Word Embedding for Text-based Network Industries

Jiahui Bi

Abstract

The estimations of industry boundaries, competitions, and evolution are necessary for the study of industrial organization. This study proposes a new data analytical measurement of company proximity that analyzes the product, market and technology information in the unstructured text describing the firm's business using word-embedding and Word Mover Distance. We also classify companies' industries using a network approach with the firm's proximity. The new industry network captures not only the homogeneity within the industry but also the industry changes as well as the distance between industries. We compare our network industry with traditional industry classification methods Standard Industrial Classification (SIC) and North American Industrial Classification System (NAICS), along with the existing text-based firm proximity measurements (Hoberg and Phillips 2010a; Shi, Lee, and Whinston 2016). Finally, we validate our method by extracting network metrics to predict firms' Mergers and Acquisitions (M&A) in the U.S. public trade companies.

Introduction

Measuring company vicinity and identifying industry boundaries is a critical metric for managers to identify potential partners and competitors. A well-defined, traceable industry boundary is also crucial to the studies in many research areas such as industrial organization, finance, and accounting since the industry label or company similarity is widely used as a control variable in these research areas. However, with the convergence of industries, industry boundaries are gradually blurring, the clear lines of demarcation are reducing, and this profound change calls into question the validity of traditional industry classification. Considering the telecommunications industry, the Internet sectors are taking market share. Similarly, the entertainment industry, information industry, and communication industry are becoming more closely, as PCs/Phones have turned into a popular platform for listening to music and playing games. So, how to measure company similarity and define industrial boundaries is a critical and urgent question that needs to be answered.

In this research, we use the Word Embedding and Word Mover Distance method to develop a new measurement of firms' relatedness that analyzes the product, market, and technology information from the open, unstructured text, which describes the firm’s business and product descriptions filed with the Securities and Exchange Commission (SEC). Our starting point is to extract the most frequent and proper nouns in 10-K files, we process the text using word embedding and evaluate how firms are related to each other by Word Mover Distance, which avoids words frequent near-orthogonality problems and will capture the distance between individual words in the documents. With the pairwise company proximity, we build an industry network that clearly illustrates the industry boundaries as well as the network position of each company.

This industry network has the potential of extracting economically meaningful information from publicly available, unstructured text. The products’ similarities can help identify peer firms and show how firms are related to each other. Stocks of a focal firm are closely associated with related firms’ stock movements. In the same vein, peers’ valuation makes a significant contribution to a firm's investment. It can also be used to extract information like industry competitive level and product market threats. Furthermore, our research contributes to the study of corporate finance decisions. It helps us predict takeover targets and acquires.

Hoberg and Phillips (2010) prove that merging firms with more similar product descriptions in their 10-Ks experience more successful outcomes(Hoberg and Phillips 2010b). Take the merger of Disney and Pixar for example, Disney was classified in the business services industry based on SIC code, while Pixar was classified in the motion pictures industry. Firms are placed in the predefined industry group using the traditional classification system, we expect our method will indicate they have product and business similarities based on the text information.

Research Background

SIC and NAICS

The Standardized Industry Classification (SIC) system was established by an Interdepartmental Committee on Industrial Classification under the jurisdiction of the Central Statistical Board. The SIC system has a hierarchical structure, and the codes are assigned based on common characteristics shared in the products, services, production, and delivery system of a business. This system was widely used until the arrival of the North American Industry Classification System (NAICS) in the 90s. NAICS provides a greater level of detail about a firm's activity than SIC. These two industry classification schemes have some similarities: both assign industry code by humans; both have a hierarchical lineage; and both group companies based on their products, service and technology information. If two companies share the same industry code, they are considered as absolutely homogeneous, and the same on the other way.

Though these two codes are broadly used by all sorts of government agencies and scholars in many disciplines, the SIC and NAICS categories are blamed for failing to exhibit more similar financial/accounting homogeneity within industries(Bhojraj, Lee, and Oler 2003). Part of the problem arises because the Federal Census Bureau assigns the classification work to various data vendors and this assignment is not made on a consistent basis across these data vendors. Also, firms SIC and NAICS codes update infrequently, neither can easily accommodate innovations that create entirely new product markets.

Text-Based Network Industry Classification

The most related researches to our study are Text-Based Network Industry Classification (TNIC) developed by Hoberg and Phillips (Hoberg and Phillips 2010a) and firm's proximity measurement developed by Zhan Shi, Lee and Whinston (Shi, Lee, and Whinston 2016).

TNIC is a time-varying industry classification based on product overlap of firm's business descriptions filed within the SEC. The network is based on the premise that product similarity is a core identifier to classify industries. Companies are network nodes, and we connect two companies based on their product and business similarity. Each firm can be represented as a binary vector using the Bag-of-Words model.

Compared to conventional industry classification methods, TNIC has three significant advantages: First, it is time-varying hence good at capturing the firm's products and industry changes. Second, it provides a measurement of both within-industry heterogeneity and cross-industry similarity. Third, it is based on text analysis algorithms hence contains very little inconsistency caused by manual intervention. TNIC has been used in many empirical studies to identify peer firms (Dougal, Parsons, and Titman 2015; Foucault and Fresard 2014) or extract information like industry-level competition and product market threats (Hoberg, Phillips, and Prabhala 2014; Li, Lundholm, and Minnis 2013; Valta 2012). Instead of classifying companies into industries, Shi, Lee and Whinston used Latent Dirichlet Allocation (LDA) model (D. M. Blei, Ng, and Jordan 2003) to develop a continuous measure of the firm's business proximity (Shi, Lee, and Whinston 2016). The LDA models represent each firm's textual description as a probabilistic distribution over a set of underlying topics, which can be interpreted as aspects of its business(Shi, Lee, and Whinston 2016). These two studies share many commonalities: both used a text mining approach to analyze firm's business description; both developed a continuous measurement of firm's business closeness; both are machine-based algorithms hence will include little inconsistency caused by human invention. While unlike TNIC first extracted nouns as a firm's products and mapped the company into the product space, Shi used LDA to construct "Topics" and mapped the company into the high-dimension topic space. These topics are not limited to product information, but also contain other information like market and technology. Both measurements are proved to exhibit a higher in-industry/sector homogeneity and are good at predicting firm's mergers and acquisitions (M&As) (Hoberg and Phillips 2010b; Shi, Lee, and Whinston 2016).

However, both measurements have some drawbacks from a document retrieval perspective. TNIC uses the Bag of Word model (BOW) to present product features and calculate firms' pairwise similarities. The BOW encodes every word in the vocabulary as a one-hot-encoded vector. It ignores the context as well as the semantic meaning of the individual words. Moreover, the BOW model is proved not good at measuring document similarity due to frequent near-orthogonality problems (Greene and Cunningham 2006; Schölkopf et al. 2002), which means there is a high correlation between words in the BOW presentation. As for the LDA model, the major problem comes with time costing. LDA is a static unsupervised model, which means we have to run the model repeatedly when new data arrives, and the underlying topics will keep changing whenever we run the model, hence leading to inconsistent topics when we measure the industry in the topic space. Additionally, the number of topics is predefined, and the underlying Dirichlet topic distribution ignores the correlation between topics (D. Blei and Lafferty 2006), bringing many uncertainties into the model. Considering all these limitations, more sophisticated document retrieval methods should be used to build the industry network.

In this paper, we propose to use Word Embedding and Word Mover Distance to vectorize the representation of words and consider document similarity as an optimization problem. Our solution should avoid the near-orthogonality problem. Using the word-embedding method, we could extract more affluent firm and industry features from how the companies express themselves.

Data

Data are gathered from the EDGAR database for fillings that appear as “10-K”, “10-K405”, “10KSB”, or “10KSB40” from 1996 to 2015. We then link 10K data to COMPUSTAT using the unique SEC firm identifier, the central index and COMPUSTAT gvkey. We use a regular expression based on Python text-parser to extract Item1 (Business Description Section) from 10-K reports. Companies’ Item1 contains less than 20000 characters are excluded from our sample. To be analogous with Hoberg and Phillips' dataset for later comparison, we also remove firms not included in COMPUSTAT database, firms with non-positive sales, firms with assets of less than $1 million.

Finally, our sample contains 155353 unique company-fiscal year pairs with firm fiscal years ending from 1996 to 2015. It covers 93% of companies in Hoberg and Phillips network, hence it can compare directly with theirs.

Research Methodology

Word Mover Distance

Word Mover's Distance (short for WMD) (Kusner et al. 2015) is a novel distance function to calculate document similarity. It was introduced by Matt Kusner, Yu Sun, Nicholas Kolkin and Kilian Weinberger in 2015. The WMD distance measures the dissimilarity between two text documents as the minimum amount of distance that the embedded words of one document need to "travel" to reach another document’s embedded words. Compared with the Bag of Words model (BOW), WMD avoids frequent near-orthogonality and will capture the distance between individual words in the documents.

Instead of using Euclidean Distance and other bag-of-words based distance measurement, the WMD function proposed to use word embeddings to calculate the similarities, which is an efficient way to transfer word into a highly informative feature vector. The representative word embedding model is word2vec (Mikolov et al. 2013), it trains each word vector to maximize the log probability of neighboring words in the corpus:

$$\frac{1}{T}\sum_{i=1}^{T}\sum_{j\in nb(t)} log\, p(w_j|w_t)$$

Where $nb(t)$ is the set of neighboring words of word $w_t$and $p(w_{t+j}|w_t)$ is the hierarchical softmax of the associated word vectors V_{wj} and V_{wt}.

After embedded word into word vectors, the semantic relationships are presented through vector operations, like $Vec(Berlin) - Vec(Germany) + Vec(France)$is close to $Vec(Paris)$. Hence distance between individual words can be represented through distance between their word vectors.

To calculate WMD, we should first build a normalized BOW (nBOW) presentation of the documents, then we evaluate word dissimilarity by calculating the Cosine Distance in the word2vec embedding space, denote as $c(i,j) = ||v_i||_2||v_j||_2 cos\theta$, where v_i is the word vector of one word. Let d and d' be the nBOW representation of two text documents "travel" to d' is to solve a transportation problem between d and d': where $T_{i,j}$ denotes how much of word i in d to "travel" to word j in d'. The minimum cumulative cost of moving d to d' given the constraints is provided by the solution to the following linear program:

$$min_{T\geq 0}\sum_{i,j=1}^{n} T_{ij}c(i,j)$$

subject to:

$$\sum_{j=1}^{n} T_{ij} = d_j\ \forall i \in \{1,\dots,n\}$$

$$\sum_{i=1}^{n} T_{ij} = d'_j \forall j \in \{1,\dots,n\}$$

Relaxed Word Mover Distance

The best average time of solving WMD is about $O(p^3\ log\ log\ p\ ),$ while p is the number of unique words. To overcome the high computational cost of solving the transportation problem, we use Relaxed Word Moving Distance (RWMD). RWMD is calculated based on relaxing the WMD optimization problem and removing one of the two constraints respectively: for each word vector i in document d, identify the most similar word vector j in document d' and combine the distance of most similar words. Let the two solutions be $l_1(d,d')$ and $l_2(d,d')$ respectively, the RWMD solution can be:

$$RWMD = l_r(d,d') = max(l_1(d,d'), l_2(d,d'))$$

RWMD will be a little additional overhead than WMD and has a time complexity of $O(p^2)$, hence we choose RWMD as an approximate value to WMD.

Embeddings from Language Models (ELMo)

The previous word embedding method word2vec posed several problems, especially that all senses of a polysemous word had to share the same vector representation. A deeply contextualized word embeddings came into the picture. Embedding for Language Models (EMLo) (Peters et al. 2018) finds a way to capture the word meaning in that context and other contextual information. Instead of using a static embedding vector for each word, ELMo takes a look at the entire sentence to learn words and their context using 2-layer bidirectional LSTM. The ELMo LSTM would be trained on a massive text corpus, and then we extract word embedding from the pre-trained network of corresponding words in each layer of the network as new features to be added to the downstream task. The trained language model groups together the traditional word embedding, hidden layer forward representations, and hidden layer backward representations into a new weighted task representation. So, this language model has a sense of the next word as well as the previous word.

Bidirectional Encoder Representation from Transformers (BERT)

Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al. 2018) is a method that's deeply bidirectional, pre-trained language representations using a plain text corpus. BERT builds on top of the bidirectional idea from ELMo, but uses the relatively new transformer architecture (Vaswani et al. 2017) to compute word embeddings, which reads the entire sequence of words at once and takes advantage of both left and right contexts simultaneously, even ELMo concatenates both left-to-right and right-to-left information. Transformer is designed to solve sequence-to-sequence tasks while handling long-range dependencies. The concept of self-attention allows the model to take other words in the input sequence into account to better understand a word in the sequence. Moreover, self-attention is multi-head attention that is computed many times both in parallel and independently.

BERT has two steps: pre-training and fine-tuning. Pre-training BERT uses two unsupervised tasks that are Masked Language Modeling and Next Sentence Prediction (NSP) to train. In the Masked Language Model, 15% of words in sentences are masked and replaced by a specific token. Then the model predicts the original token based on other unmasked words in the sequence to learn the relationship between words. Next Sentence Prediction task is used for understanding the relationship between sentences and predicting if the second sentence in the pair is the subsequent sentence during pre-training. The model then can be fine-tuned with just one additional output layer to solve other NLP tasks with fewer data.

In our research, to evaluate document similarity, BERT was not designed to produce useful sentence embeddings that can be used with cosine similarities. Cosine-similarity treats all dimensions equally, which puts high requirements for the created embeddings. And instead of using cosine similarity, we can employ the Word Mover Distance method. Moreover, when using word2vec, our method extracts proper nouns based on the context, we can use entire documents for BERT method since BERT can successfully capture semantic information in sentences.

Latent Dirichlet allocation

Latent Dirchlet allocation (LDA) is an unsupervised topic modelling approach that can automatically identify topics in a collection of studies. Each word in each document is probabilistically drawn from the vocabulary of a topic discussed in that document. We choose Python Gensim library to implement LDA model and calculate document similarities.

Text Mining and Industry Network

Given that the 10-K document becomes increasingly essential for the firms, we find the total words of firm Business Descriptions Sections have an increasing trend along the years. Instead of using the Hoberg and Phillips's method to extract Nouns based on word case, we extract Nouns and Proper Nouns based on the context. We use an open-source Natural Language Processing library spaCy to tag Nouns and Proper

Nouns in all of our 10K samples to represent the company's product information. SpaCy's pre-trained model has two significant advantages: entity recognition and dependency parsing. Entity recognition is a task to locate and classify named entities text into predefined categories such as the name of an organization, location, quality, etc. Dependency parsing is a task to parse sentence structure and identify word relationships. We use spaCy to preprocess all the 19 years 10-K Item1 and extracted nouns with more than 500 occurrences.

To calculate firm pairwise Word Mover Distance, we also need a symmetrical matrix that evaluates word pairwise distance. Since word-vector performance is dependent on the size of the training corpus, we use a well-known word2vec pre-trained word-vector word2vec-GoogleNews-vectors. This word-vector was trained using Mikolov's skip-gram model based on Google News corpus's 3 billion words. It contains 3 million English words and each word was represented using a 300-dimension word-vector. We also change the distance function from Euclidean Distance to Cosine Distance in this step, since the former distance will smooth the difference based on such a high dimension word-vector.

We calculate firm pairwise RWMD using the pre-trained word vector using all 10-K Item1 corpus. Then we use company as node, similarity as edges to build an industry network.

Empirical Validation and Application

Across-industry Variation

Since we hold fixed the degree of granularity in the classifications, an industry classification method that generates a higher degree of across industry variation is more informative.

We use a firm-weighted approach to calculate the industry properties. First, compute the given firm's industry value of a given characteristic as the mean of the given characteristic among all its industry peers. Then, the across-industry variation is the standard deviation of these industry characteristics across all firms in a given year. Finally, we compare our word embedding method to other industry classifications such as SIC, NAICS classifications, and TNIC classifications.

Application on M&As

In this section, we will use the similarities calculated by TNIC, LDA, and RWMD methods with other financial statistical data to predict firms' Mergers & Acquisitions in U.S. stock market. We focus on the list of companies that have made (acquirer) and received (target) a takeover offer. To ensure balanced data, we generate firm pairs randomly from the acquirer with another four firms. Then we will implement Naïve Bayes, SVM, Random Forest, and Logistic Regression to make predictions and take AUC for evaluation.

Reference

Bhojraj, Sanjeev, Charles M. C. Lee, and Derek K. Oler. 2003. "What's My Line? A Comparison of Industry Classification Schemes for Capital Market Research." *Journal of Accounting Research* 41 (5): 745–74.

Blei, David, and John Lafferty. 2006. "Correlated Topic Models." *Advances in Neural Information Processing Systems* 18: 147.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning Research: JMLR* 3 (Jan): 993–1022.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/1810.04805.

Dougal, Casey, Christopher A. Parsons, and Sheridan Titman. 2015. “Urban Vibrancy and Corporate Growth.” *The Journal of Finance* 70 (1): 163–210.

Foucault, Thierry, and Laurent Fresard. 2014. “Learning from Peers’ Stock Prices and Corporate Investment.” *Journal of Financial Economics* 111 (3): 554–77.

Greene, Derek, and Pádraig Cunningham. 2006. “Practical Solutions to the Problem of Diagonal Dominance in Kernel Document Clustering.” In *Proceedings of the 23rd International Conference on Machine Learning*, 377–84. ICML ’06. New York, NY, USA: Association for Computing Machinery.

Hoberg, Gerard, and Gordon Phillips. 2010a. “Text-Based Network Industries and Endogenous Product Differentiation.” https://doi.org/10.3386/w15991.

———. 2010b. “Product Market Synergies and Competition in Mergers and Acquisitions: A Text-Based Analysis.” *The Review of Financial Studies* 23 (10): 3773–3811.

Hoberg, Gerard, Gordon Phillips, and Nagpurnanand Prabhala. 2014. “Product Market Threats, Payouts, and Financial Flexibility.” *The Journal of Finance*. https://doi.org/10.1111/jofi.12050.

Kusner, Matt, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. “From Word Embeddings To Document Distances.” In *International Conference on Machine Learning*, 957–66.

Li, Feng, Russell Lundholm, and Michael Minnis. 2013. “A Measure of Competition Based on 10-K Filings.” *Journal of Accounting Research* 51 (2): 399–436.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. “Distributed Representations of Words and Phrases and Their Compositionality.” In *Advances in Neural Information Processing Systems 26*, edited by C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, 3111–19. Curran Associates, Inc.

Peters, Matthew E., Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. “Deep Contextualized Word Representations.” *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/1802.05365.

Schölkopf, Bernhard, Jason Weston, Eleazar Eskin, Christina Leslie, and William Stafford Noble. 2002. “A Kernel Approach for Learning from Almost Orthogonal Patterns.” In *Machine Learning: ECML 2002*, 511–28. Springer Berlin Heidelberg.

Shi, Zhan, Gene Moo Lee, and Andrew B. Whinston. 2016. “Toward a Better Measure of Business Proximity: Topic Modeling for Industry Intelligence.” *The Mississippi Quarterly* 40 (4). http://search.ebscohost.com/login.aspx?direct=true&profile=ehost&scope=site&authtype=crawler&jrnl=02767783&AN=119473816&h=eeyTzjwezu%2B6cCUng7HzeHn7VGCZsKirovzme6HDrWbeovRrBsdl00BKyP9knDcF69UkcEm8vPREHT%2F8dVqZgA%3D%3D&crl=c.

Valta, Philip. 2012. “Competition and the Cost of Debt.” *Journal of Financial Economics* 105 (3): 661–82.

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Ł. Ukasz Kaiser, and Illia Polosukhin. 2017. “Attention Is All You Need.” In *Advances in Neural Information Processing Systems 30*, edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, 5998–6008. Curran Associates, Inc.