of a firm as measured using its 10-K filings varies little over years. Thus, we use the time-independent vector θ to model the time-persistent aspects of a firm's 10-K documents across years. Third, because of the time-persistent aspects of a firm's 10-K documents over years, these documents are not independent but rather related. By sharing θ among the embeddings of these documents, Equation 16 establishes the relationship among them. Finally, both θ and $\Gamma(t)$ are firm-specific parameters, enabling Equation 16 to model distinct evolution patterns of 10-K documents filed by different firms.

To gain a deep understanding of Equation 16, we show the computation of $\mathbb{L}_j(t)$, the loading for factor j in the 10-K document filed at time t. By Equation 16, we have

$$\mathbb{L}_j(t) = \frac{\exp\left(W^{(l)}_{j,:}[\theta \odot \Gamma(t)]\right)}{\sum_{j'=1}^{n_f} \exp\left(W^{(l)}_{j',:}[\theta \odot \Gamma(t)]\right)} \propto \exp\left(W^{(l)}_{j,:}[\theta \odot \Gamma(t)]\right) \tag{17}$$

where $W^{(l)}_{j,:}$ is the jth row of $W^{(l)}$. Following Xu et al. (2020), we define $\Gamma(t)$ as

$$\begin{aligned} \Gamma(t) = [\cos(\Omega_1 t), \cos(\Omega_2 t), \ldots, \cos(\Omega_{n_t} t)]^T \oplus \\ [\sin(\Omega_1 t), \sin(\Omega_2 t), \ldots, \sin(\Omega_{n_t} t)]^T \end{aligned} \tag{18}$$

where $\Omega = [\Omega_1, \Omega_2, \ldots, \Omega_{n_t}]^T$ is the firm-specific vector of learnable frequency parameters, which specifies how each entry of the time embedding vector should vary over time via the cosine and

sine functions, and $\oplus$ is the vector concatenation operator. By plugging Equation 18 into Equation 17, we obtain

$$\mathbb{L}_j(t) \propto \exp\Big(\sum_{k=1}^{n_t} W_{j,k}^{(l)}\theta_k \cos(\Omega_k t) + \sum_{k=1}^{n_t} W_{j,n_t+k}^{(l)}\theta_k \sin(\Omega_k t)\Big). \tag{19}$$

The term inside exp() is a trigonometric polynomial, which can approximate any arbitrary function of time (Wang et al. 2021). This approximation property allows $\mathbb{L}(t)$, as defined by Equation 16, to model any evolution pattern of a firm's 10-K documents over time.

In summary, the dynamic embedding mechanism learns the evolution of a firm's 10-K documents over years. This is achieved by introducing a novel factor loading function, as specified in Equation 16. This function explicitly models the evolution of 10-K documents by factoring it into two components: one capturing variations among these documents and the other one representing time-persistent aspects shared by these documents.

3.3. ADDE and ADDE-ICS.

The overall architecture of ADDE is depicted in Figure 3. As shown, the dynamic embedding mechanism learns a series of global factor loading vectors $< l_1, l_2, \ldots, l_m >$ from a sequence of 10-K documents $< \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_m >$ filed by a firm across years, by inferring the evolution of these documents via the factor loading function $\mathbb{L}(t)$. Concretely, by Equation 16, we have

$$l_d = \mathbb{L}(t_d) = \text{Softmax}\big(W^{(l)}\big[\theta \odot \Gamma(t_d)\big]\big),$$

where $d = 1, 2, \ldots, m$. For each 10-K document $\mathcal{D}_d$, the adaptive gating mechanism transforms its global factor loading vector l_d to its locally adapted version $l_d^{(c)}$ through the gating framework specified in Equation 7 and the gating function defined in Equation 13:

$$l_d^{(c)} = g(v_c) \odot l_d = g(v_c) \odot \text{Softmax}\big(W^{(l)}\big[\theta \odot \Gamma(t_d)\big]\big),$$

where $d = 1, 2, \ldots, m$. It then uses the resulted local factor loading vector $l_d^{(c)}$ to predict the probability $P(w|c,d)$ of target word w given its local context c and global context $\mathcal{D}_d$ according to Equations 10 and 9:

$$P(w|c,d) \propto \exp\big({v_d^{(c)}}^T v_w^{(o)}\big)$$

and

$$v_d^{(c)} = W^{(f)} l_d^{(c)} = W^{(f)}\Big[g(v_c) \odot \text{Softmax}\big(W^{(l)}\big[\theta \odot \Gamma(t_d)\big]\big)\Big].$$

This prediction is conducted for each word in the corpus. ADDE learns its parameters by optimizing the following objective function:

$$\min_{\Phi(\text{ADDE})} -\sum_{i \in \mathcal{I}} \sum_{d \in \mathcal{C}_i} \sum_{w \in d} \log P(w|c,d). \tag{20}$$

In Equation 20, $\Phi(\text{ADDE})$ denotes the learnable parameters of ADDE and we have

$$\Phi(\text{ADDE}) = \left(W^{(f)}, W^{(o)}, \Phi(\text{GENN}), \Phi(\mathbb{L})\right) \tag{21}$$

where $\Phi(\text{GENN})$ denotes the parameters of GENN, which consist of the weights and biases of the neural network depicted in Figure 2, and $\Phi(\mathbb{L}) = (\{\theta^{(i)}, \Omega^{(i)} | i \in \mathcal{I}\}, W^{(l)})$ represents the parameters of the factor loading function defined by Equation 16, with $\theta^{(i)}$ and $\Omega^{(i)}$ being the parameters specific to firm i.

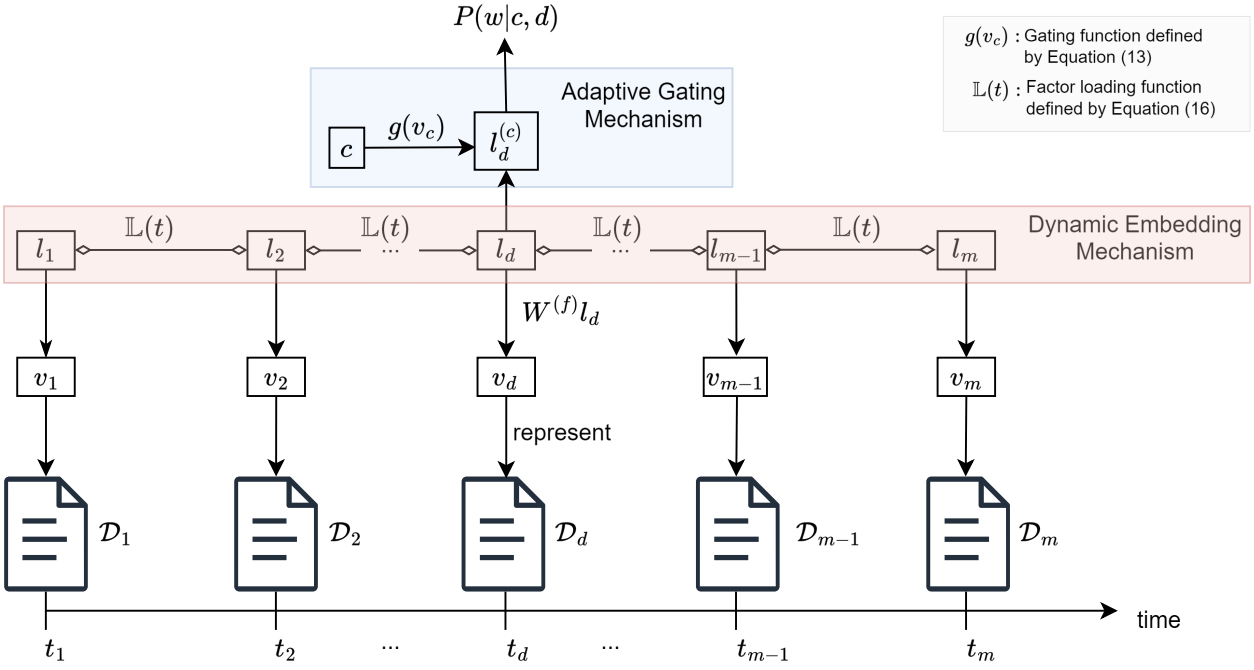


Figure 3 The Adaptive and Dynamic Document Embedding Model

Our proposed ICS, namely ADDE-ICS, is constructed based on ADDE. Given an investigated year Y, we apply ADDE to the corpus of firms' 10-K documents accumulated between year $Y-z+1$ and year Y, where z is the during of an investigated time window. For example, $z=3$ means a 3-year time window. Using the parameters learned by ADDE, each firm's 10-K document in year Y is represented using vector v_d, as computed in accordance with Equation 6. The similarity between two firms' 10-K documents in year Y can then be calculated using the cosine similarity between their corresponding vector representations, where the cosine similarity between two vectors v and v' is given by[7]

$$f_{cos}(v, v') = \frac{v^T v'}{\sqrt{v^T v}\sqrt{v'^T v'}}. \tag{22}$$

[7] Cosine similarity is a commonly used function for measuring the similarity between two vectors, each of which represents a firm (Jaffe 1986, Hoberg and Phillips 2016).

Finally, a focal firm's peer firms are ranked according to their similarities to the focal firm, as calculated using Equation 22. The ADDE-ICS of year Y contains ranked peers for each firm that files a 10-K document in that year as well as similarity scores between each firm and its peers. A description of ADDE-ICS can be found in Appendix D.

4. Empirical Evaluation

We describe the data and benchmark ICSs in Section 4.1, detail the evaluation procedure in Section 4.2, and report the evaluation results in Section 4.3. We then analyze the factors contributing to the performance of our proposed ICS in Section 4.4 and demonstrate the economic value that could be derived from our ICS in Section 4.5.

4.1. Data and Benchmark ICSs

We collected 10-K reports filed between the years 2018 and 2022 from the Stage One 10-X Parse Data provided by Loughran and Mcdonald (2016).[8] Following the practice of Jegadeesh and Wu (2013), if a company had multiple 10-K reports in a year, we retained only the earliest filed report. We then extracted the business description portion (i.e., Items 1 and 1A) from each 10-K report and referred to the extracted portion as a 10-K document. Next, each 10-K document, in the form of a long string, was lowercased and tokenized into a sequence of words. Finally, 10-K documents with fewer than 200 words were removed because there is no substantial content in these documents. The processed 10-K documents were partitioned into yearly corpora based on their filing dates. Table 3 tabulates the summary statistics of the 10-K corpora by year. As shown, the number of 10-K documents in a 10-K corpus, n_d, ranges from 5,340 (year 2021) to 5,889, (year 2018). Taking year 2019 as an example, a 10-K document has an average of 18,185 words (i.e., avg($|d|$)) and contains as many as 109,888 words (i.e., max($|d|$)).

Table 3 Summary Statistics of the 10-K Corpora

Year	n_d	avg($\|d\|$)	max($\|d\|$)
2018	5,889	16,703	83,880
2019	5,741	18,185	109,888
2020	5,343	19,427	117,381
2021	5,340	21,328	110,480
2022	5,636	23,801	109,132

We benchmarked our proposed ICS, ADDE-ICS, against five existing ICSs: GICS, NAICS, SIC, TNIC, and ETNIC. As reviewed before, existing ICSs are categorized as expert-driven or algorithm-driven. We selected GICS, NAICS, and SIC as benchmarks because they are mainstream expert-driven ICSs widely used in business research and practice (Bhojraj et al. 2003). We chose TNIC

[8] This data source standardizes several preprocessing steps of the original 10-K filings downloaded from the EDGAR website. Please refer to https://sraf.nd.edu/data/stage-one-10-x-parse-data/ for the data source and detailed preprocessing steps.

(Hoberg and Phillips 2016) as a benchmark because it is a widely-adopted algorithm-driven ICS (Cao et al. 2019, Gong 2025). We also compared our proposed ICS with ETNIC, the state-of-the-art advancement of TNIC (Hoberg and Phillips 2024). For our evaluation, we retrieved GICS, NAICS, and SIC from the Compustat Company Header History (COMPHIST) dataset provided by Wharton Research Data Services (WRDS) and downloaded TNIC and ETNIC from the repositories maintained by their developers.[9]

In addition, we compared our ADDE-ICS with ICSs constructed using existing deep learning-based DEMs. Specifically, an ICS can be built using an existing DEM by following how ADDE-ICS is constructed: applying the DEM to produce vector representations of firms' 10-K documents, calculating firms' pairwise similarities based on their vector representations, and ranking peer firms for each focal firm based on the similarity scores between the focal firm and its peers. Our literature review in Section 2.2 identifies three categories of existing DEMs applicable to constructing ICSs: learning-based compositional, aggregation-based compositional, and holistic. We chose representative methods in each category to construct ICSs as benchmarks. For the category of learning-based DEMs, we considered proprietary as well as open-source methods that employ LLMs to encode document semantics. In particular, OpenAI's Text-embedding-3-large[10] is a state-of-the-art model for document embedding and is widely recognized as one of the most advanced commercial learning-based DEMs (Nie et al. 2024), while E5-mistral-7b (Wang et al. 2024) is a cutting-edge open-source alternative available to the public. Both models can be used off-the-shelf, and we denoted the ICSs constructed based on them as OpenAI-ICS and E5-ICS, respectively. For the categories of aggregation-based and holistic DEMs, we selected the state-of-the-art method in each category: VLAWE (Ionescu and Butnaru 2019) and JoSE (Meng et al. 2019), and referred to the ICSs built using them as VLAWE-ICS and JoSE-ICS, respectively. Moreover, topic models, such as LDA (Blei et al. 2003), can represent a 10-K document as a vector of topic distribution. An advanced variation of LDA, the Embedded Topic Model (ETM), integrates word and topic embeddings into the LDA framework and has been shown superior performance over LDA (Dieng et al. 2020). Therefore, we also benchmarked an ICS built on ETM, namely, ETM-ICS. Table 4 summarizes the ICSs compared in the evaluation. All ICSs listed in the table are time-varying (e.g., TNIC of year 2019).

Next, we detail the implementations of the DEMs (except for off-the-shelf ones) and the topic model that were used to construct the ICSs in Table 4. All of them produced vector representations

[9] TNIC can be downloaded from https://hobergphillips.tuck.dartmouth.edu/idata/tnic_all_data.zip, with its description available at http://hobergphillips.tuck.dartmouth.edu/tnic_poweruser.htm. The complete version of ETNIC can be downloaded from https://hobergphillips.tuck.dartmouth.edu/idata/Etnic_all_data.zip, and its description is available at https://hobergphillips.tuck.dartmouth.edu/tnic_doc2vec.html.

[10] https://platform.openai.com/docs/guides/embeddings/faq

Table 4 ICSs Compared in the Evaluation

	ICS	Notes
ADDE-ICS	Our Proposed ICS	Built on our proposed DEM, i.e., ADDE
GICS	Benchmark ICS	Mainstream expert-driven ICS
NAICS	Benchmark ICS	Mainstream expert-driven ICS
SIC	Benchmark ICS	Mainstream expert-driven ICS
TNIC	Benchmark ICS	Widely-used algorithm-driven ICS
ETNIC	Benchmark ICS	State-of-the-art algorithm-driven ICS
OpenAI-ICS	Benchmark ICS	Built on a commercial learning-based compositional DEM, i.e., OpenAI's Text-embedding-3-large
E5-ICS	Benchmark ICS	Built on an open-source learning-based compositional DEM, i.e., E5-mistral-7b
VLAWE-ICS	Benchmark ICS	Built on an aggregation-based compositional DEM, i.e., VLAWE
JoSE-ICS	Benchmark ICS	Built on a holistic DEM, i.e., JoSE
ETM-ICS	Benchmark ICS	Built on an advanced topic model, i.e., ETM

of the 10-K documents in a yearly 10-K corpus described in Table 3. For each model, we searched for its optimal configuration of hyperparameters using the 10-K corpus of year 2018 and the evaluation procedure depicted in Section 4.2.[11] We coded our proposed ADDE with PyTorch and trained it using the Adam optimizer (Kingma and Ba 2015) with the learning rate of 10^{-3} and the batch size of 4,096. The structure of GENN (Figure 2) was specified as a standard MLP with two hidden layer of 1024 neurons, each activated by the LeakyReLU function with slope 0.01. Other important hyperparameters for ADDE are n_e (document, word, and factor embedding dimension), n_t (time embedding dimension), n_{ws} (window size for each side of the context), n_f (number of hidden factors), z (during of an investigated time window), and n_{epoch} (number of training epochs). Specifically, we set n_e to 512, n_t to 256, n_{ws} to 20, n_f to 512, z to 3, and n_{epoch} to 20. The key hyperparameters for JoSE and their values (listed in parentheses) are as follows: document and word embedding dimension (400), and window size for each side of the context (10). VLAWE (Ionescu and Butnaru 2019) takes pretrained word embedding vectors as input and then clusters word vectors using the K-means clustering algorithm. The resulting centroids are treated as the embedding vectors of codewords. A document vector is then computed by concatenating the cumulative difference of the document word vectors w.r.t. each codeword vector. Finally, a power normalization is applied to each entry of the document vector. Consequently, the key hyperparameters for VLAWE include the number of clusters required by the clustering

[11] Specifically, we ran each model to embed 10-K documents in year 2018 and constructed its corresponding ICS based on the document embeddings. Next, we set the number of peers to $k = 10$, and evaluated the performance of a hyperparameter configuration based on the average R^2 of the monthly regressions specified in Equation 23 for year 2019.

algorithm and the exponent for the power normalization. We set the former as 10 and the latter as 0.5. Finally, for ETM (Dieng et al. 2020), we set the number of topics to 400.[12]

4.2. Evaluation Procedure

To compare the performance of the ICSs listed in Table 4, we adopted a standard evaluation procedure introduced by Lee et al. (2015). This procedure evaluates an ICS in terms of how well top-k peer firms identified by the ICS can explain the cross-sectional variation in focal firms' monthly stock returns as well as other important financial ratios. Take monthly stock return as an example to illustrate this procedure. Given a base year y (e.g., 2019), an ICS of year y was employed to identify the top-k peers for each focal firm. Next, the following cross-sectional regression was performed for every month t in the *subsequent* year $y+1$ (e.g., 2020):

$$r_{t,i} = \alpha_t + \beta_t r^p_{t,i} + \epsilon_{t,i}, \tag{23}$$

where $r_{t,i}$ is the stock return for firm i at month t and $r^p_{t,i}$ is the average stock return for firm i's top-k peers at month t. The R^2 of the regression reveals the percentage of the cross-sectional variation in firms' monthly stock returns that can be explained by their peers identified by the ICS. The evaluation process was then repeated with years $y+1$ and $y+2$ as the base years, respectively. Finally, the average R^2 across 36 regressions measures the performance of the ICS (Lee et al. 2015).[13] Peer firms that are more economically related to their corresponding focal firms should exhibit greater contemporaneous correlation with them in monthly stock returns (Lee et al. 2015). Therefore, a larger value of the average R^2 indicates better performance of an ICS (Lee et al. 2015). This procedure can be applied to other financial ratios by replacing monthly stock return in Equation 23 with an investigated financial ratio.

To conduct the evaluation described above, we computed firms' key financial ratios, such as stock return and price-to-earnings ratio, using data provided by WRDS. Specifically, firms' financial data, such as earnings, were obtained from WRDS's Compustat database, with each firm identified by the Global Company Key (GVKEY), and firms' stock returns came from WRDS's CRSP database, with each security identified by the Permanent Security-level Identification Number (PERMNO).

[12] We used the code for JoSE (https://github.com/yumeng5/Spherical-Text-Embedding), VLAWE (https://github.com/raduionescu/vlawe-boswe/), and ETM (https://github.com/adjidieng/ETM) provided by the authors.

[13] As discussed in Section 1, expert-driven ICSs, such as GICS, consider all firms belonging to the same industry as equally similar to a focal firm, and therefore cannot identify ranked peers of the focal firm. To measure the performance of GICS, we exactly followed the procedure suggested by Lee et al. (2015). Specifically, we constructed the peers for each focal firm by randomly selecting k other firms in the same 6-digit GICS industry as the focal firm. We then ran the regression given by Equation 23. This process was conducted 500 times for each month, and the R^2 for the month was set to the average across these 500 runs. We applied the same procedure to evaluate the performance of SIC and NAICS by respectively focusing on the 2-digit SIC industries and the 3-digit NAICS industries, because these industry levels are comparable to the 6-digit GICS industries (Bhojraj et al. 2003).

However, firms in the evaluated ICSs are identified by a different key. For example, in its 10-K document, a firm is identified by the Central Index Key (CIK). Consequently, for ICSs built on vector representations of 10-K documents, such as ADDE-ICS and ETM-ICS, firms are identified by their CIKs. We therefore linked firms in an ICS to their corresponding financial ratios by mapping the firm key used by the ICS to GVKEY and PERMNO. For instance, to connect firms in ADDE-ICS to their financial ratios, we used the mapping between CIK and GVKEY in WRDS's COMPHIST dataset as well as the mapping between GVKEY and PERMNO in WRDS's CCM Link History dataset.

4.3. Evaluation Results

Following the evaluation procedure, we compared the performance of the ICSs in Table 4 by setting the financial ratio to monthly stock return, the (starting) base year y to 2019, and the number of peers k to 10. Table 5 summarizes the evaluation results. ADDE-ICS outperforms the five existing ICSs: SIC, NAICS, GICS, TNIC, and ETNIC, by 210.98%, 188.98%, 128.76%, 25.96% and 19.13%, respectively. Our proposed ICS also substantially outperforms ICSs built on existing DEMs or topic model, with improvements ranging from 17.35% (over JoSE-ICS) to 63.47% (over ETM-ICS). The evaluation results suggest the following. First, ADDE-ICS can identify economically related firms more effectively than the benchmark ICSs, as evidenced by the fact that the top-10 peers identified by ADDE-ICS explain the cross-sectional variation in focal firms' monthly stock returns significantly better than the top-10 peers identified by any of the benchmark ICSs. The performance advantage of ADDE-ICS stems primarily from the methodological novelties of ADDE, which enable it to better capture the semantics of 10-K documents than existing document representation methods. Second, the relatively lower performance of GICS, NAICS and SIC can be attributed to the limitation of expert-driven ICSs, which are unable to produce granular pairwise similarities between firms. Third, among the ICSs constructed using DEMs, those based on holistic DEMs, such as ADDE-ICS, JoSE-ICS, and ETNIC, outperform those based on compositional DEMs, including E5-ICS, OpenAI-ICS, and VLAWE-ICS. This suggests that holistic DEMs are better suited for representing 10-K documents than compositional DEMs, because a compositional DEM may lose focus when aggregating a large number of words that convey a diverse set of concepts. Finally, the extremely low average R^2 obtained by randomly selecting k peers for a focal firm underscores the challenge of identifying economically related firms and highlights the value of developing advanced methods for industry classification.

We conducted robustness checks by varying the experimental setting. First, we changed the (starting) base year to 2020, while keeping the financial ratio to monthly stock return and the number of peers k to 10. As reported in Table 6, ADDE-ICS still consistently and significantly

Table 5 Performance Comparison between ADDE-ICS and Benchmark ICSs: Monthly Stock Return (Base Year y = 2019, Number of Peers = 10)

	Average R^2	Improvement by ADDE-ICS (%)
ADDE-ICS	0.08026	
GICS	0.03509	128.76
NAICS	0.02777	188.98
SIC	0.02581	210.98
ETNIC	0.06737	19.13
TNIC	0.06372	25.96
JoSE-ICS	0.06839	17.35
E5-ICS	0.06454	24.36
OpenAI-ICS	0.06183	29.80
VLAWE-ICS	0.05203	54.26
ETM-ICS	0.04910	63.47
Random Peers	0.00032	25355.70

Note: The performance difference between ADDE-ICS and each benchmark ICS is statistically significant at the 1% level.

Table 6 Performance Comparison between ADDE-ICS and Benchmark ICSs: Monthly Stock Return (Base Year y = 2020, Number of Peers = 10)

	Average R^2	Improvement by ADDE-ICS (%)
ADDE-ICS	0.07164	
GICS	0.03132	128.75
NAICS	0.02350	204.89
SIC	0.02195	226.33
ETNIC	0.06299	13.72
TNIC	0.05971	19.97
JoSE-ICS	0.06227	15.05
E5-ICS	0.05703	25.61
OpenAI-ICS	0.05523	29.71
VLAWE-ICS	0.05137	39.47
ETM-ICS	0.04921	45.58
Random Peers	0.00031	22772.38

Note: The performance difference between ADDE-ICS and each benchmark ICS is statistically significant at the 1% level.

outperforms each benchmark ICS. Second, we varied the number of peers k from 5 to 15, with a step size of 5, while maintaining the financial ratio to monthly stock return and the base year y to 2019. Table 7 shows the results of this robustness check. Again, ADDE-ICS consistently and substantially outperforms benchmark ICSs, regardless of the choice of k. Lastly, we conducted

experiments with other key financial ratios listed in Table 8, while keeping the base year y to 2019 and the number of peers k to 10. Because these financial ratios are reported on a quarterly basis, the regression specified in Equation 23 (with the stock return replaced by one of these financial ratios) was run on a quarterly basis for this set of experiments. Table 9 reports the average R^2 for each ICS under each investigated financial ratio. As shown, ADDE-ICS again outperforms all of the benchmark ICSs across the investigated financial ratios. In summary, our robustness checks further confirm the superiority of ADDE-ICS over the benchmark ICSs and highlight the challenge of identifying economically related firms, as evidenced by the extremely low average R^2 achieved by random peers. Additionally, in Appendix E, we demonstrate the superior performance of ADDE-ICS compared to both search-based ICS (Lee et al. 2015) and a naive approach that constructs an ICS based on the correlations of firms' stock returns.

Table 7 Performance Comparison between ADDE-ICS and Benchmark ICSs: Varying Number of Peers (Base Year $y = 2019$, Financial Ratio = Monthly Stock Return)

	5	10	15
ADDE-ICS	0.06880	0.08026	0.08439
GICS	0.02452	0.03509	0.04056
	(180.54)	(128.76)	(108.07)
NAICS	0.01910	0.02777	0.03226
	(260.20)	(188.98)	(161.60)
SIC	0.01763	0.02581	0.02913
	(290.32)	(210.98)	(189.67)
ETNIC	0.05748	0.06737	0.07092
	(19.69)	(19.13)	(18.99)
TNIC	0.05470	0.06372	0.06602
	(25.78)	(25.96)	(27.81)
JoSE-ICS	0.05594	0.06839	0.07277
	(22.98)	(17.35)	(15.97)
E5-ICS	0.05587	0.06454	0.06877
	(23.14)	(24.36)	(22.72)
OpenAI-ICS	0.05078	0.06183	0.06581
	(35.47)	(29.80)	(28.22)
VLAWE-ICS	0.04466	0.05203	0.05690
	(54.05)	(54.26)	(48.30)
ETM-ICS	0.04104	0.04910	0.05446
	(67.64)	(63.47)	(54.94)
Random Peers	0.00031	0.00032	0.00032
	(22171.43)	(25355.70)	(25951.59)

Note: The percentage improvement by ADDE-ICS over a benchmark ICS is displayed in parentheses under the average R^2 of the benchmark. The performance difference between ADDE-ICS and each benchmark ICS is statistically significant at the 1% level.

Table 8 Other Key Financial Ratios

Financial Ratio	Description	Calculation
pb	Price-to-book ratio	Market capitalization/total common equity
pe	Price-to-earnings ratio	Market capitalization/net income before extraordinary items
evs	Enterprise value-to-sales ratio	(Market capitalization + long-term debt)/net sales
roe	Return on equity	Net income before extraordinary items/total common equity
lev	Leverage	Long-term debt/total stockholder's equity

Table 9 Performance Comparison between ADDE-ICS and Benchmark ICSs: Other Key Financial Ratios (Base Year $y = 2019$, Number of Peers = 10)

	pb	pe	evs	roe	lev
ADDE-ICS	0.13849	0.05402	0.39965	0.04282	0.08154
GICS	0.05817	0.01848	0.22876	0.02153	0.04456
	(138.10)	(192.37)	(74.70)	(98.86)	(82.98)
NAICS	0.02893	0.01053	0.13970	0.01162	0.03171
	(378.68)	(413.09)	(186.08)	(268.57)	(157.16)
SIC	0.02085	0.00600	0.11805	0.00918	0.03127
	(564.25)	(800.33)	(238.54)	(366.23)	(160.72)
ETNIC	0.11329	0.02879	0.36487	0.03623	0.06373
	(22.24)	(87.67)	*(9.53)*	(18.19)	(27.93)
TNIC	0.12236	0.03505	0.35275	0.02697	0.06799
	(13.18)	(54.11)	(13.30)	(58.76)	(19.93)
JoSE-ICS	0.12069	0.03168	0.36536	0.03535	0.07521
	(14.75)	(70.51)	*(9.38)*	(21.12)	*(8.41)*
E5-ICS	0.11335	0.03360	0.32941	0.02927	0.07762
	(22.19)	(60.76)	(21.32)	(46.32)	*(5.05)**
OpenAI-ICS	0.11308	0.02513	0.30886	0.02962	0.06430
	(22.47)	(114.99)	(29.39)	(44.56)	(26.81)
VLAWE-ICS	0.11267	0.02299	0.30557	0.03285	0.05907
	(22.92)	(134.94)	(30.79)	(30.34)	(38.03)
ETM-ICS	0.11002	0.02223	0.29514	0.02979	0.05211
	(25.87)	(142.97)	(35.41)	(43.75)	(56.46)
Random Peers	0.00090	0.00110	0.00088	0.00119	0.00090
	(15219.27)	(4831.78)	(45288.12)	(3502.39)	(9003.91)

Note: The percentage improvement by ADDE-ICS over a benchmark ICS is displayed in parentheses under the average R^2 of the benchmark. The performance difference between ADDE-ICS and each benchmark ICS is statistically significant at the 1% level (body font) or 5% level (italic font), except for the case of E5-ICS on lev (italic font with an asterisk mark).

4.4. Performance Analysis

It is interesting to understand the superior performance of ADDE-ICS by connecting the methodological innovations of ADDE to the performance of ADDE-ICS. Methodologically, ADDE features two innovative components: (i) the adaptive gating mechanism (referred to as AGT) and (ii) the dynamic embedding mechanism (referred to as DYN). To isolate the contribution of each component to the performance of ADDE-ICS, we dropped them from ADDE one at a time. First, we dropped AGT from ADDE and referred to the resulting DEM as W/o-AGT and the ICS constructed based on this DEM as W/o-AGT-ICS. The performance difference between ADDE-ICS and W/o-AGT-ICS reveals the contribution of the adaptive gating mechanism to the performance of ADDE-ICS. Next, we removed DYN from W/o-AGT and referred to the resulting DEM as W/o-AGT-DYN and its associated ICS as W/o-AGT-DYN-ICS. The performance difference between W/o-AGT-ICS and W/o-AGT-DYN-ICS uncovers the contribution of the dynamic embedding mechanism to the performance of ADDE-ICS.

Table 10 Performance Comparison among ADDE-ICS, W/o-AGT-ICS, and W/o-AGT-DYN-ICS: Monthly Stock Return

(Base Year $y = 2019$, Number of Peers = 10)

	Average R^2	Improvement by ADDE-ICS (%)	Improvement by W/o-AGT-ICS (%)
ADDE-ICS	0.08026		
W/o-AGT-ICS	0.07709	4.11	
W/o-AGT-DYN-ICS	0.07400		4.18

We compared the performance of ADDE-ICS, W/o-AGT-ICS, and W/o-AGT-DYN-ICS by setting the financial ratio to monthly stock return, the base year y to 2019, and the number of peers k to 10. As reported in Table 10, ADDE-ICS outperforms W/o-AGT-ICS by 4.11%, whereas W/o-AGT-ICS surpasses W/o-AGT-DYN-ICS by 4.18%, indicating that the adaptive gating mechanism and the dynamic embedding mechanism contribute comparably significant to the performance of ADDE-ICS.

4.5. Economic Value

Having demonstrated the superior performance of ADDE-ICS in identifying economically related firms, we conducted further experiments to show the economic value that could be generated from such advantage. Our experiments build on the well-documented phenomenon of industry momentum (Moskowitz and Grinblatt 1999, p. 1249), which suggests that it is profitable to "buy stocks from past winning industries and sell stocks from past losing industries," and closely follow the experimental procedure introduced by Lee et al. (2019). More specifically, Lee et al. (2019)

demonstrate how firms' ranked peers can be used to construct winning and losing portfolios in a trading strategy, thereby revealing the economic value of the mechanism (e.g., an ICS) in identifying these ranked peers. Consider an investigated year Y and an ICS of year Y. Let t be a month in year $Y+1$. For each firm i at month t, we computed $\tilde{r}^p_{t,i}$ as the weighted average of firm i's top-k peers' stock returns at month t, where the peers are weighted according to their similarities to firm i. Both top-k peers and pairwise similarities between firms were provided by the ICS. Next, firms were sorted according to their $\tilde{r}^p_{t,i}$. We used the top q firms with the highest $\tilde{r}^p_{t,i}$ to construct a winning portfolio and the bottom q firms with the lowest $\tilde{r}^p_{t,i}$ to form a losing portfolio. The trading strategy was to buy the winning portfolio while short selling the losing portfolio at month $t+1$. Finally, the excess return of the trading strategy was calculated as the average return of the stocks in the winning portfolio at month $t+1$ minus the average return of the stocks in the losing portfolio at month $t+1$. We conducted the experiments for investigated years from $Y=2019$ to $Y=2022$. The average excess return across the experiments reveals the economic value of an investigated ICS, because the trading strategy directly employs pairwise firm similarities and ranked peers provided by the ICS.

Table 11 Economic Value
(Number of Peers = 15, Portfolio Size = 5% of Firms)

	Average excess return (%)	Improvement by ADDE-ICS (%)
ADDE-ICS	2.453	
JoSE-ICS	2.111	16.22
ETNIC	2.036	20.50

We evaluated the economic value of ADDE-ICS, ETNIC (the best-performing existing ICS in identifying economically related firms), and JoSE-ICS (the best-performing ICS constructed using an existing DEM). In our experiments, we set the number of peers k to 15 and the portfolio size q to 5% of the firms. As reported in Table 11, the average excess return of ADDE-ICS is 2.453%, indicating that, by following the trading strategy constructed using ADDE-ICS, buying a winning portfolio worth \$100 while short selling a losing portfolio of the same value yields an average monthly profit of \$2.453, excluding trading costs.[14] It is noted that the average excess return of ADDE-ICS outperforms that of ETNIC by 20.50% and that of JoSE-ICS by 16.22%. That is, the trading strategy informed by ADDE-ICS generates substantially higher profit than that constructed using ETNIC or JoSE-ICS.

[14] The magnitudes of excess returns reported in this table are comparable to those reported in the literature, e.g., Lee et al. (2019).

5. Conclusion

5.1. Summary and Contributions

We propose a novel DEM, which can effectively capture the semantics of 10-K documents by modeling concept switching within a 10-K document and tracking the evolution of a firm's 10-K documents over time. Built upon the proposed DEM, we develop a new ICS. Through extensive empirical evaluations, we demonstrate the superiority of our proposed ICS in identifying economically related firms, compared to representative existing ICSs as well as ICSs constructed using state-of-the-art DEMs. We also show substantial economic value generated from the superior performance of our ICS in a case study.

Our study falls into the computational genre of design science research in the IS discipline (Rai 2017, Abbasi et al. 2024). In general, computational design science research develops novel computational methods to solve important business and societal problems, thereby making methodological contributions to the literature (Rai 2017, Padmanabhan et al. 2022). Specifically, this study addresses a critical business problem of industry classification and develops an effective ICS based on a novel DEM. Driven by the need to capture the semantics of 10-K documents, the key innovations of our DEM lie in its adaptive gating mechanism and dynamic embedding mechanism, which together constitute the methodological contributions of our study. Additionally, ICSs play a crucial role in both business research and practice. In this regard, our study contributes to business research and practice by developing a new ICS that can effectively identify economically related firms.

5.2. Research and Managerial Implications

Our study offers several implications for research. First, effective identification of economically related firms is important for business research. Our proposed ADDE-ICS enables researchers to identify economically related firms more effectively than existing ICSs, which could, in turn, result in more rigorous inferences in answering a wide range of business research questions investigated using ICSs (e.g., Li et al. 2013, Hoberg and Phillips 2018, Cao et al. 2019). Moreover, ADDE-ICS's ranked peer designations facilitate research settings where researchers need to select a focal firm's most similar firms to form a control group or to serve as a benchmark (e.g., Cao et al. 2019). Traditional expert-driven ICSs are less appropriate for these research settings, because these ICSs view all firms in the same industry of a focal firm as being equally similar to the focal firm. Second, our study has important implications for IS research as well. Various ICSs have been employed to examine the relationship between information technology and firm performance (Bharadwaj 2000, Tanriverdi 2005, Bharadwaj et al. 2007). To tease out the impact of industry characteristics on firm performance, it is essential for these studies to control for industry characteristics (Tanriverdi 2005), form control samples matched by industry (Bharadwaj 2000), or construct industry-adjusted

variables (Bharadwaj et al. 2007). In comparison to existing ICSs, ADDE-ICS could be more effective in teasing out the compounding effects of industry characteristics and provide more rigorous inferences because ADDE-ICS can identify economically related firms more effectively. Furthermore, recent IS studies develop novel metrics for identifying a firm's competitors or predicting future labor market competition (Pant and Sheng 2015, Liu et al. 2020). The performance of the proposed metrics is evaluated by benchmarking them against metrics derived from industry peers identified by an ICS. In this vein, one could develop more competent benchmark metrics based on ADDE-ICS, which, in turn, could better reveal the performance and value of the proposed metrics. Third, textual analysis has become a major tool for business research (Loughran and Mcdonald 2016). Our proposed document representation method offers a powerful tool for analyzing business documents. It is particularly useful for capturing rich semantics of business documents that feature shifting concepts within a single document and evolving patterns across a sequence of documents, such as firms' financial reports including 10-K and 10-Q reports.

Our proposed ADDE-ICS is of great interest to business practitioners. For example, *ex ante*, in setting executive compensation packages, a firm's board of directors frequently uses the pay contracts of executives from self-selected compensation peers as a benchmark (Bizjak et al. 2011). ADDE-ICS enables the board to efficiently and effectively identify the most comparable peers in terms of economical relatedness. This not only leads to efficient contracting but also reduces litigation risks, since U.S. security regulation requires disclosure of the identified compensation peers. Proper identification of compensation peers could also enhance *ex post* executive performance evaluation. To conduct this evaluation, a firm's board needs to filter out common shocks that affect both the firm and its peers. More accurate peer identifications allow for better capture of common shocks to firms' product markets, thereby filtering out stock return movements unrelated to the actions of the firm's executives. Another potential application of ADDE-ICS is to facilitate the design of expert-driven ICSs. The pairwise firm similarities produced by ADDE-ICS offer a useful reference for experts to decide which firms should be grouped into one industry. Moreover, because pairwise firm similarities in ADDE-ICS evolve over time, it can signal potential membership changes in an expert-driven ICS, prompting its designers to make adjustments accordingly. Specifically, using pairwise firm similarities provided by ADDE-ICS, one can calculate the similarity between a firm and an industry in an expert-driven ICS. For instance, the similarity can be computed as the average of the pairwise similarities between the firm and each firm in the industry. One can then monitor the changes of these computed similarities over time. A decreasing trend of the similarity between a firm and its current industry suggests that the firm may exit its current industry, while those industries featuring increasing similarity trend with the firm serve as candidates for its entering industry.

5.3. Future Work

We acknowledge the limitations of our study. First, as described in Appendix C, ADDE employs TinyBERT$_4$ to represent local contexts. Future work could fine-tune TinyBERT$_4$ using 10-K corpora to further improve the performance of ADDE (Huang et al. 2022). However, it is crucial to avoid information leakage during this fine-tuning process. Specifically, when applying ADDE to embed 10-K documents in year Y, TinyBERT$_4$ should not be fine-tuned with 10-K documents after year Y. Second, because the objective of this study is to develop a novel ICS, our empirical evaluation focuses on comparing our proposed ICS against benchmark ICSs using a standard procedure for evaluating ICSs. Future work could assess the performance of our proposed document embedding model, ADDE, using standard evaluation procedure and metrics in NLP. Third, our ADDE-ICS only covers public firms because business descriptions of private firms are not readily available. Future work could build crawlers to extract business descriptions from private firms' websites, apply ADDE to represent these extracted business descriptions, and expand ADDE-ICS to include both public and private firms. Finally, beyond addressing the aforementioned limitations, future research could explore leveraging the ability of ADDE to embed chronologically evolving documents with shifting concepts in other business domains. For example, ADDE could be applied to legal and patent documents to answer important questions in these fields.

Acknowledgments

The authors thank Dr. Roger Silvers for inspiring their interest in industrial classification and for his valuable discussions and suggestions during the preliminary stages of this study.

References

Abbasi A, Parsons J, Pant G, Sheng ORL, Sarker S (2024) Pathways for design research on artificial intelligence. *Information Systems Research* 35(2):441–459.

Andreou PC, Harris T, Philip D (2020) Measuring Firms' Market Orientation Using Textual Analysis of 10-K Filings. *British Journal of Management* 31(4):872–895, ISSN 1467-8551.

Arora S, Liang Y, Ma T (2017) A simple but tough-to-beat baseline for sentence embeddings. *International Conference on Learning Representations*.

Bahdanau D, Cho K, Bengio Y (2015) Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*.

Banerjee S, Mishra B, Jawanpuria P, Shrivastava MS (2022) Generalised Spherical Text Embedding. *Proceedings of the 19th International Conference on Natural Language Processing (ICON)*, 80–85 (New Delhi, India: Association for Computational Linguistics).

Bharadwaj AS (2000) A resource-based perspective on information technology capability and firm performance: an empirical investigation. *MIS Quarterly* 169–196.

Bharadwaj AS, Bharadwaj SG, Konsynski BR (1999) Information technology effects on firm performance as measured by tobin's q. *Management Science* 45(7):1008–1024.

Bharadwaj S, Bharadwaj A, Bendoly E (2007) The performance effects of complementarities between information systems, marketing, manufacturing, and supply chain processes. *Information Systems Research* 18(4):437–453.

Bhojraj S, Lee CMC, Oler DK (2003) What's my line? A comparison of industry classification schemes for capital market research. *Journal of Accounting Research* 41(5):745–774.

Bishop CM (1994) Mixture density networks. Technical report.

Bizjak J, Lemmon M, Nguyen T (2011) Are all CEOs above average? An empirical analysis of compensation peer groups and pay design. *Journal of Financial Economics* 100(3):538–555.

Blei DM, Ng AY, Jordan MI, Lafferty J (2003) Latent dirichlet allocation. *Journal of Machine Learning Research* 3(4/5):993–1022.

Bowman SR, Vilnis L, Vinyals O, Dai A, Jozefowicz R, Bengio S (2016) Generating sentences from a continuous space. *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 10–21.

Cao J, Liang H, Zhan X (2019) Peer effects of corporate social responsibility. *Management Science* 65(12):5487–5503.

Chen M (2017) Efficient vector representation for documents through corruption. *International Conference on Learning Representations*.

Chung S, Animesh A, Han K, Pinsonneault A (2019) Software patents and firm value: A real options perspective on the role of innovation orientation and environmental uncertainty. *Information Systems Research* 30(3):1073–1097.

Debreceny R, Farewell S, Piechocki M, Felden C, Gräning A (2010) Does it add up? Early evidence on the data quality of XBRL filings to the SEC. *Journal of Accounting and Public Policy* 29(3):296–306.

Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186.

Dieng AB, Ruiz FJR, Blei DM (2020) Topic Modeling in Embedding Spaces. *Transactions of the Association for Computational Linguistics* 8:439–453.

Doersch C (2016) Tutorial on variational autoencoders. *arXiv:1606.05908 [cs, stat]* .

Fama EF, French KR (1997) Industry costs of equity. *Journal of Financial Economics* 43(2):153–193.

Gong R (2025) Peers' performance and sensitivity of investment to peers' stock price: Examining the moderating role of CEO overconfidence. *Journal of Business Research* 186:114948, ISSN 0148-2963.

Hoberg G, Phillips G (2016) Text-based network industries and endogenous product differentiation. *Journal of Political Economy* 124(5):1423–1465.

Hoberg G, Phillips G (2018) Conglomerate industry choice and product language. *Management Science* 64(8):3735–3755.

Hoberg G, Phillips GM (2024) Scope, Scale, and Concentration: The 21st-Century Firm. *The Journal of Finance* Forthcoming.

Huang AH, Wang H, Yang Y (2022) FinBERT: A Large Language Model for Extracting Information from Financial Text. *Contemporary Accounting Research* 40(2):806–841.

Ionescu RT, Butnaru A (2019) Vector of Locally-Aggregated Word Embeddings (VLAWE): A Novel Document-level Representation. *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)*, 363–369 (Minneapolis, Minnesota: Association for Computational Linguistics).

Jaffe AB (1986) Technological opportunity and spillovers of R&D: Evidence from firms' patents, profits, and market value. *The American Economic Review* 76(5):984–1001.

Jegadeesh N, Wu D (2013) Word power: A new approach for content analysis. *Journal of Financial Economics* 110(3):712–729.

Joulin A, Grave E, Bojanowski P, Mikolov T (2017) Bag of tricks for efficient text classification. *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, volume 2, 427–431.

Kingma DP, Ba J (2015) Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Le Q, Mikolov T (2014) Distributed representations of sentences and documents. *Proceedings of the 31st International Conference on Machine Learning*, volume 32, 1188–1196.

Lee CMC, Ma P, Wang CCY (2015) Search-based peer firms: Aggregating investor perceptions through internet co-searches. *Journal of Financial Economics* 116(2):410–431.

Lee CMC, Sun ST, Wang R, Zhang R (2019) Technological links and predictable returns. *Journal of Financial Economics* 132(3):76 – 96.

Li F, Lundholm R, Minnis M (2013) A measure of competition based on 10-k filings. *Journal of Accounting Research* 51(2):399–436.

Li J, Luong T, Jurafsky D (2015) A hierarchical neural autoencoder for paragraphs and documents. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, volume 1, 1106–1115.

Liu Y, Pant G, Sheng OR (2020) Predicting labor market competition: Leveraging interfirm network and employee skills. *Information Systems Research* 31(4):1443–1466.

Logeswaran L, Lee H (2018) An efficient framework for learning sentence representations. *International Conference on Learning Representations*.

Loughran T, Mcdonald B (2016) Textual analysis in accounting and finance: A survey. *Journal of Accounting Research* 54(4):1187–1230.

Mekala D, Gupta V, Paranjape B, Karnick H (2017) SCDV : Sparse composite document vectors using soft clustering over distributional representations. *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 659–669.

Meng Y, Huang J, Wang G, Zhang C, Zhuang H, Kaplan L, Han J (2019) Spherical text embedding. *Advances in Neural Information Processing Systems* 32.

Miao Y, Yu L, Blunsom P (2016) Neural variational inference for text processing. *Proceedings of the 33rd International Conference on Machine Learning*, volume 48, 1727–1736.

Moskowitz TJ, Grinblatt M (1999) Do industries explain momentum? *The Journal of Finance* 54(4):1249–1290.

Nie Z, Feng Z, Li M, Zhang C, Zhang Y, Long D, Zhang R (2024) When Text Embedding Meets Large Language Model: A Comprehensive Survey. ArXiv:2412.09165 [cs].

OpenAI (2023) GPT-4 Technical Report. ArXiv:2303.08774[cs].

Padmanabhan B, Fang X, Sahoo N, Burton-Jones A (2022) Machine learning in information systems research. *MIS Quarterly* 46(1):iii–xix.

Pagliardini M, Gupta P, Jaggi M (2018) Unsupervised learning of sentence embeddings using compositional n-gram features. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, 528–540.

Pant G, Sheng OR (2015) Web footprints of firms: Using online isomorphism for competitor identification. *Information Systems Research* 26(1):188–209.

Pennington J, Socher R, Manning CD (2014) GloVe: Global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, 1532–1543.

Petukhova A, Matos-Carvalho JP, Fachada N (2024) Text clustering with LLM embeddings. ArXiv:2403.15112 [cs].

Phillips RL, Ormsby R (2016) Industry classification schemes: An analysis and review. *Journal of Business & Finance Librarianship* 21(1):1–25.

Rai A (2017) Editor's comments: Diversity of design science research. *MIS Quarterly* 41(1):iii–xviii.

Reimers N, Gurevych I (2019) Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3982–3992 (Hong Kong, China: Association for Computational Linguistics).

Shi Z, Lee GM, Whinston AB (2016) Toward a better measure of business proximity: Topic modeling for industry intelligence. *MIS Quarterly* 40(4):1035–A53.

Sprenger TO, Welpe IM (2011) Tweets and peers: Defining industry groups and strategic peers based on investor perceptions of stocks on Twitter. *Algorithmic Finance* 1(1):57–76.

Sun X, Meng Y, Ao X, Wu F, Zhang T, Li J, Fan C (2022) Sentence Similarity Based on Contexts. *Transactions of the Association for Computational Linguistics* 10:573–588, ISSN 2307-387X.

Tanriverdi H (2005) Information technology relatedness, knowledge management capability, and performance of multibusiness firms. *MIS Quarterly* 311–334.

Touvron H, Lavril T, Izacard G, Martinet X, Lachaux MA, Lacroix T, Rozière B, Goyal N, Hambro E, Azhar F, Rodriguez A, Joulin A, Grave E, Lample G (2023) LLaMA: Open and Efficient Foundation Language Models. ArXiv:2302.13971 [cs].

Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, Kaiser L, Polosukhin I (2017) Attention is all you need. *Advances in Neural Information Processing Systems*, 5998–6008.

Wang B, Di Buccio E, Melucci M (2021) Word2Fun: Modelling Words as Functions for Diachronic Word Representation. *Advances in Neural Information Processing Systems*, volume 34 (Curran Associates, Inc.).

Wang L, Yang N, Huang X, Yang L, Majumder R, Wei F (2024) Improving Text Embeddings with Large Language Models. Ku LW, Martins A, Srikumar V, eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 11897–11916 (Bangkok, Thailand: Association for Computational Linguistics).

Weiner C (2005) The impact of industry classification schemes on financial research. *SSRN* .

Wood S, Muthyala R, Jin Y, Qin Y, Rukadikar N, Rai A, Gao H (2017) Automated industry classification with deep learning. *2017 IEEE International Conference on Big Data (Big Data)*, 122–129.

Xu D, Ruan C, Körpeoglu E, Kumar S, Achan K (2020) Inductive representation learning on temporal graphs. *8th International Conference on Learning Representations, ICLR 2020* (OpenReview.net).

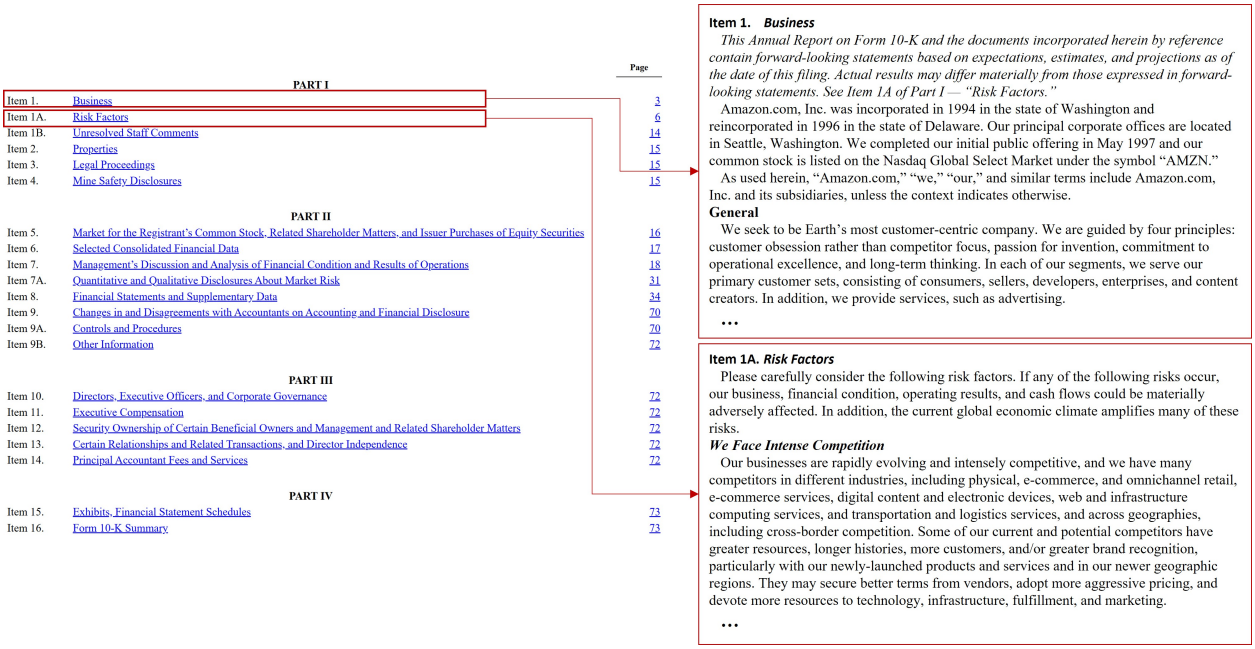
Yang SY, Liu FC, Zhu X, Yen DC (2019) A graph mining approach to identify financial reporting patterns: An empirical examination of industry classifications. *Decision Sciences* 50(4):847–876.

Zhao X, Fang X, He J, Huang L (2023) Exploiting Expert Knowledge for Assigning Firms to Industries: A Novel Deep Learning Method. *MIS Quarterly* 47(3):1147–1176.

Supplementary Material

A. An Example of Item 1 and Item 1A in a 10-K Report

Figure A.1 displays an annual 10-K report of Amazon.[15] The left side of the figure presents the table of contents of the 10-K report, which consists of 16 items. The right side of the figure shows text excerpts from Item 1 and Item 1A of the 10-K report (i.e., 10-K document).



		Page
PART I		
Item 1.	Business	3
Item 1A.	Risk Factors	6
Item 1B.	Unresolved Staff Comments	14
Item 2.	Properties	15
Item 3.	Legal Proceedings	15
Item 4.	Mine Safety Disclosures	15
PART II		
Item 5.	Market for the Registrant's Common Stock, Related Shareholder Matters, and Issuer Purchases of Equity Securities	16
Item 6.	Selected Consolidated Financial Data	17
Item 7.	Management's Discussion and Analysis of Financial Condition and Results of Operations	18
Item 7A.	Quantitative and Qualitative Disclosures About Market Risk	31
Item 8.	Financial Statements and Supplementary Data	34
Item 9.	Changes in and Disagreements with Accountants on Accounting and Financial Disclosure	70
Item 9A.	Controls and Procedures	70
Item 9B.	Other Information	72
PART III		
Item 10.	Directors, Executive Officers, and Corporate Governance	72
Item 11.	Executive Compensation	72
Item 12.	Security Ownership of Certain Beneficial Owners and Management and Related Shareholder Matters	72
Item 13.	Certain Relationships and Related Transactions, and Director Independence	72
Item 14.	Principal Accountant Fees and Services	72
PART IV		
Item 15.	Exhibits, Financial Statement Schedules	73
Item 16.	Form 10-K Summary	73

Item 1. *Business*

This Annual Report on Form 10-K and the documents incorporated herein by reference contain forward-looking statements based on expectations, estimates, and projections as of the date of this filing. Actual results may differ materially from those expressed in forward-looking statements. See Item 1A of Part I — "Risk Factors."

Amazon.com, Inc. was incorporated in 1994 in the state of Washington and reincorporated in 1996 in the state of Delaware. Our principal corporate offices are located in Seattle, Washington. We completed our initial public offering in May 1997 and our common stock is listed on the Nasdaq Global Select Market under the symbol "AMZN."

As used herein, "Amazon.com," "we," "our," and similar terms include Amazon.com, Inc. and its subsidiaries, unless the context indicates otherwise.

General

We seek to be Earth's most customer-centric company. We are guided by four principles: customer obsession rather than competitor focus, passion for invention, commitment to operational excellence, and long-term thinking. In each of our segments, we serve our primary customer sets, consisting of consumers, sellers, developers, enterprises, and content creators. In addition, we provide services, such as advertising.

...

Item 1A. *Risk Factors*

Please carefully consider the following risk factors. If any of the following risks occur, our business, financial condition, operating results, and cash flows could be materially adversely affected. In addition, the current global economic climate amplifies many of these risks.

We Face Intense Competition

Our businesses are rapidly evolving and intensely competitive, and we have many competitors in different industries, including physical, e-commerce, and omnichannel retail, e-commerce services, digital content and electronic devices, web and infrastructure computing services, and transportation and logistics services, and across geographies, including cross-border competition. Some of our current and potential competitors have greater resources, longer histories, more customers, and/or greater brand recognition, particularly with our newly-launched products and services and in our newer geographic regions. They may secure better terms from vendors, adopt more aggressive pricing, and devote more resources to technology, infrastructure, fulfillment, and marketing.

...

Figure A.1 An Annual 10-K Report of Amazon

B. An Example of Concept Switching

Table A.1 presents an example of concept switching in a 10-K document of Amazon.[16] As illustrated, the underlying concept shifts as the narrative progresses through the document.

C. Embedding Local Context

We employed TinyBERT$_4$ (Jiao et al. 2020) to embed each local context. As a distilled version of the pretrained BERT$_{\text{BASE}}$ (Devlin et al. 2019), TinyBERT$_4$ significantly reduces inference time compared to BERT$_{\text{BASE}}$, making it particularly well-suited for local context embedding in our study. Specifically, TinyBERT$_4$ summarizes the meaning of a local context by using

[15] The report was retrieved from https://www.sec.gov/Archives/edgar/data/1018724/000101872419000004/0001018724-19-000004-index.htm.

[16] The document was retrieved from https://www.sec.gov/Archives/edgar/data/1018724/000101872419000004/0001018724-19-000004-index.htm.

Content	Underlying Concept
"We serve consumers through our online and physical stores and focus on selection, price, and convenience. We design our stores to enable hundreds of millions of unique products to be sold by us and by third parties across dozens of product categories."	Customer Serving
...	
"Our businesses encompass a large variety of product types, service offerings, and delivery channels. The worldwide marketplace in which we compete is evolving rapidly and intensely competitive, and we face a broad array of competitors from many different industry sectors around the world."	Competition
...	
"We regard our trademarks, service marks, copyrights, patents, domain names, trade dress, trade secrets, proprietary technologies, and similar intellectual property as critical to our success, and we rely on trademark, copyright, and patent law, trade-secret protection, and confidentiality and/or license agreements with our employees, customers, partners, and others to protect our proprietary rights."	Intellectual Property
...	
"Our international activities are significant to our revenues and profits, and we plan to further expand internationally. In certain international market segments, we have relatively little operating experience and may not benefit from any first-to-market advantages or otherwise succeed. It is costly to establish, develop, and maintain international operations and stores, and promote our brand internationally. Our international operations may not be profitable on a sustained basis."	Risk Factors
...	

Table A.1 An Example of Concept Switching

a special [MASK] token. The embedding procedure in our study is described as follows. Given the word sequence of a local context $c_t =< w_{t-n_{\text{ws}}}, \cdots, w_{t-1}, w_{t+1}, \ldots w_{t+n_{\text{ws}}} >$, we extended the local context by inserting the [MASK] token at the center of the sequence, resulting in $\tilde{c}_t =< w_{t-n_{\text{ws}}}, \cdots, w_{t-1}, [\text{MASK}], w_{t+1}, \ldots w_{t+n_{\text{ws}}} >$. Next, we fed the token sequence $\tilde{c}_t$ into the pretrained TinyBERT$_4$ model, and the embedding vector of the [MASK] token represented the local context c_t. We employed the off-the-shelf version of TinyBERT$_4$, and froze its parameters when used it as the local context embedding module of ADDE.[17] Because the pretraining of TinyBERT$_4$ does not involve any textual information available after year 2018 nor the 10-K corpus, integrating it in our method should not introduce any looking-ahead bias in the evaluation results reported in Section 4.

[17] The model parameters of TinyBERT$_4$ can be downloaded from https://huggingface.co/huawei-noah/TinyBERT_General_4L_312D.

D. A Description of ADDE-ICS

Our proposed document embedding method, ADDE, takes firms' annual 10-K documents as inputs and produces a time-varying ADDE-ICS for each year. In this study, we have constructed ADDE-ICSs for the years 2019 to 2022, with each year having its own ADDE-ICS. For example, in year 2019, 5,741 firms filed 10-K reports. Accordingly, ADDE-ICS of year 2019 consists of 5,741 tables, one for each firm. Each table contains a focal firm, its peers ranked based on semantic similarities between the focal firm's 10-K document and those of its peers, along with similarity scores. Table A.2 presents an example table for the focal firm *Alphabet Inc.*. According to ADDE-ICS, the most economically related peer to *Alphabet Inc.* in 2019 was *Facebook Inc.*, followed by *Twitter Inc.* as its second most economically related peer, and so on. Some of the identified top peers are

Focal Firm	Peer Firm	Similarity Score	Peer Rank
Alphabet Inc.	Facebook Inc.	0.7734	1
	Twitter Inc.	0.7321	2
	Akamai Technologies Inc.	0.7105	3
	Snap Inc.	0.7088	4
	Zynga Inc.	0.6933	5
	Zedge Inc.	0.6890	6
	Yelp Inc.	0.6757	7
	Leaf Group LTD.	0.6753	8
	TripAdvisor Inc.	0.6668	9
	GoPro Inc.	0.6628	10
	...	...	...
	...	...	5740

Table A.2 An Example Table from ADDE-ICS of Year 2019

economically related to *Alphabet Inc.* in that they provide internet-based services and products such as online advertising services. For example, *Facebook Inc.* directly competes with *Alphabet Inc.* in the online advertising market. Firms running vertical search engines such as *Yelp Inc.* and *TripAdvisor Inc.* compete with *Alphabet Inc.* by attracting some users directly to their websites rather than through Google.[18] There are collaborations between *Alphabet Inc.* and some other firms. For example, *Akamai Technologies Inc.* provides content delivery network (CDN) services that enhance the performance and reliability of *Alphabet Inc.*'s online platforms (such as Google and YouTube).

[18] See the 10-K report of *Alphabet Inc.*: https://www.sec.gov/Archives/edgar/data/1652044/000165204419000004/goog10-kq42018.htm.

E. Comparing with Additional Benchmark ICSs

E.1. Comparing with Search-based ICS

Lee et al. (2015) measure the similarity between two firms using their co-search frequency on the SEC's EDGAR website and then rank a focal firm's peers according to their similarities to the focal firm. We refer to the ICS constructed using this method as search-based ICS (SB-ICS). In Table 3 of Lee et al. (2015), the authors report the performance of SB-ICS, evaluated using the same procedure as described in Section 4.2 of our paper with base year $y = 2008$[19], except that the firm universe is restricted to be either the S&P 1500 or the S&P 500 firms. To ensure a fair comparison with SB-ICS, we constructed ADDE-ICSs for the years 2008, 2009, and 2010, and then ran the evaluation procedure specified in Section 4.2 by following the same setup as in Lee et al. (2015). Specifically, the base year, the number of peers, and the financial ratio were set to 2008, 10, and monthly stock return, respectively, and the firm universe consisted of firms from either the S&P 1500 or the S&P 500 index as of January 1, 2008.

Table A.3 Performance Comparison between ADDE-ICS and SB-ICS under S&P 1500 or S&P 500 Firms (Base Year $y = 2008$, Number of Peers=10, Financial Ratio = Monthly Stock Return)

	S&P 1500	S&P 500
ADDE-ICS	0.157	0.232
SB-ICS-TW	0.125	0.214
	(25.60)	(8.41)
SB-ICS-EW	0.114	0.189
	(37.72)	(22.75)

Note: The percentage improvement by ADDE-ICS over a benchmark ICS is displayed in parentheses under the average R^2 of the benchmark.

Table A.3 presents the evaluation results of ADDE-ICS and SB-ICS, where the average R^2s of SB-ICS-TW and SB-ICS-EW are directly taken from Table 3 of Lee et al. (2015). Both SB-ICS-TW and SB-ICS-EW are based on SB-ICS, but differ in the definition of the regression variable $r^p_{t,i}$ in Equation 23. Recall that $r^p_{t,i}$ denotes the average of firm i's top-k peers' stock returns at month t. For SB-ICS-EW, $r^p_{t,i}$ is computed as an unweighted average, while for SB-ICS-TW it is calculated as a weighted average, with weights being peer firms' similarities to focal firm i. As reported in the table, ADDE-ICS respectively outperforms SB-ICS-TW and SB-ICS-EW by 25.60% and 37.72% under the S&P 1500 firm universe, and by 8.41% and 22.75% under the S&P 500 firm universe.[20]

[19] Lee et al. (2015) evaluate the performance of SB-ICSs of years 2008, 2009, and 2010 using monthly stock return data in years 2009, 2010, and 2011, respectively.

[20] We can not test the statistical significance of the improvements by ADDE-ICS, because Lee et al. (2015) do not provide the sequence of monthly R^2 values of SB-ICS-EW or SB-ICS-TW but only their means.

E.2. Comparing with Return-based ICS

A naive approach to constructing an ICS of year Y is to measure the similarities between firms using the correlations of their key financial ratios in that year. To assess the performance of this approach, we built RET-ICS based on the correlations of firms' stock returns. Specifically, to construct RET-ICS of year Y, we measured the similarities between firms using the correlations of their monthly stock returns in year Y and then ranked a focal firm's peers according to their similarities to the focal firm. Using the evaluation procedure described in Section 4.2, we benchmarked our ADDE-ICS against RET-ICS by setting the base year to 2019, the number of peers to 10, and the financial ratio to monthly stock return. As reported in Table A.4, ADDE-ICS outperforms RET-ICS by 550.57%. For robustness checks, we changed the base year to 2020, and evaluated with different numbers of peers and other financial ratios. As shown in Tables A.5-A.7, ADDE-ICS still outperforms RET-ICS substantially. All the improvements by ADDE-ICS over RET-ICS are statistically significant at the 1% level.

Table A.4 Performance Comparison between ADDE-ICS and RET-ICS: Monthly Stock Return (Base Year $y = 2019$, Number of Peers = 10)

	Average R^2	Improvement by ADDE-ICS (%)
ADDE-ICS	0.08026	
RET-ICS	0.01234	550.57

Table A.5 Performance Comparison between ADDE-ICS and RET-ICS: Monthly Stock Return (Base Year $y = 2020$, Number of Peers = 10)

	Average R^2	Improvement by ADDE-ICS (%)
ADDE-ICS	0.07164	
RET-ICS	0.01232	481.49

Table A.6 Performance Comparison between ADDE-ICS and RET-ICS: Varying Number of Peers (Base Year $y = 2019$, Financial Ratio = Monthly Stock Return)

	5	10	15
ADDE-ICS	0.06880	0.08026	0.08439
RET-ICS	0.00513	0.01234	0.01672
	(1240.39)	(550.57)	(404.68)

Note: The percentage improvement by ADDE-ICS over a benchmark ICS is displayed in parentheses under the average R^2 of the benchmark.

Table A.7 Performance Comparison between ADDE-ICS and RET-ICS: Other Key Financial Ratios (Base Year $y = 2019$, Number of Peers = 10)

	pb	pe	evs	roe	lev
ADDE-ICS	0.13849	0.05402	0.39965	0.04282	0.08154
RET-ICS	0.04636	0.00592	0.11576	0.00414	0.00516
	(198.71)	(812.77)	(245.24)	(935.06)	(1480.72)

Note: The percentage improvement by ADDE-ICS over a benchmark ICS is displayed in parentheses under the average R^2 of the benchmark.

The underperformance of RET-ICS can be explained as follows. The objective of industry classification is to identify the economic relatedness between firms based on their fundamental economic characteristics, which persist over a period of time, rather than their transient traits (Lee et al. 2015, Back 2017). A firm's stock return captures many of its transient factors. Consequently, RET-ICS, which is constructed based on the correlations of firm's stock returns, performs poorly. In contrast, ADDE-ICS is developed through the analysis of firm's 10-K documents, which contain firms' fundamental information such as products and markets. Moreover, as described in Section 4.2, an ICS of year Y (e.g., 2019) is evaluated based on how well top peer firms identified by the ICS can explain cross-sectional variations in focal firms' monthly stock returns in the following year $Y+1$ (e.g., 2020) (Lee et al. 2015). Given transient factors in stock return, it is not surprising that RET-ICS, constructed using the correlations of stock returns in year Y, fails to explain the variations in stock returns in year $Y+1$.

References

Back KE (2017) *Asset Pricing and Portfolio Choice Theory* (Oxford University Press), 2 edition edition, ISBN 978-0-19-024114-8.

Devlin J, Chang MW, Lee K, Toutanova K (2019) BERT: Pre-training of deep bidirectional transformers for language understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 4171–4186.

Jiao X, Yin Y, Shang L, Jiang X, Chen X, Li L, Wang F, Liu Q (2020) TinyBERT: Distilling BERT for Natural Language Understanding. Cohn T, He Y, Liu Y, eds., *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4163–4174 (Online: Association for Computational Linguistics).

Lee CMC, Ma P, Wang CCY (2015) Search-based peer firms: Aggregating investor perceptions through internet co-searches. *Journal of Financial Economics* 116(2):410–431.