Textual Factors: A Scalable, Interpretable, and Data-driven Approach to Analyzing Unstructured Information[*]

Lin William Cong[†] Tengyuan Liang[§] Xiao Zhang[¶]

July 25, 2018

PRELIMINARY & INCOMPLETE. PLEASE DO NOT CIRCULATE.

[CLICK HERE FOR AN UPDATED DRAFT]

Abstract

Modern firms leverage on big, unstructured data, in particular texts, for originating loans, predicting asset returns, improving customer service, etc. Moreover, interpretable textual information sheds light on key economic mechanisms and explanatory variables. We therefore develop a general framework for analyzing large-scale text-based data, combining the strengths of neural network language models such as word embedding and generative statistical modeling such as topic modeling. Our data-driven approach captures complex linguistic structures while ensuring computational scalability and economic interpretability. We also discuss applications of our methodology to issues in finance and economics, such as forecasting or backfilling asset returns or macroeconomic outcomes, interpreting existing models, and creating new domain knowledge to expand the frontier of analysis.

JEL Classification: C55, C80, G10

Keywords: Big Data, Factor Models, Machine Learning, Return Predictability, Text-based Analysis, Topic Models, Unstructured Data.

[*] We thank Oliver Xie, Michael Fortunato, and Ellis Soodak for excellent research and programming assistance. Thanks also go to Shuyan Huang, Chloe Shin, Jiashu Sun, Connie Xu, and Guanyu Zhou for significant research assistance. We gratefully acknowledge funding from the Kauffman Foundation, the Kenan Institute of Private Enterprise, and the Becker Friedman Institute of Economics at the University of Chicago. Contact Authors: Cong and Liang.

[†] University of Chicago Booth School of Business. E-mail: will.cong@chicagobooth.edu

[§] University of Chicago Booth School of Business. E-mail: tengyuan.liang@chicagobooth.edu

[¶] University of Chicago Booth School of Business. E-mail: xiao.zhang@chicagobooth.edu

1 Introduction

In recent years, researchers, regulators, practitioners, and consumers have increasingly relied on “big data” and “alternative data” for various analyses. For example, financial analysts and investors used to focus heavily on firms’ quarterly earning numbers or macroeconomic variables forecasted from its own time series, but now detect market sentiment analysis using news media articles and forecast business activities using satellite pictures of parking lots; firms and statistical agencies around the globe used to produce statistical information using data collected through household and business surveys, and now start to examine data maintained by payroll providers, medical records, or even users’ search records and mobile payment transactions.

One pre-dominant form of emerging data entails unstructured data such as texts. Because social communications are primarily mediated in languages rather than statistics, and there is as much information in language data as there is in numbers, not to mention the potential interpretability texts offer. They not only enable econometricians to supplement or replace traditional survey data or numerical data, but also allows researchers to capture information that is more granular, more up-to-date (after all, earnings are not announced every week), and complement information currently produced from structured data such as past returns and financial ratios. Yet despite the emergence of an ocean of textual data that can help better answer many of the questions posed in social sciences, it has been challenging to develop econometric tools for extracting information in general settings that preserve computational efficiency and economic interpretability.

The difficulties in analyzing textual data are three-fold: first, language structures are intricate and complex, and representing or summarizing them using simple frequency/count-based approaches is highly reductive and may lose important informational content; second, textual data are high-dimensional and processing a large corpus of documents is computationally demanding; third, there lacks a framework relating textual data to sparse regression analysis traditionally used in social sciences while maintaining interpretability.

We tackle these problems by drawing insights and strengths from both neural network models for natural language processing and topic models in statistical machine learning. In particular, we develop a framework to summarize and analyze textual data in a way that preserves the informational structure (syntactic and semantic) encoded in natural languages, ensures computational scalability and economic interpretability, and relates to linear regression models commonly used in social sciences. We then demonstrate the efficacy of the textual factors generated and apply them to the analysis of financial time series and cross section, information transmission, and the interpretation and construction of explanatory variables.

Specifically, we first generate textual factors in three steps. We start with a continuous vector embedding of each word in a large vocabulary using neural networks (word2vec) to construct the semantic and syntactic links of words in the texts. This essentially captures the rich information and complex language structure in texts. We then cluster words based on their vector embedding using a large scale approximate nearest-neighbor search (Local-Sensitive Hashing), which further reduces the dimensionality of our textual universe and groups close vector representations for easy interpretation. Finally, we overlay a topic model to enhance interpretability by describing frequency distributions over textual factors and over the supporting words within each factor.

We also discuss how our framework allows a combination of domain-knowledge-guided factor seeding with data-generated textual factors, as well as task-specific textual factor selection to reduce feature noise in financial data.

It takes years for people to master a language (and can still get lost in miscommunication), therefore one cannot over-estimate the difficulty in parsing millions of documents and synthesizing the information content in texts into quantitative features. Therefore, one cannot relegate computational efficiency to the backseat when developing a framework for textual analysis. At the same time, interpretability of outputs is crucial in social sciences. Our key innovation is to use link vector representation and topic modeling using clustering methods development in theoretical computer science. This speeds up the processing of textual data and renders the textual factors generated more interpretable than many existing approaches.

Arguably, pre-defining dictionaries or manually adding labels as done in previous stud-

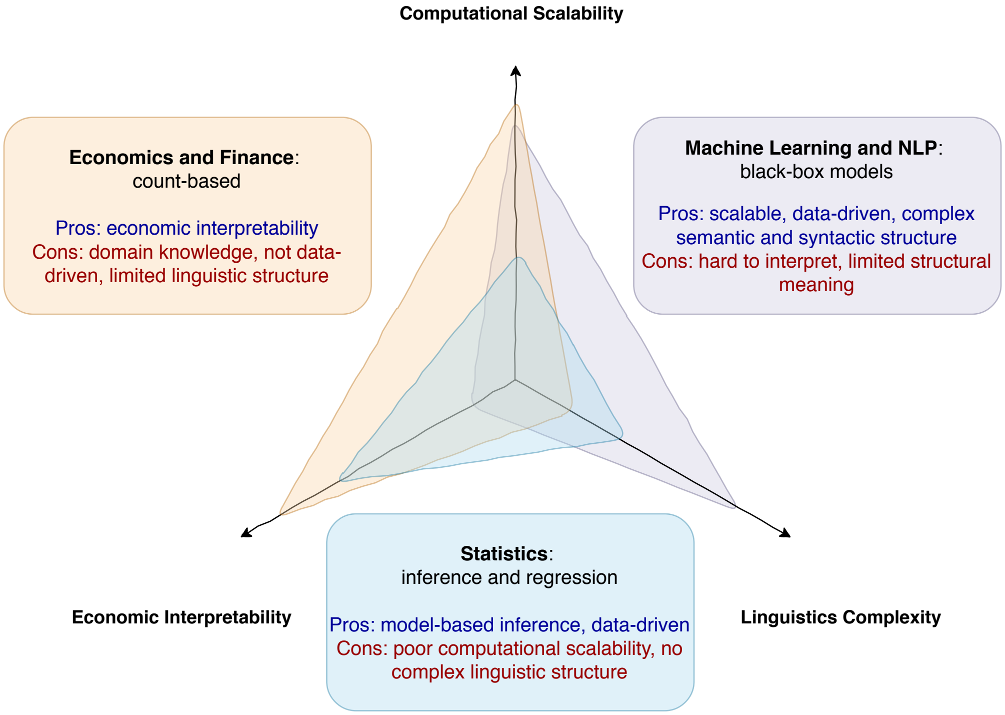
Computational Scalability
Economics and Finance: count-based
Pros: economic interpretability
Cons: domain knowledge, not data-driven, limited linguistic structure
Machine Learning and NLP: black-box models
Pros: scalable, data-driven, complex semantic and syntactic structure
Cons: hard to interpret, limited structural meaning
Economic Interpretability
Statistics: inference and regression
Pros: model-based inference, data-driven
Cons: poor computational scalability, no complex linguistic structure
Linguistics Complexity

Figure 1: Tradeoffs in Various Approaches

ies achieves computational efficiency and interpretability, and could perform rather well. However, such an approach is task-specific and requires domain expertise. It is thus not generalizable or flexible as an analytics tool for studying a wide range of problems. It achieves scalability once variables are guided or constructed by the researchers. But to select the right model and construct the variables, researchers also have searched over a complex space which is computationally expensive, and domain knowledge takes years to accumulate. In that regard, we provide a complementary data-driven approach that does not require specific domain knowledge. Moreover, to the extent that domain knowledge is distilled from historical data and earlier incidents, our textual-factor approach utilizes unsupervised learning to directly generate new domain knowledge from the data. We recapitulate various challenges in existing methods in Figure 1.

Having laid out the foundations of the proposed methodology, we move on to demonstrate its effectiveness. In particular, we discuss in Sections 4 two simple examples highlighting the interpretability of textual factors, and the convenience and effectiveness of the analysis using their loadings. To justify the quality of the topics/clusters obtained, we first compare with the plain-vanilla LDA, and print-out the support of the word clusters. The gain in interpretability is apparent. We also test the robustness and sensibility of loadings on our constructed textual factors. Specifically, we inspect the trends of loadings over time from 1900-2000 on WSJ data, for representative clusters such as "Recession", "War" and "Computer". Last but not the least, the computational advantage over the plain-vanilla LDA approach is the other key improvements besides the interpretability gain.

We then elaborate on various domain applications in economics and finance in detail in Section 5. Specifically, we analyze (i) regulatory filings, such as firms' IPO Prospectus, Annual Report (10K), Quarterly Report (10Q), Current Report (8K), (ii) analyst reports from equity researchers, (iii) credit reports from credit rating agencies, (iv) conference call transcripts, (v) news from reputable sources, (vi) FOMC announcements and (vii) social media data. We compare the performance of the textual-factor approach with those in earlier studies, in terms of information extraction, explanatory or predictive power, and economic interpretability.

Doing so allows us to illustrate the flexibility and efficacy of our framework along three dimensions. First, textual factors can help predict or explain outcomes in cross-section, time series, and panel data analysis. Second, they more clearly illustrate information transmissions and offer interpretation of existing explanatory variables constructed from structured data, such as Fama-French 3 factors. Finally, they allow a data-driven way for constructing explanatory variables with economic interpretations.

The last dimension also points to the possibility for textual factors to create new domain knowledge and opens new frontiers of analysis. For example, using structured data such as revenue and user base to value start-ups has been challenging because most early projects do not generate stable or positive cash flow, and their valuation largely depends on investors' beliefs and perception. In contrast, information extracted from unstructured data in news, forum discussion, user feedback and ratings can provide meaningful and more accurate insights into start-up's valuation. Rather than making ad-hoc and arbitrary assumptions to model future cash flow, users' view towards services or products provided by start-ups could predict its firm value more accurately. Ultimately, how much a start-up worth depends on its users' willingness to pay so information from unstructured data could potentially be more useful than analysts' often over-optimistic forecasts (Gompers, Gornall, Kaplan, and Strebulaev (2016)) or investors' often erroneous valuation templates (Gornall and Strebulaev (2017)).

Literature — Our paper foremost contributes to text-based analytics in social sciences. Gentzkow, Kelly, and Taddy (2017), Evans and Aceves (2016), and Grimmer and Stewart (2013) survey text-based analysis in economics, sociology, and political science. In particular, Gentzkow, Kelly, and Taddy (2017) point out that new techniques are needed to deal with the large-scale and complex nature of textual data.

Early attempts to utilize textual information in economics and finance are mostly count-based and prediction-focused. For example, Antweiler and Frank (2004) and Tetlock (2007) use Naive Bayes algorithm and supervised word-counting approaches to predict stock prices from the Internet or news articles.[1] More recent studies go further and use text to estimate parameters in structural models or infer causal relationships. For example, Gentzkow and Shapiro (2010) estimate news outlet's political slant and analyze the equilibrium of news slant; Engelberg and Parsons (2011) separate the causal effect of news on stock prices from other correlations. These supervised learning approaches, which are easy to interpret economically, often are task-specific, may lose informational content of the data (the structure of domain knowledge, in statistics language), and are typically computationally expensive.

For textual analysis of documents that contain a variety of topics or significant variations in information content across topics or factors, generative models such as the Latent Dirichelet Allocation (LDA) are used in recent studies. For example, Bellstam, Bhagat, and Cookson (2016) measure corporate innovation using Topic modeling; similarly, Jegadeesh and Wu (2017) quantify the economic and policy content of the Federal Reserve communications and their impact on financial markets. Still, this approach usually does not scale well when facing textual datasets with millions of words, and cannot capture complex linguistic structure within a computationally reasonable budget.

[1] Other "bag-of-words" applications include Hanley and Hoberg (2010) and Loughran and McDonald (2011), among others.

Recent development in the natural language processing literature (NLP) presents an alternative: cutting-edge machine learning techniques – for example, neural networks language models – preserve the syntactic and semantic structure well with computational tractability. However, these models are far less transparent which limits their direct usage in social sciences, which emphasize inference and interpretability. To our best knowledge, our framework is the first general, data-driven approach to capture rich language structure in texts, while striking the balance between computational efficiency and economic interpretability.

To illustrate its wide applicability, we elaborate on how to adapt and implement our methodology on a variety of domains in economics and finance. We compare and contrast our methods with some of those used in the most highly-cited and important paper. These include studies trying to quantify information content in financial texts (such as Hanley and Hoberg (2010) who use IPO prospectuses, Jegadeesh and Wu (2017); Hansen, McMahon, and Prat (2017) who focus on FOMC announcements, and Cohen, Malloy, and Nguyen (2016) who study changes in firms' 10K and 10Q), measuring market sentiment (e.g. Tetlock (2007); Loughran and McDonald (2011); Bollen, Mao, and Zeng (2011); Loughran and McDonald (2013)), predicting outcomes (e.g. Hoberg and Phillips (2010); Jegadeesh and Wu (2013)), and constructing, backfilling or nowcasting important macroeconomic variables (e.g. Baker, Bloom, and Davis (2016); Manela and Moreira (2017); Kelly, Manela, and Moreira (2018)).

We add to the literature in three ways. First, we provide an alternative which is as interpretable, and arguably more effective in terms of predictive and explanatory power, computation, and general applicability. Second, we allow better interpretation of previous analysis using structured data and black-box explanatory variables or algorithms. Third, we can analyze issues thus far intractable with existing approaches either due to the large nature of data quantity or limited interpretability. One salient example is the valuation of start-up companies using forum discussions, industry news, user and media reaction, and investor opinions.

2 The Textual-Factor Framework

In this section, we detail our textual-factor based framework that naturally integrates two advanced approaches in natural language processing (NLP): the statistical generative modeling approach represented by topic models (e.g., Blei, Ng, and Jordan (2003)), and the black-box machine learning approach epitomized in neural network language models (e.g., Bengio, Ducharme, Vincent, and Jauvin (2003)).

Our data-driven approach involves two stages. First, we form an interpretable set of basis for textual data that "spans" the "natural language space". In other words, we identify a small number of textual factors that explain main variations in the texts. Second, we decompose each data sample of texts to find their loadings on various textual factors, in order to make predictions or draw inferences in ways suitable for the specific application. In doing so, we are essentially projecting textual data onto meaningful basis to form quantitative measurements/explanatory features, for downstream regression tasks.

The first stage further comprises of three steps: (1) learn, in a distributed fashion, a continuous vector embedding of each word in a large vocabulary using neural networks (word2vec) to understand the semantic and syntactic meanings of words; (2) cluster words based on their vector embedding using a large scale approximate nearest-neighbor search (Local-Sensitive Hashing); (3) finally, employ topic modeling to extract out interpretable factors, based on the fully data-driven word-clusters obtained from previous steps.

The goal of the first stage is to generate textual factors to adequately represent the textual data, allow fast computation to process large data, and preserve interpretability. To this end, step (1) helps capture complex language structure and rich information in the data, and at the same time reduces the dimensionality of the analysis; step (2) ensures both computational efficiency through further reduction in dimensionality and economic interpretability through generating vocabulary supports of textual factors; step (3) enhances the interpretability by describing frequency distribution over textual factors and over the supporting words within each factor.

We detail each step next.

2.1 Word Embedding through Semantic Vector Representation

As it is the case for any textual analysis, the first step is to summarize or represent words or multi-grams in the texts. The vast majority of literature on textual analysis in social sciences adopts the "one-hot" representation: words (or N-grams) are treated as very high dimensional vectors/indices over a vocabulary with only one 1 and lots of 0's, leaving out any consideration of the semantic relations among words. The drawback of this approach is apparent: it lacks natural notions of similarity among words, and the representation is inherently sparse, high dimensional, and very noisy.

Many recent studies argue that vector-based representation exhibits both syntactic/semantic and computational advantages over the classic index-representation and count-based methods. There are two main approaches for learning semantic vector representations.

Bengio, Ducharme, Vincent, and Jauvin (2003); Mikolov, Chen, Corrado, and Dean (2013); Mikolov, Sutskever, Chen, Corrado, and Dean (2013) propose one-hidden-layer neural networks models to learn the representation (word2vec). The hidden layer (with p hidden units) encodes the vector representation $w, \tilde{w} \in \mathbb{R}^{p \times V}$[2]. Then based on *local* context windows, one aims to optimize $w, \tilde{w}$,

$$\min_{w,\tilde{w}} \quad - \sum_{\substack{i \in \text{corpus} \\ j \in \text{context}(i)}} \left\{ \langle w_i, \tilde{w}_j \rangle - \log \left(\sum_{k \in V} \exp(\langle w_i, \tilde{w}_k \rangle) \right) \right\} \quad .$$

Mikolov, Sutskever, Chen, Corrado, and Dean (2013) subsequently propose computationally efficient approximation schemes including "negative sampling" and "hierarchical soft-max" to train word2vec models, which scales well with tasks involving billions of words (Mikolov, Chen, Corrado, and Dean, 2013).

Another representation learning approach is proposed by Pennington, Socher, and Manning (2014) and is based on *global* concurrence X_{ij}. For some pre-chosen weights function

[2] Here we focus on the skip-gram model to predict its context based on a word, w corresponds to weights between input layer and hidden layer, and $\tilde{w}$ denotes the weights between hidden layer and output layer.

$f(\cdot)^3$, one optimize the weighted least squares to learn $w, \tilde{w} \in \mathbb{R}^{p \times V}$

$$\min_{w,\tilde{w}} \quad \sum_{i,j \in V} f(X_{ij}) \left(\langle w_i, \tilde{w}_j \rangle - \log X_{ij} \right)^2 \quad .$$

We directly apply these semantic vector representations obtained in the NLP literature, which account for word similarities, preserve the language structure, and reduce uninformative ambient noise, to improving textual analysis in social sciences.[4]

Specifically, for each unique word, we map it to a real-valued p-dimensional vector, where $p \ll V$. Note that the dimensionality p of the real-valued vectors is order-of-magnitude smaller than the dimensionality V of the naive atomic "one-hot" representation (about hundreds times smaller). However, as we demonstrate shortly using empirical data, the language structure and word similarities are captured accurately.

We denote the *learned embedding* as matrix $w \in \mathbb{R}^{p \times V}$. It later serves as a guidance or data-driven domain knowledge (priors) to classic interpretation-friendly statistical models, which can lower computational difficulty and improve model performance empirically, be it prediction or inference.

2.2 Scalable Clustering through Locality-Sensitive-Hashing

Building on the vector representation, we further employ state-of-the-art unsupervised learning methods to summarize the structure of the textual data. This second step is a key innovation and aims at balancing the interpretability and complexity of our model, and reducing dimensionality for computational ease. Moreover, the clusters serve as a guidance for the topic modeling step because they are essentially "educated guess" of the true topics.

The semantic vector representation has significantly reduced the dimension compared to the classic "one-hot" representation. However, to capture the language structure well, the representation is still inherently high dimensional (with p typically being few hundreds). In addition, the total number of words in the vocabulary is oftentimes very large (V at least

[3] A good choice of $f(x) = (x/x_{\max})^{3/4} \wedge 1$, see Pennington, Socher, and Manning (2014).

[4] Related are Taddy (2015) and Le and Mikolov (2014), which also apply vector-space language models to extract document attributes.

ten thousands for real applications). Since the semantic vector representation preserves similarity, a natural next step is to cluster words that are similar to each other through unsupervised learning, yielding in a data-driven way a number of "topics/clusters" that are easy-to-interpret.

That said, clustering in high dimension is notoriously hard both statistically and computationally. For most classic clustering methods, the computation complexity ($O(V^2p)$ in our case) exhibits an undesirable dependence on the number of items to cluster (denoted as V), which prevents their practical use due to poor scalability. To overcome this challenge, we build upon the latest theoretical computer science literature and apply the so-called locality-sensitive-hashing (LSH) (Datar, Immorlica, Indyk, and Mirrokni, 2004; Andoni, Indyk, Laarhoven, Razenshteyn, and Schmidt, 2015) in our setting.

Let us briefly review the ideas behind LSH. LSH returns near-neighbor information in near-linear time through constructing a family of hash functions H, which assert the similarity of items, with the following property: for a random element $h(\cdot) \in H$

$$\begin{aligned} &h(x) = h(y) \text{ with probability at least } 1 - p_1, && \text{for any } x, y \text{ such that } d(x,y) \leq d_1 \\ &h(x) = h(y) \text{ with probability at most } p_2, && \text{for any } x, y \text{ such that } d(x,y) \geq d_2, \end{aligned}$$

where the probability is w.r.t. the sampling of the hash functions.[5] Intuitively, the hash functions are good in assessing similarity, they rarely claim two items are similar when they are actually far away, nor do they conclude two close items to be not similar. By a series of so called AND-OR and OR-AND compositions of multiple hash functions, one can boost the performance of Hash function by having p_1 and p_2 very close to zero.

In semantic vector-space models, *cosine similarity* is widely used as a distance metric

$$d(w_i, w_j) = \arccos\left(\frac{\langle w_i, w_j\rangle}{\|w_i\| \cdot \|w_j\|}\right).$$

[5] By near-linear time, we mean $O(Vk)$ where k is the number of hash functions to generate the Hash table.

For cosine similarity, *random hyperplane* class turns out to be a good LSH family

$$h_v(w) = \operatorname{sgn}\left(\langle v, w\rangle\right), \quad \text{for } v \text{ sampled from unit sphere } S^{p-1}.$$

Therefore, by generating independent random directions $v_k, k \in [K]$ and compositions of $h_{v_k}, k \in [K]$, one can obtain powerful hash function with very good performance in finding near neighbors of a query word w_i, with computational time linear in V.

Build upon the LSH technique for approximate near neighbor search, we introduce the following scalable clustering Algorithms 1 and 2. Plausible choice of subroutines of the algorithm can be found in Leskovec, Rajaraman, and Ullman (2014) (See Section 3.4.3 for *lsh-cand-pairs*, *lsh-near-neigh* and Section 7.2.1 for *cluster-roid*).

Algorithm 1: Hierarchical Word Clustering based on LSH

```
Output: $K$ word clusters
Input  : number of clusters $K$; a subroutine LSH algorithm that returns word pair
         candidates that are sufficiently similar, denoted by lsh-cand-pairs; a
         subroutine cernter-finding algorithm that returns a representative point of
         the cluster, denoted by cluster-roid.
initialization: $numClusters = V \gg K$, and each cluster to be a word embedding
  vector;
while $numClusters > K$ do
    1. Run lsh-cand-pairs on all current clusters;
    (Optionally) calculate the cosine similarity over all candidate pairs to pick the
      most similar candidate pair;
    2. Pick one (best) candidate pair to merge, and combine the corresponding two
      clusters into one;
    3. Run cluster-roid to find the center of the new cluster then set
      $numClusters = numClusters - 1$;
end
```

Algorithm 2: Sequential Word Clustering based on LSH

```
Output: Word clusters
Input  : a subroutine LSH algorithm that returns approximate near neighbors of a
         query point, denoted by lsh-near-neigh.
initialization: each point to be a word embedding vector, and a sequence of points
  (ordered) to be considered pointsNotVisited;
while pointsNotVisited $\neq \emptyset$ do
    1. Take queryPoint to be the head of the pointsNotVisited;
    2. Run lsh-near-neigh based on queryPoint, save the new word cluster to be
       lsh-near-neigh $\cap$ pointsNotVisited;
    3. Take out lsh-near-neigh from pointsNotVisited ;
end
```

To demonstrate the quality of the clusters and the scalability of the methodology, we test on the 200K word vocabulary provided by Google in Section 2.6.

Clustering in high dimension with a large number of observations takes days to run even on distributed computing grids. However, in our numerical investigations, it takes a single machine less than half an hour to run.

2.3 Structured Topic Modeling: A "Clustering" View

In this step, we leverage on the clustering results obtained earlier to guide and enhance our topic model. The clustering information as an "educated guess" for the true topics reduces computation and alleviates the drawback of a plain-vanilla latent Dirichlet allocation (LDA) approach that a large portion of stop-words/common-words dominate every clusters.

As a generative probabilistic modeling (of word-counts) approach for learning topics, LDA has gained significant attention in social sciences because of its purported interpretablity and wide applicability (e.g., Hofmann (1999), Blei, Ng, and Jordan (2003), and Hoffman, Bach, and Blei (2010)). LDA proposes a hierarchical Bayesian model for the generative process of each document d. Adopting the notations from Hoffman, Bach, and Blei (2010): first, each topic $\beta_k \sim \text{Dirichlet}(\eta)$ is a multinomial distribution over the vocabulary of words.

Second, one generates a multinomial distribution over K-topics for this particular document d, denoted as $\theta_d \sim \text{Dirichlet}(\alpha)$. The word generation process for this document d is as follows: for word W_{di} in this document, sample a specific topic $z_{di} \in \{1, 2, \ldots, K\}$ with $z_{di} \sim \theta_d$, then sample the observed word $W_{di} \sim \beta_{z_{di}}$. Or equivalently, the probability of word W_{di} to be word w in dictionary is

$$P(W_{di} = w|\theta_d, \beta_1, \ldots, \beta_K) = \sum_k \theta_{dk}\beta_{kw} =: [\Theta B]_{dw} \ ,$$

where we the matrix notation $\Theta := [\theta_1, \ldots, \theta_D]' \in \mathbb{R}^{D\times K}$ and $B = [\beta_1, \ldots, \beta_K]' \in \mathbb{R}^{K\times V}$. Denote N_{dw} as the number of times word w appears in document d, and $N \in \mathbb{R}^{D\times V}$ to be the document-term matrix. From a Bayesian view, LDA aims to approximate a local mode of posterior distribution

$$\begin{aligned} P(\Theta, B|N) &\propto P(N|\Theta, B)P(\Theta)P(B) && (1) \\ &= \prod_{d\in[D]} \left(N_d! \prod_{w\in[V]} \frac{([\Theta B]_{dw})^{N_{dw}}}{N_{dw}!} \right) P(\Theta)P(B) \ . && (2) \end{aligned}$$

Here $P(\Theta)$ and $P(B)$ are Dirichlet priors. Mean-field variational inference (Blei, Ng, and Jordan, 2003; Hoffman, Bach, and Blei, 2010) is widely used to approximate the posterior distribution. The key idea is to describe the posterior by a simple mean-field distribution indexed by a set of free parameters, then minimize their KL divergence (equivalent to maximize the Evidence Lower Bound) between the truth posterior and the approximation, over the free parameters.

A frequentist view of topic modeling and LDA (Anandkumar, Foster, Hsu, Kakade, and Liu, 2012; Arora, Ge, Halpern, Mimno, Moitra, Sontag, Wu, and Zhu, 2013) offers a new perspective: given the topic-term matrix B to be some unknown parameter, the word counts N_{dw} is a multinomial with total number of N_d, and parameter $[\Theta B]_{dw}$,

$$\mathbb{E}\left[\frac{N_{dw}}{\sum_{w'\in[V]} N_{dw'}}|\Theta, B\right] = [\Theta B]_{dw} \ , \qquad (3)$$

with each row of Θ, i.e., θ_d to be i.i.d. drawn from some unknown Dirichlet distribution in $\mathbb{R}^K$. Given observed counts N, one can then estimate B. This approach conveniently connects the problem of topic modeling to non-negative matrix factorization. Spectral methods and alternating minimization approach (on matrix N and its normalized versions) have been employed to show provable guarantees for topic modeling under various separation conditions.

However, it has been observed that LDA becomes very computationally expensive on large data sets (Mikolov, Chen, Corrado, and Dean, 2013), and without principled prior choices extremely common words tend to dominate all topics (Wallach, Mimno, and McCallum, 2009). Therefore, applying LDA directly to financial or economic settings with big data could be ineffective or even misleading.

Instead, we advocate a "clustering" perspective of topic modeling. Much of the statistical and computational difficulty for topic modeling roots in the fact that LDA allows topics to have overlap in terms of words, rather than separability (or, "anchor words", meaning words that only appear in one unique topic). And without the separability of topics, it is very hard to disentangle various topics provided only the document-term matrix (in fact, it is provably NP-hard (Sontag and Roy, 2011) without additional structure when the number of topics is large). We overcome this difficulty by learning the separability of topics in a data-driven way by incorporating the semantic vector representation. In our proposed framework, we utilize the vector representation of words as guidance and enhance our topic modeling approach. Based on the semantic similarity among words captured by the vector representation, it is more likely that close-by words belong to the same topic. This prior knowledge significantly reduces the search space/complexity of the topicword distributions, therefore easing the optimization approach, while eliminating potential ambient noise.

With the word clusters obtained from Algorithms. 1 and 2, we advocate two computationally efficient and conceptually simple methods to learn textual factors, in order to avoid the computational difficulties of LDA. The key computational advantage is that given the separability of clusters, one can estimate one topic at a time. Because the vocabulary supports of topics are disjoint, the topics are distinct from each other, in contrast to the case in plain-vanilla LDA where extremely common words (or, stop words) dominate multiple topics (Wallach, Mimno, and McCallum, 2009). For instance, given the i-th cluster, with support (set of indices) $S_i \subset [V]$, we focus on the document-term submatrix N_{S_i} where the columns consist of words only in i-th cluster. For a vector v

```
Algorithm 3: Topic factor via frequency count
  Output: Topic factor $i$, represented by $F_i \in \mathbb{R}^{|S_i|}$, that satisfies $\|F_i\|_{\ell_1} = 1$, and the
          topic importance $d_i \in \mathbb{R}$.
  Input : The document-term submatrix $N_{S_i} \in \mathbb{R}^{D \times |S_i|}$
  Return $F_i = \frac{1}{\mathbf{1}^T N_{S_i} \mathbf{1}} N_{S_i}^T \mathbf{1}$;
  youtu Return the topic importance $d_i = \frac{\langle \mathbf{1}^T N_{S_i}, F_i \rangle}{\langle F_i, F_i \rangle} \in \mathbb{R}$;
```

```
Algorithm 4: Topic factor via SVD (rank-1)
  Output: Topic factor $i$, represented by $F_i \in \mathbb{R}^{|S_i|}$, that satisfies $\|F_i\|_{\ell_2} = 1$, and the
          topic importance $d_i \in \mathbb{R}$
  Input : The document-term submatrix $N_{S_i} \in \mathbb{R}^{D \times |S_i|}$
  Return $F_i$ as the top right singular vector of matrix $N_{S_i}$;
  Return the topic importance $d_i = \|N_{S_i} F_i\| \in \mathbb{R}$ (top singular value);
```

This data-driven guidance significantly enhances the performance of the topic model for unsupervised learning, as we demonstrate in the next section and Section 4.

2.4 Seeded Textual Factors

Our way for generating textual factors can easily accommodate the use of pre-existing domain knowledge. For example, if we believe certain topics or key words are important, we can seed them when we cluster vectors.

Suppose we want to use social media texts to predict credit outcomes for individuals. Using the plain-vanilla textual factor framework, we would let the entire corpus of textual information on the social networks generate the textual factors. However, experience and economics may tell us that issues related to borrowing and loans should be relevant, so are topics on income, family support, etc. To incorporate the domain knowledge, we can use “loan,” “income,” etc as seed words, and require some factors to be clustered around them, before we go three the third step of topic modeling.

The seed information fits in naturally in both the hierarchical clustering Algorithm 1 and the point assignment clustering Algorithm 2. To be specific, in Algorithm 1, one start with first considering the candidates containing seeded words before picking other candidates to merge, and one may use the seeded words as cluster-roid. In Algorithm 2, one order the sequence of words to be considered in a way such that all the seeded words appear first. Namely, the algorithm will consider first assigning words that are similar to the seeds. The final output of the clustering stage is a mixture of seeded topics (textual factors) and other remaining data-driven topics discovered. Note that our seeded textual factors contain both the word support and relative frequency of words, and therefore are more comprehensive compared to simply defining a bag of words or searching for synonyms.

We emphasize on the flexibility and importance of subsuming the seed information in our textual framework. In certain business and economic applications involving textual data and beyond, the noise is inevitably heavy-tailed and thus the signal-to-noise ratio can be relatively weak. In this situation, many state-of-the-art machine learning methods dealing with high dimensional data may suffer poor generalization result, if bluntly applied without principled understanding. The key to improve the generalization in this setting is to impose structure to rule out the irrelevant noise in the measurement. Incorporating the domain knowledge comes in as one way of imposing structure. As we shall discuss next, another way of imposing structure is through selecting the relevant task-specific factors using simple yet effective statistical methods.

2.5 Task-specific Factor Selection

In many settings, especially those concerning financial markets, the data available are extremely noisy. This issue inevitably generates spurious textual factors that reduce the explanatory, predictive power of our framework, and the interpretability of the outputs. To intuit this, one can see that a combination of a large number of pure noise features can generate high spurious correlation with any given response variable. To deal with this situation, we propose a natural approach to reduce the noise and select relevant factors in several steps.

First, we can make our textual factors more precise by removing infrequent words, or stopping words. This is what researcher has been referring to as the data-cleaning procedure, and serves as rough sieves for low signal-to-noise ratio component in the data.

Second, when we have a specific predictive or inference task in mind, we can analyze the correlation of each textual factor with the dependent variable, and get rid of the low correlation ones. One can also substitute this step with a careful model selection analysis such as LASSO or Dantzig selector, with a careful data-driven choice of tuning parameters. However, for simplicity and efficiency, we advocate using simple correlation thresholding approach. As we shall see in real-data applications found in Section 5, these approaches generate similar results.

Finally, we can use domain expertise and human judgment to remove some factors that are irrelevant. This can be done either by inspecting words and relative scores in the selected topic factors.

We *advocate the importance* of this simple model-selection procedure when applying machine learning methods to finance. Many modern high-dimensional machine learning methods such as deep neural networks and complicated kernel machines can generate fragile (non-robust) and even wrong results when blindly applied to economics and finance, due to the nature of the data. Therefore, whenever needed, we advocate the use of relevant structured features, selected by either the domain knowledge or data-driven model selection approach, before resorting to the comprehensive machine learning methods to discover the non-linear patterns. In applications shown in Section 5, we verify and evaluate our claim by studying the effectiveness in several real-data problems.

2.6 Beta Loadings on Textual Factors

From our first-stage analysis, suppose we obtain K number of textual factors, where K is endogenously specified and can potentially depend on the data. We denote the set of textual factors by the triplet $(S_i, F_i \in \mathbb{R}^{|S_i|}, d_i \in \mathbb{R}_{\geq 0})$, where S_i denotes the support of word-cluster i, a real-valued vector representing the textual factor F_i, and the factor importance d_i. Given the factors, and a new data-point (document d), represented by a document-term vector $N^{(d)} \in \mathbb{R}^V$, the loadings of the textual factor i is simply the projection

$$x_i^{(d)} := \frac{\langle N_{S_i}^{(d)}, F_i \rangle}{\langle F_i, F_i \rangle} \tag{4}$$

and the document D can be represented quantitatively as $(x_1^{(d)}, \ldots x_K^{(d)}) \in \mathbb{R}^K$.

To understand the meaning of these loadings, let us think about a publicly listed firm. Viewed through the lens of structured data, the company discloses numbers on revenues, profits, liabilities, etc. Each one has a number to it that informs how the company does along that dimension. In the land of unstructured information, texts about the company could center discussions on profitability, social responsibility, innovativeness, etc, each of which is a topic. The $x_k^{(d)}$ we obtain, again allows us to assign a number that measures how much the company loads on that topic—a metric we can use in simple sparse regression framework.

Finally, we would like to remark that one can easily generalize our result to apply to document-term matrix that include multi-grams. And in that case, one can significantly reduces the dimensionality of the multi-gram space by considering multi-grams with words in only one, or say a few topics. To see this, suppose there are V words, then bi-gram space consists of $O(V^2)$ pairs. However, if there are K clusters and by only considering bi-grams in the same cluster, one reduces the dimensionality to $O(V^2/K)$.

We next apply the framework to a variety of textual data, to illustrate the flexibility and efficacy of the methodology.

3 Data

The following non-exhaustive list of data include the ones we use to discuss applications of our framework and methodology, as well as data useful for other applications.

- Wall Street Journal. In our motivating example, We use title and abstract of all front-page articles of the Wall Street Journal from July 1889 to April 2018. The Wall street Journal data are widely used in various academic studies (e.g. Tetlock (2007); Manela and Moreira (2017); Kelly, Manela, and Moreira (2018)) and particularly suitable to our research settings. We compare our topic modeling results with historically important events to make sure our methodology captures the true topics reasonably well. We also compare our results against vanilla LDA to understand the how much improvement we could achieve. We choose to focus on front-page articles only because these are manually edited and corrected. This is particular useful for newspaper in earlier years as they are scanned and digitized using OCR (optical character recognition), which inevitably generates typos.

  Similarly, we also collect other newspapers such as the New York Times, the Financial Times and the Economists from Proquest (`https://www.proquest.com`). We are primarily interested in the Economic, Business and Finance sections of these newspaper.

  In an ongoing effort, we are manually collecting firm-specific news from Factiva (`https://www.dowjones.com/products/factiva`). These firm-specific news enable us to explore variation in texts among firms in the cross-section.

- Company Filings from SEC Edgar (`https://www.sec.gov/edgar/`). To facilitate the rapid dissemination of financial and business information about companies, the United States Securities and Exchange Commission (the SEC) approved a rule requiring publicly-listed firms to file their securities documents with the SEC via the Electronic Data Gathering, Analysis and Retrieval (EDGAR) system. This has made regulatory filings publicly available since 1993. We start with Management Discussion and Analysis (MD&A) sections of both the quarterly report (10-Q) and the annual report (10-K) and then study the informativeness of entire text documents. One can use information from other types of forms, such as IPO prospectus (S-3) and current reports (8-K) to capture firm specific events.

- Conference Call Transcript. Most publicly-traded firms hold regular conference calls with their analysts and other interested parties. During the conference call, management give its view on the firms past and future performance and respond to questions from call participants. Both conference call audio recordings and transcripts are available. We obtain conference call transcripts from SeekingAlpha (`https://seekingalpha.com/`).

- Analyst reports from Investext via Thomson One (`https://www.thomsonone.com/`). Equity analysts from major investment banks periodically write about firms' past performance and their view about firms' future stock price. We have manually collected analyst reports for more than 20 years for 700 publicly listed firms. All of these analyst reports are in PDF format so we convert them to .txt files and clean them up using a python script before analyzing them.

- FOMC Meeting Transcript (`https://www.federalreserve.gov/monetarypolicy/fomc_historical.htm`. Every year, The Federal Open Market Committee (FOMC) holds eight regularly scheduled meetings. FOMC meeting members discuss the economic outlook and formulate monetary policy during these meetings. All policy changes were made public in a short meeting statement immediately after the meeting. In addition, detailed records of the discussions during each meeting (minutes) were released a day later. We collected all texts document for FOMC meeting from federal Reserve website.

- Non-public Twitter Data: We collected all tweets from 2007 to 2016 related to Russell 3000 companies and Federal Open Market Committee (FOMC) announcements.

4 Empirical Results: Textual Factors

We illustrate the viability of our methodological framework using two examples.

First, we compare the textual factors we create with the topics generated in a plain-vanilla LDA (without guidance), and show that textual factors capture themes and topics more concisely and precisely, allowing easy interpretations. Table 1 illustrates the effectiveness of our clustering method based on LSH. In contrast, in Table 2 we display the top three obtained "clusters", or topics by plain-vanilla LDA to showcase the improvements in terms of interpretability. As we can see, extremely common words dominate each cluster, which clouds the meaning of different topics.

Second, we apply the two-stage procedure on data from the Wall Street Journal, and look at the textual factors (generated based on Algorithms 3 and 4) that are related to "computer", "war", and "recession", as a sanity check. From the plots of loadings over time, we are assured that our approach makes sense empirically because the intensity of the textual factors accurately captures the prominence of these topics in history. In addition, we emphasize that the coordination of generic patterns (trends), regardless of the specific loadings generating algorithms we use (easily seen by comparing subfigures in the same column).

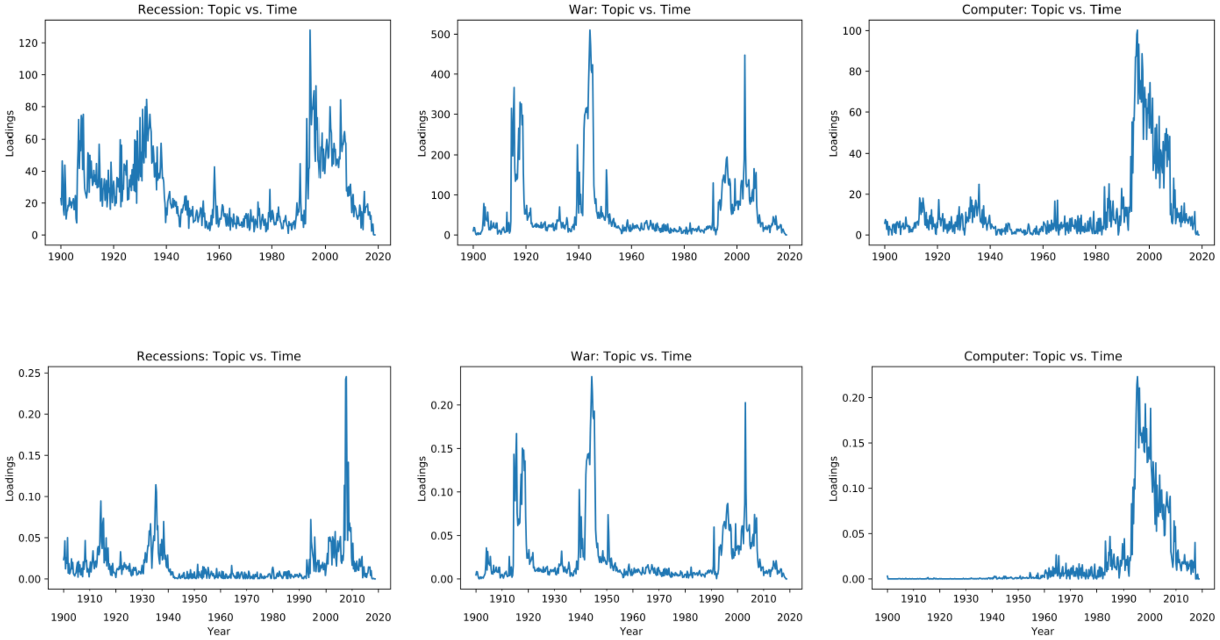

Figure 2: Loadings on textual factors over time, WSJ data. Here in the first row, the loadings are calculated based on Alg. 3, and second row based on Alg. 4. The three columns correspond to "Recession", "War" and "Computer", respectively.

Cluster	Support
Tax	quotas, visa, harvestable, import, preferential, abolished, tariffs, quota, sanction, compulsory, tariff, compulsorily, stipulating, fisheries, cess, exports, pricing, export, telcos, exporters, import, liberalization, preferential, excise, tariffs, tax, tariff, importers, deregulation, antidumping, subsidy
Oil	refiners, refiner, refineries, refinery, petrochemical, feedstock, refiners, pipelines, smelters, crudes, oil, bpd, gasoline, refiner, petrochemicals, petroleum, refining, ethanol, refineries, tankers, refinery, coker, petrochemical, ethylene, feedstock, crude
Unemployment	stimulus, foreclosures, recession, claimants, workweek, unemployed, housing, unemployment, jobless, economy, workers
Volatility	correction, uptrend, readjustment, reversal, retest, revision, divergence, retrenchment, steepening, selloff, rebalancing, bearish, pullbacks, corrective, correcting, reversion, stabilization, selldown, snapback, reassessment, volatility, pullback, bull, corrections, bottoming, downtrend
Exports	consignments, foodstuffs, exports, tins, cargo, goods, warehouses, equipments, importers, exporting, containers, tonnages, exporters, import, imports, perishable, cartons, cargoes, export, adulterated, tankers, pallets, wholesalers, demurrage, customs, transporters, consignment, consignee, exported
Investment	development, capitalization, differentiation, invest, macro, optionality, strategic, capex, macroeconomic, countercyclical, investments, investing, outperformance, diversification, equity, arbitrage, diversify, cyclicality, underperformance, diversifying, expansion, diversified, geographies, reinvest, specialization, profitability, deleveraging, consolidation, renewables, volatility, investment, liquidity, growth, maximization, sector, cyclical, synergy, reinvesting, investors, reinvestment
Stimulus	appropriation, moneys, underfunded, money, reauthorization, subsidies, budget, fundings, budgeted, allocations, budgets, budgetary, stimulus, funded, appropriations, funds, grant, nonfederal, appropriated, earmarked, infrastructure, reauthorized, assistance, unfunded, funding, financing, grants, monies, support, underfunding
Disasters	disturbances, occurance, instances, recur, disasters, incidences, occur, occurence, occurrences, causes, occurred, occurrence, phenomenon, earthquakes, anomaly, outbreaks, accidents, incidents, emergencies, observations, tragedies, ultramafic, catastrophes, polymetallic, anomalous, occurence, outbreaks, disturbances, incidences, occur, calamities, occurrences, infrequent, phenomena, anomalies, occurance, happening, intrusions, contaminations, occurrence, occurring, incidents
War	battles, confrontation, dispute, fighting, showdown, struggle, fight, battle, wars, fierce, war, battles, confrontation, showdown, matchups, fight, battle, victory
Election	political, intellectual, politically, election, politicians, democratic, religious, republican, incumbency, diplomatic, politics, economic

Table 1: Sample clusters based on our methodology in Section 2.2

Cluster	Support
Topic ID: 62, Prob: 0.20071%	**washington**, tax, business, york, labor, letter, bulletin, wire, report, old, many, big, **president**, like, long, economic, **price**s, time, ago, federal, outlook, city, get, high, sales, white, house, back, people, even, state, just, home, world, much, **american**, man, next, **government**, job, million, still, work, companies, workers, economy, men, three, little
Topic ID: 1272, Prob: 0.17438%	stock, dividend, steel, business, **american**, oil, common, market, york, **earnings**, months, outlook, **cent**s, made, record, way, chicago, share, company, united, net, time, **president**, rate, **price**s, increase, railroad, states, june, **price**, **general**, review, shares, declared, july, report, cotton, preferred, sales, **washington**, present, large, month, regular, production, exchange, pacific, cars, quarterly, september
Topic ID: 1828: Prob: 0.11747%	steel, states, business, united, outlook, review, railroad, stock, way, market, york, country, time, **president**, great, made, **american**, **price**s, copper, increase, **earnings**, corporation, public, **government**, per, national, **general**, since, **washington**, cotton, crop, bank, report, months, state, much, commission, present, **cent**, railroads, rate, conditions, **price**, large, street, ago, letter, pacific, trade, three

Table 2: Sample plain-vanilla LDA clusters.

5 Applications in Economics and Finance

We now apply our methodology to problems including but not restricted to the ones studied in several classical articles in finance and economics. We compare its performance to that of earlier approaches. Our selection of the articles follows a two-fold criteria: first, we aim to cover a wide range of topics involving the use of text data; second, we aim to include all widely-adopted existing approaches to analyzing textual data.

5.1 Cross-section Inference and Time-series Predictions

Stock Returns and Volatility

Tetlock (2007) adopts a dictionary-based approach to analyzing the role of media. Using 77 predetermined categories from the Harvard psychosocial dictionary, he counts the keywords from the Wall Street Journal's Abreast of the Market column and constructs a time-series sentiment score by performing principal components analysis. He finds that negative media sentiment predicts downward pressure on market prices followed by a reversion to fundamentals. This approach heavily weights the priors and is most suitable when prior is strong and reliable. Therefore, many context-specific predefined dictionaries become available for researchers. Loughran and McDonald (2011)) construct a finance-specific dictionary of positive and negative terms and shows that the predictive power improves using financial texts. Bollen, Mao, and Zeng (2011) uses OpinionFinder and Google's Profile of Mood States to measure sentiment in Twitter messages and shows its correlation with stock market returns.

Whether dictionary-based methods perform well is an empirical question and depends on applications. Jegadeesh and Wu (2013) use a text regression in a similar framework and compare its result with dictionary-based methods. They also study whether text information in the annual report (10k) can predict firms' stock returns. They find that using term weights estimated via regressions can improve out-of-sample performance more than refined dictionary-based indices from Loughran and McDonald (2011)). Text regression also suitable for other many finance applications. Manela and Moreira (2017) use support vector machines approach to construct news-implied market volatility using the Wall Street Journal. Their approach could identify a small set of words which are useful in predicting market volatility.

IPO Prospectus and Underpricing

To understand the information content of IPO prospectuses, Hanley and Hoberg (2010) decompose the texts into standard and informative components and show that greater informative contents lead to less underpricing. They achieve this by estimating the similarity of an IPO prospectus to a "boilerplate" texts constructed using S-1 filings that either recent or issued by firms in the same industry. Most paper in this area follows this methodology or used dictionary-based methods. For example, Cohen, Malloy, and Nguyen (2016) compute similarity scores of firms' 10-k filing across years. They show that when firms make an active change in their reporting practices, this conveys an important signal about the firm. Loughran and McDonald (2013) use a word list to measure the tone and definitiveness of S-1 filing to study the effect of language choice on IPO returns, price revisions, and subsequent volatility.

Mergers and Acquisitions

Hoberg and Phillips (2010) (*Review of Financial Studies*) analyzes how similarity and competition impact the incentives to merge and whether mergers with potential product market synergies through asset complementarities add value.

Specifically, this paper examine 3 hypotheses: 1. Asset Similarity: Firms are more likely to merge with firms whose assets are highly similar or related to their own assets. 2. a. Differentiation from Rivals: Acquirers in competitive product markets should be more likely to choose targets that help them to increase product differentiation relative to their nearest ex-ante rivals. b. Competition for Targets: Firms with high local product market competition are less likely to be targets of restructuring transactions given the existence of multiple substitute target firms. 3. Synergies through Asset Complementarities: Acquirers buying targets similar to themselves are likely to have asset complementarities and experience future higher profitability, sales growth, and new product introductions.

The paper uses 49,408 (fiscal years 1997-2006) 10-K product descriptions obtained from the Securities Exchange Commission (SEC) Edgar website. The paper uses logistic models to test whether firms are more likely to merge when they are broadly more similar to other firms (Hypothesis 1) and when they are locally more similar to their nearest rivals (Hypothesis 2b). To test Hypothesis 3, the authors 1. Examine announcement returns using OLS regressions with the acquirers and targets combined abnormal announcement return as the dependent variable. 2. Examine real performance by OLS regression with the acquirer's change from year $t+1$ to $t+2$, $t+4$ in: (1) industry-adjusted operating income dividend by assets, (2) industry-adjusted operating income dividend by sales, (3) industry-adjusted sales growth. 3. Proxy for new product development by the logarithmic growth in the number of words used in the product market description from year $t+1$ to either year $t+2$ or $t+4$, and use it as dependent variable in OLS.

One can use textual factors to predict product similarity and outcomes associated with mergers and acquisitions.

5.2 Interpretation and Information Transmission

Z-score, F-score, M-score

Asset Pricing Factors

Take a traditional factor model in asset pricing, for example, the four factor model of Fama-French 3 factor plus momentum. Let us explore if textual data provide information about factor risk premium and firms' beta loadings.

$$R_{it}-Rf_t = \alpha+\beta_i^{Market}(R_{Mt}-Rf_t)+\beta_i^{Size}SMB_t+\beta_i^{Value}HML_t+\beta_i^{Momentum}UMD_t+\epsilon_{i,t}, \quad (5)$$

where $R_{it}$is the stock return of firm i in month t, Rf_t is the risk-free return; R_{Mt} is the return on the value-weight market portfolio; SMB_t is the return on a diversified portfolio of small stocks minus the return on a diversified portfolio of big stocks; HML_t is the difference between the returns on diversified portfolios of high and low B/M stocks; UMD_t is the

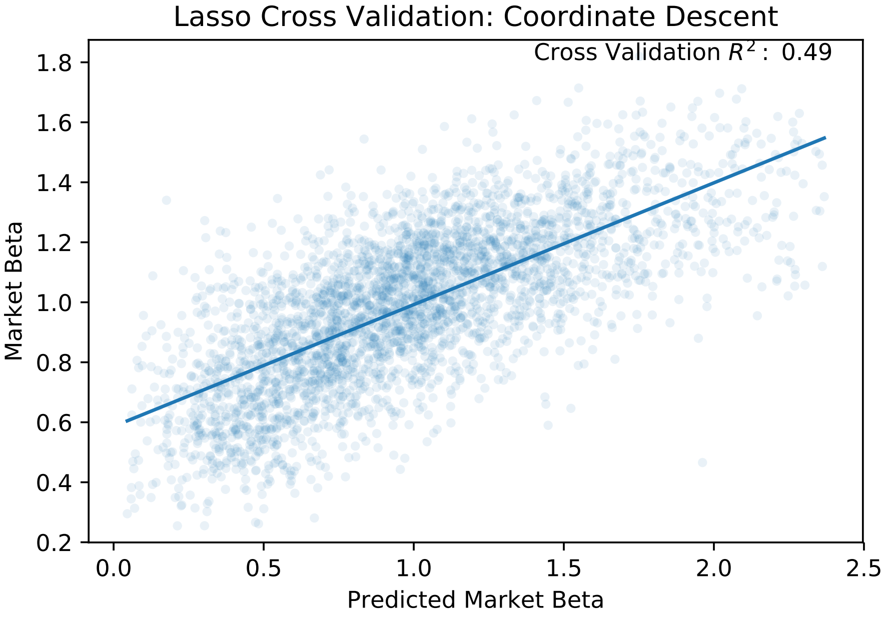

Figure 3: Market Beta

difference between the returns on diversified portfolios of high and low return stocks.

MD&A is portion of a public company's annual report in which management addresses the companys performance over the previous twelve months. We aggregate that to the firm level, and then use that to explain the cross section of firms' beta loadings on the four factors.

Our textual factors have rather significant predictive power of beta loadings of conventional asset pricing factors, for example, in the cross-validation exercises presented below.

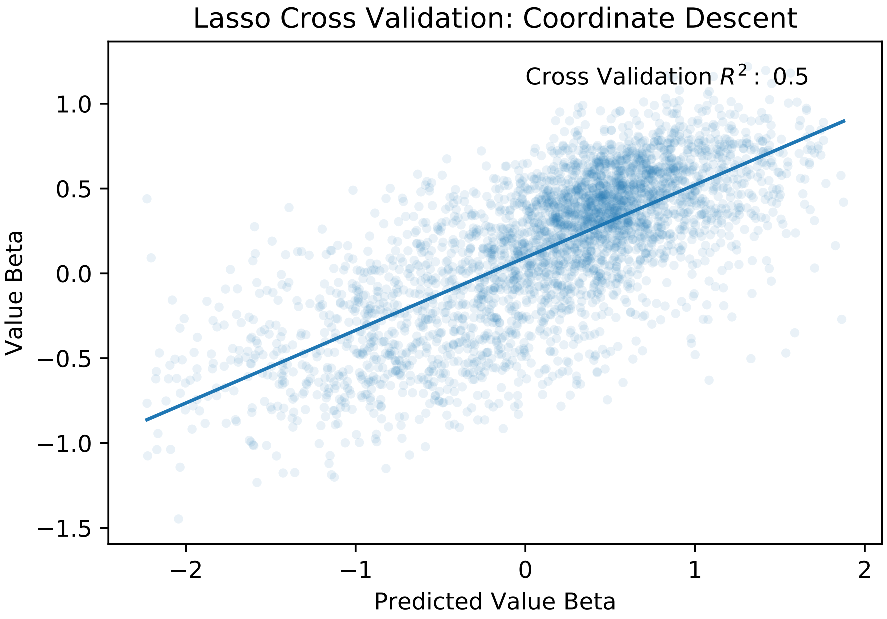

Figure 4: Value Beta

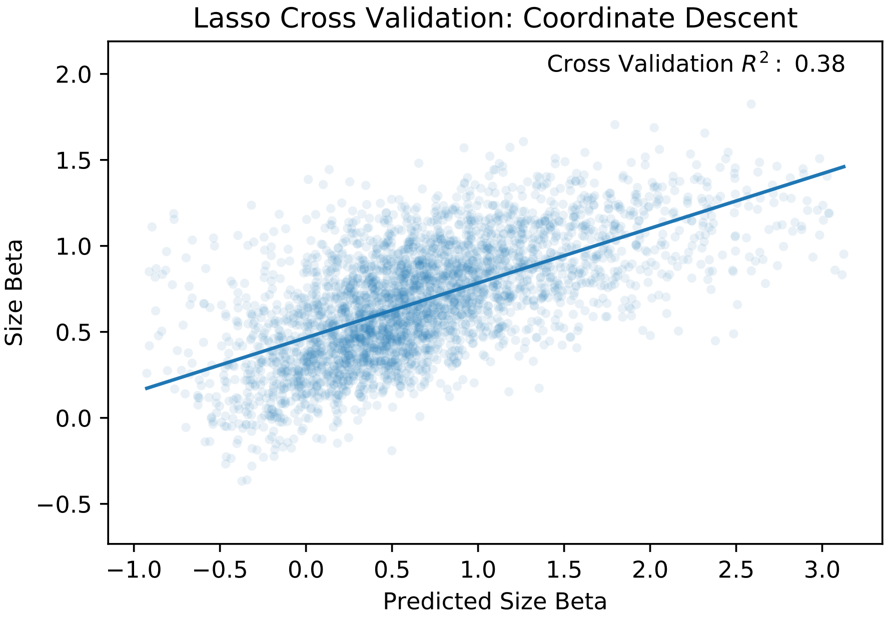

Figure 5: Size Beta

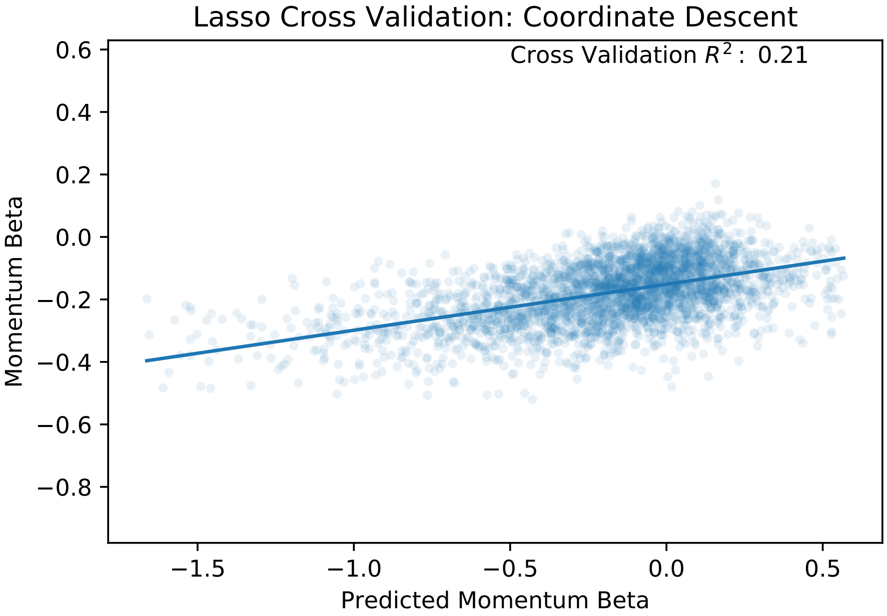

Figure 6: Momentum Beta

Market Sentiment

Garcia (2013) (*Journal of Finance*) asks to what extent and when can financial news sentiment be used to predict stock returns? More specifically, the literature from psychology and economics suggests that investors are most sensitive to news during periods of economic hardship. The paper tests this phenomenon, carefully distinguishing the effect of news sentiment from new information.

Garcia (2013) measures sentiment using the reaction of positive and negative words in two columns of financial news in the NY Times. The author disentangles the effect from new information vs the effect from sentiment by doing the following: the effect from news partially reverses after a few days, with more than half of the drift disappearing after 4 days. This lends credence to the sentiment interpretation of the columns effects on the markets over the new information interpretation. Garcia (2013) looks at intraday predictability. He argues that if the effect were informational, then after the information is incorporated into the price, the columns would cease to be predictive later into the day. He finds that the predictability is maintained throughout the day, well after the NYSE opens.

The author finds that the predictability of stock returns using financial news sentiment is strongest during recessions. One standard deviation change in the sentiment measure predicts a change in the daily average of the DJIA of 12 basis points during recessions compared to 3.5 basis points during expansions.

One can use textual factors to find a good proxy for sentiment, and then tests how that helps predict intraday price changes.

Macroeconomics and Information Transmission

Many macroeconomic variables are measured at low frequency and released with lags; some other macroeconomic variables are hard to measure, even with surveys. Text data turns out to be a fruit venue to estimate these quantities. Baker, Bloom, and Davis (2016) use a dictionary-based method to develop a new index of economic policy uncertainty based on newspaper coverage. They count the keywords such as policy, uncertainty, and Federal Reserve, in a given newspaper-month and construct a proxy for policy uncertainty in the economy. They also perform a careful manual audit to validate their approach and shows that their simple method yields consistent results.

Central bank communication is another area in which textual analysis contribute a lot to finance research. Lucca and Trebbi (2009) use Google and Factiva searches to determine the sentiment (hawkish vs. dovish) of Federal Open Market Committee statements. The semantic orientation of each sentence is measured by the relative frequency with which the sentence and the word hawkish (or dovish) jointly occur in search engine results. They find that interest rates react to changes in communication around announcements.

Unsupervised machine learning methods, such as topic modeling, is also used to study FOMC documents. Jegadeesh and Wu (2017)) use latent Dirichlet allocation (LDA) to study the information content of the Federal Reserve communications. They dissect the FOMC minutes into eight distinct economic topics and examine the informativeness of the Fed's discussion of each of these topics for the stock market and for interest rates. Hansen, McMahon, and Prat (2017) also use LDA to study FOMC meeting transcripts during Alan Greenspans tenure. They find that inexperienced members discuss a wider range of topics and make more references to data when discussing economic conditions in a more transparent era. The results support the notion that transparency leads to greater accountability.

5.3 Construction of Explanatory Variables

Competition, tone, similarity, financial constraints, innovation. These are intermediate variables that have sound economic theory backing. Can we better construct such metrics? Our methodology provides a data-driven, and universally applicable approach, as compared to ad hoc pre-defined constructions.

Firm and Industry Characteristics

Hoberg and Maksimovic (2014) develop a text-based measure of financial constraints by analyzing the Management's Discussion and Analysis (MD&A) section in annual reports. They start with a dictionary-based method and identify firms that are deemed financial constrained. Then they calculate the cosine similarity of each firm's annual report to those firms to obtain a continuous measure of financial constraint. Comparing with Tobins Q and other existing measures, they show that their text-based measure capture incremental information about firms' financial decision.

Using a similar approach, Hoberg and Phillips (2010) classify industries based on product descriptions in annual reports. They calculate the cosine similarity between texts across firms and years. This generates a flexible industry classification that could change over time. This is different from the traditional type of industry classifications, such as SIC and NAICS, and allow the authors to answer questions related to industry competitions.

Backfilling VIX Index

Kelly, Manela, and Moreira (2018) propose an improved version of text regression. They combine Heckman (1979) selection model with Taddy (2015) distributed multinomial regression. In this framework, word count provides useful information in two dimensions. First, whether a word appears or not conveys information. Second, conditional on appearance, the word frequency also helps to predict outcomes. Using this methodology, they backfill and forecast some macroeconomic variables that were not available either historically or immediately.

Motivated by Manela and Moreira (2017), we attempt to backfill VIX index using title and abstract of all front-page articles of the Wall Street Journal from July 1889 to April 2018. We obtain VIX data for 1990 to 2016 from Chicago Board Options Exchange (CBOE). We also use VXO data from 1986 to 1990 as a substitute for VIX index so that we can have a longer sample.[6] Using VIX data from 1996 to 2016 as our training sample, we estimate the following model:

$$VIX_t = \alpha + \gamma x_t^T + \eta_t \tag{6}$$

, where VIX_t is VIX index in month t and x_t^T are the loadings on 200 topics estimated by LDA using Wall Street Journal data in the same month. To reduce dimensionality, I apply

[6] While VIX and VXO indices are not the same, they are 0.99 correlated in post-1990 sample. Using VXO data only does not change our results.

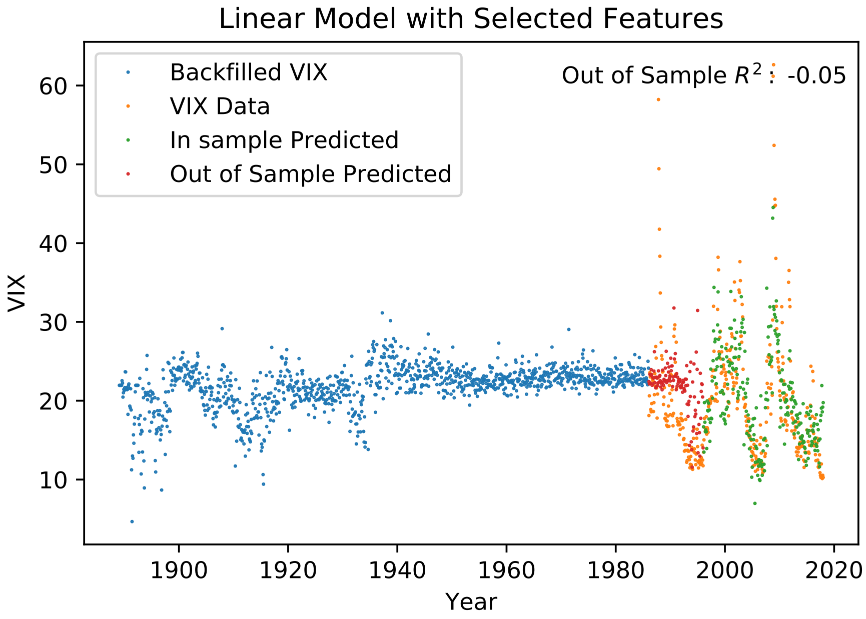

Figure 7: Backfilling VIX Index

LASSO penalization to estimate (6). This would allow us to identify topics that are most useful in backfilling VIX index.

Manela and Moreira (2017) approach this problem by using support vector machine (SVM), which is an alternative methodology to select a relatively small number of variables and ignoring the rest. In their regressions, x_t^T is the vector of word frequencies. This approach aims to predict outcome variables by exploiting variations in word frequencies. Our method, in contrast, model the topic first and then use information of learned topic to predict outcome variables. The rationale of our methodology is to reduce noise in text and hope that this could help to make better prediciton. Ultimately, which method is better is an empirical question and depends on specific application. Therefore, it is important to evaluate out-of-sample fit against alternative methodology.

5.4 Backfilling Expectation Error

In this section, we attempt to backfill an another important economic variable, expectation errors of future credit spreads. This choice is motivated by theoretical work on credit

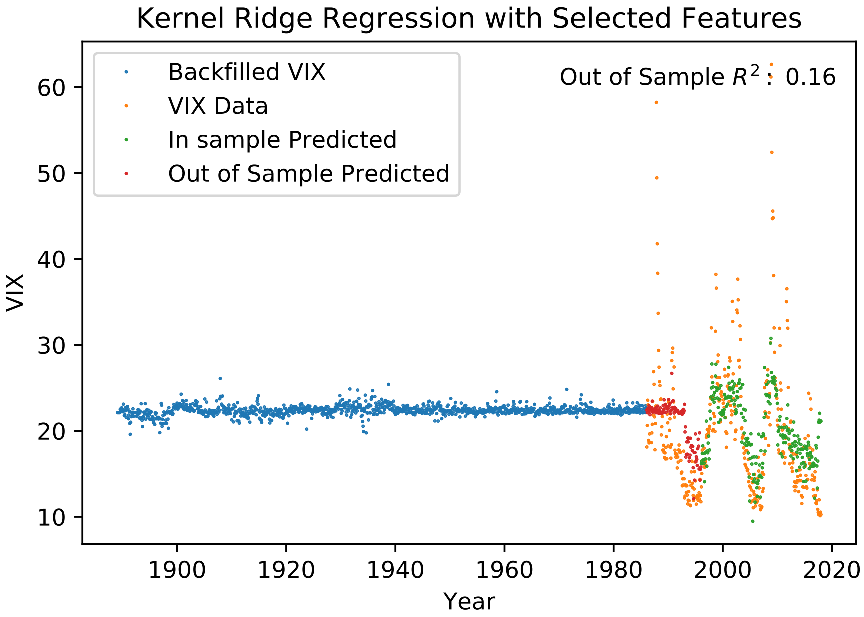

Figure 8: Backfilling VIX Index

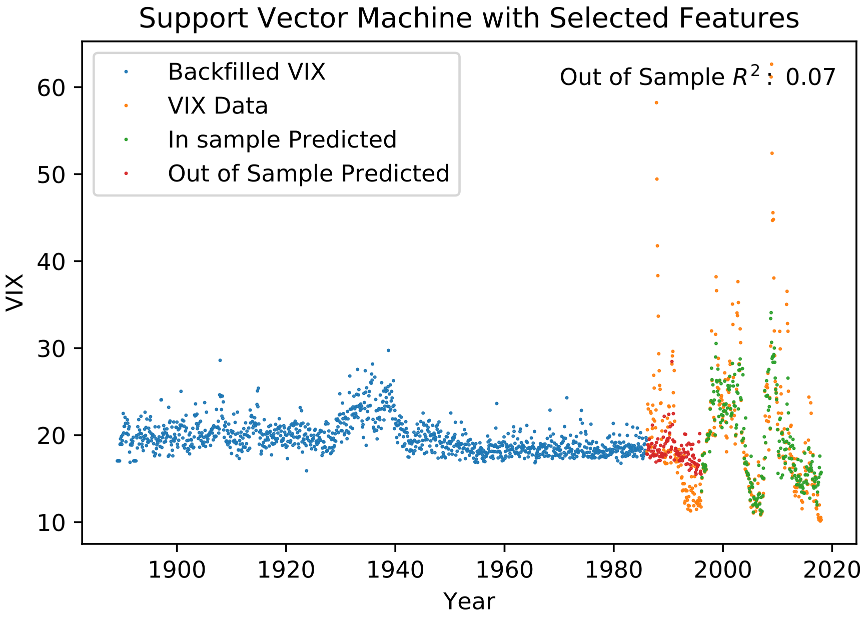

Figure 9: Backfilling VIX Index

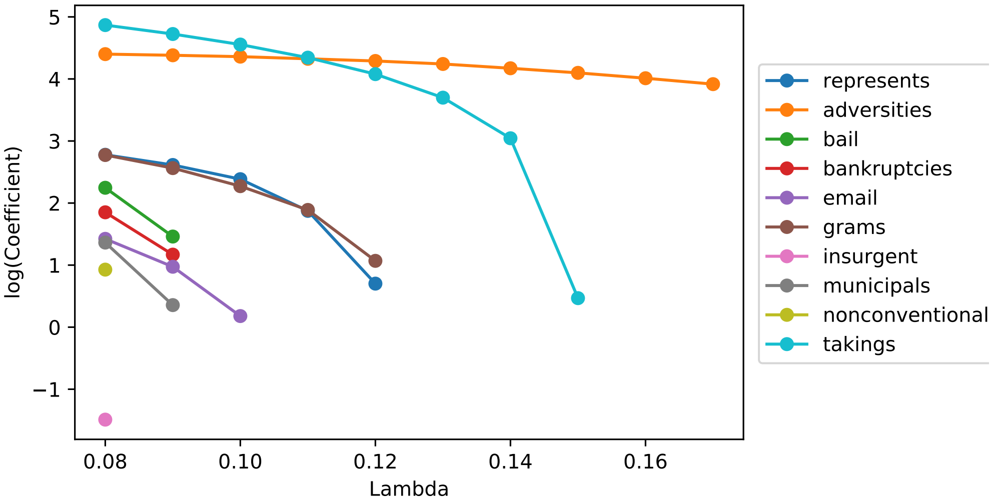

Figure 10: Backfilling VIX Index

cycles. In Bordalo, Gennaioli, and Shleifer (2018), expectation error play a critical role in generating boom and bust patterns. In their model, expectations about future credit defaults are overly influenced by current news, and investors optimism is exhibited in credit spread. Excessively narrow credit spread will lead to expansions of credit, and real activity will pick up. Importantly, all of these patterns will reverse when future states turn to be disappointing.

Can expectation errors predict future macroeconomic outcomes? To answer this question, we need a long time series of expectation data. However, forecast data of credit spread was available only for recent years. To conquer this problem, we backfill expectation error data by applying textual analysis on various the Wall Street Journal. We use Blue Chip Financial Forecast data from 1999 to 2017 as our training sample and use text data to backfill expectation from 1929 to 1998.

To backfill expectation error, we first estimated the following model:

$$error_t = \alpha + \gamma x_t^T + \eta_t \tag{7}$$

where $error_t$ is the difference between the expectation and the realized Baa corporate bond spread. The expectation of Baa corporate bond spread is defined as consensus forecast of Baa corporate bond yield minus consensus forecast of 10-year Treasury yield. Both are

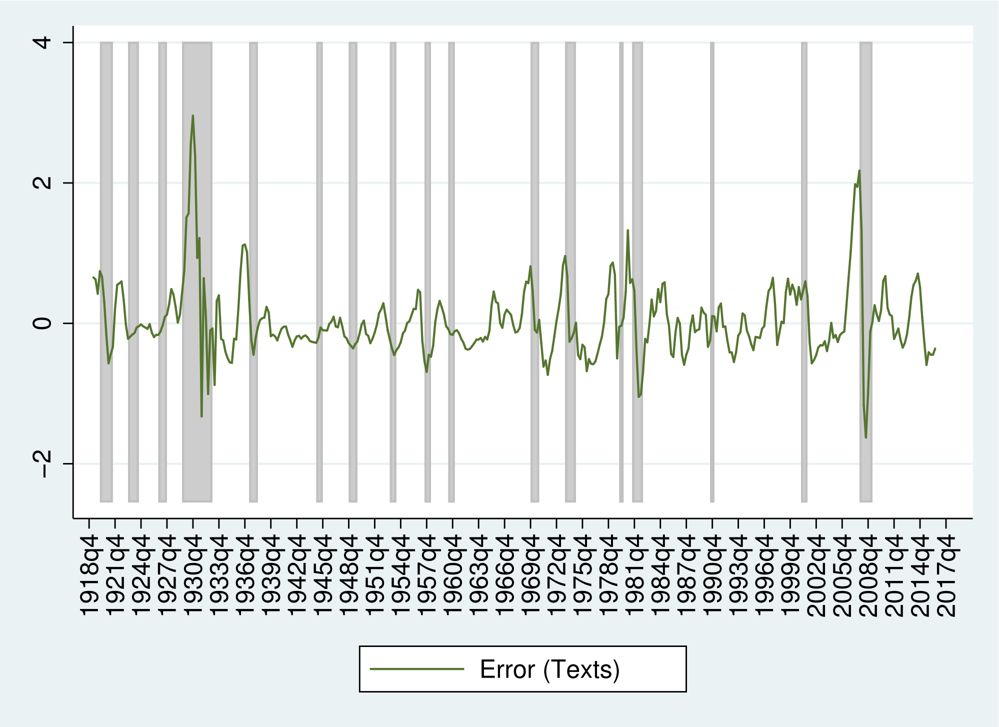

Figure 11: Backfilling Expectation Error

one-year forecasts collected from Blue Chip Financial Forecasts. The realized Baa corporate bond Spread is calculated in the same way using historical value.

To further manage model dimensionality, I apply LASSO penalization to estimate (7). We find that discussions about government (e.g taxes, president, white house, and Washington), finance (money, banks, treasury, credit, and stock), recessions (e.g great depressions, great recessions, crisis, and economic downturns), war (e.g military, world war, and Iraq) are the most useful in constructing expectation error. Using estimated γ and topic loadings, we backcast expectation errors for a long horizon, shown in figure 11

A clear pattern emerges from figure 11: error tends to be positive (overly optimistic) at the end of of booms and negative (overly pessimistic) during recessions. The countercyclical nature suggests that expectation error may predict business cycles. We explore this pattern more carefully in the following predictive regression framework:

Table 4:Predictive Regressions: Real GDP Growth

	(1)	(2)	(3)	(4)	(5)	(6)
	h=0	h=1	h=2	h=0	h=1	h=2
Error_t	-0.072	-0.334***	-0.179	-0.071	-0.328***	-0.174
	(0.008)	(0.010)	(0.010)	(0.008)	(0.010)	(0.009)
Change in Credit	-0.412***	-0.236**	-0.097	-0.423***	-0.293***	-0.169
Spread_t	(0.014)	(0.017)	(0.016)	(0.014)	(0.017)	(0.015)
Credit Spread_t				0.056	0.292**	0.368***
				(0.004)	(0.005)	(0.005)
Obsersvations	86	86	85	86	86	85
R^2	0.506	0.196	0.179	0.508	0.257	0.276

$control_{j,t}$ also include change in GDP, and other significant variables documented in literature such as CPI inflation rate and changes in short-term and long-term Treasury yields. ***,**,* Coefficient statistically different than zero at the 1%, 5% and 10% confidence level, respectively.

$$\Delta y_{t+h} = \beta_0 + \beta_1 \widehat{error}_t + \sum \beta_j controls_{j,t} + \epsilon_{t+h}$$

Δy_{t+h} is the log-difference of real GDP per capita over the course of year t + h. $\widehat{error}_t$ is the backfilled expectation error averaged over year t-1 to year t. $controls_{j,t}$ include change in credit spreads over year t, change in GDP per capita from year t-1 to t, CPI inflation rate, and changes in short-term and long-term Treasury yields. As a robustness check, we also include several lags of the control variables to ensure that mean-reversion in GDP growth is not responsible for the results.

Table 4 presents various specification of the predictive regression for different horizon. The explanatory variable of interest in this table is $\widehat{error}_t$. From column 1 to 3, we vary one-year output growth on the left-hand side from being contemporaneous to two years into the future. As can be seen from column 2, expectation error at t have substantial forecasting power for GDP growth in year t+1 and t+2, even after controlling for changes in credit spread: a one standard deviation increase in expectation error is associated with a step-down in real GDP growth per capita of 0.45-0.5 standard deviations, or about 1.2 percentage points. In column 4 to 6, I add levels of credit spread as an additional control. The results remain largely unchanged. Neither changes nor levels of credit spread are predictive of real GDP growth in year t+1 or t+2. Instead, expectation error is a strong predictor of future GDP growth.

Word Tones and Content Analysis

Jegadeesh and Wu (2013) (*Journal of Financial Economics*) present a new approach to quantify document tone. Compared to previous methods, the strength of each word in conveying positive and negative tones is determined objectively from markets reaction instead of assuming equal. Moreover, while this approach initially is based on a lexicon with categories (positive and negative), it exhibits independence of the subjectively predetermined classification of words. This method also reveals significant relation between the tone of 10-Ks and stock returns during the filing period, and it can be applied in other economics context. For example, the tones of IPO prospectuses computed in the same way has a negative relation with IPO underpricing.

The authors use OLS to train the coefficients on the tone scores they define, and show that their model has different implications from the inverse document frequency (idf) employed by previous literature for content analysis. While least frequent words are considered most impactful in idf, they can be both most and least impactful with WP weights here. The authors then use their model to predict stock returns and IPO underpricing and find their model performs better.

One can use textual factors to predict stock returns and IPO underpricing, and compare our results with those in the paper.

6 Conclusion

Modern institutions leverage big/alternative/unstructured data, in particular texts, for originating loans, predicting asset returns, improving customer service, etc. Moreover, interpretable textual information sheds light on key economic mechanisms and explanatory variables. We therefore develop a general framework for analyzing large-scale text-based data, which captures complex linguistic structures while ensuring computational scalability and economic interpretability. We then demonstrate potential applications of our methodology to issues in finance and economics, such as forecasting asset returns or macroeconomic outcomes, valuing startups, and interpreting existing models. By combining the strengths of neural network language models, especially vector representation, and generative statistical modeling, our data-driven approach leverages high-performance computation and strikes the balance between model complexity and interpretability.

References

Anandkumar, Anima, Dean P Foster, Daniel J Hsu, Sham M Kakade, and Yi-Kai Liu, 2012, A spectral algorithm for latent dirichlet allocation, in *Advances in Neural Information Processing Systems* pp. 917–925.

Andoni, Alexandr, Piotr Indyk, Thijs Laarhoven, Ilya Razenshteyn, and Ludwig Schmidt, 2015, Practical and optimal lsh for angular distance, in *Advances in Neural Information Processing Systems* pp. 1225–1233.

Antweiler, Werner, and Murray Z Frank, 2004, Is all that talk just noise? the information content of internet stock message boards, *The Journal of finance* 59, 1259–1294.

Arora, Sanjeev, Rong Ge, Yonatan Halpern, David Mimno, Ankur Moitra, David Sontag, Yichen Wu, and Michael Zhu, 2013, A practical algorithm for topic modeling with provable guarantees, in *International Conference on Machine Learning* pp. 280–288.

Baker, Scott R, Nicholas Bloom, and Steven J Davis, 2016, Measuring economic policy uncertainty, *The Quarterly Journal of Economics* 131, 1593–1636.

Bellstam, Gustaf, Sanjai Bhagat, and J Anthony Cookson, 2016, A text-based analysis of corporate innovation, .

Bengio, Yoshua, Réjean Ducharme, Pascal Vincent, and Christian Jauvin, 2003, A neural probabilistic language model, *Journal of machine learning research* 3, 1137–1155.

Blei, David M, Andrew Y Ng, and Michael I Jordan, 2003, Latent dirichlet allocation, *Journal of machine Learning research* 3, 993–1022.

Bollen, Johan, Huina Mao, and Xiaojun Zeng, 2011, Twitter mood predicts the stock market, *Journal of computational science* 2, 1–8.

Bordalo, Pedro, Nicola Gennaioli, and Andrei Shleifer, 2018, Diagnostic expectations and credit cycles, *The Journal of Finance* 73, 199–227.

Cohen, Lauren, Christopher J Malloy, and Quoc H Nguyen, 2016, Lazy prices, .

Datar, Mayur, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni, 2004, Locality-sensitive hashing scheme based on p-stable distributions, in *Proceedings of the twentieth annual symposium on Computational geometry* pp. 253–262. ACM.

Engelberg, Joseph E, and Christopher A Parsons, 2011, The causal impact of media in financial markets, *The Journal of Finance* 66, 67–97.

Evans, James A, and Pedro Aceves, 2016, Machine translation: mining text for social theory, *Annual Review of Sociology* 42.

Garcia, Diego, 2013, Sentiment during recessions, *The Journal of Finance* 68, 1267–1300.

Gentzkow, Matthew, Bryan T Kelly, and Matt Taddy, 2017, Text as data, Discussion paper, National Bureau of Economic Research.

Gentzkow, Matthew, and Jesse M Shapiro, 2010, What drives media slant? evidence from us daily newspapers, *Econometrica* 78, 35–71.

Gompers, Paul, William Gornall, Steven N Kaplan, and Ilya A Strebulaev, 2016, How do venture capitalists make decisions?, Discussion paper, National Bureau of Economic Research.

Gornall, William, and Ilya A Strebulaev, 2017, Squaring venture capital valuations with reality, Discussion paper, National Bureau of Economic Research.

Grimmer, Justin, and Brandon M Stewart, 2013, Text as data: The promise and pitfalls of automatic content analysis methods for political texts, *Political analysis* 21, 267–297.

Hanley, Kathleen Weiss, and Gerard Hoberg, 2010, The information content of ipo prospectuses, *The Review of Financial Studies* 23, 2821–2864.

Hansen, Stephen, Michael McMahon, and Andrea Prat, 2017, Transparency and deliberation within the fomc: a computational linguistics approach, *The Quarterly Journal of Economics* 133, 801–870.

Heckman, James J, 1979, Sample selection bias as a specification error, *Econometrica* 47, 153–161.

Hoberg, Gerard, and Vojislav Maksimovic, 2014, Redefining financial constraints: A text-based analysis, *The Review of Financial Studies* 28, 1312–1352.

Hoberg, Gerard, and Gordon Phillips, 2010, Product market synergies and competition in mergers and acquisitions: A text-based analysis, *The Review of Financial Studies* 23, 3773–3811.

Hoffman, Matthew, Francis R Bach, and David M Blei, 2010, Online learning for latent dirichlet allocation, in *advances in neural information processing systems* pp. 856–864.

Hofmann, Thomas, 1999, Probabilistic latent semantic analysis, in *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence* pp. 289–296. Morgan Kaufmann Publishers Inc.

Jegadeesh, Narasimhan, and Di Wu, 2013, Word power: A new approach for content analysis, *Journal of Financial Economics* 110, 712–729.

Jegadeesh, Narasimhan, and Di Andrew Wu, 2017, Deciphering fedspeak: The information content of fomc meetings, .

Kelly, Brian, Asaf Manela, and Alan Moreira, 2018, Text selection, *Working Paper*.

Le, Quoc, and Tomas Mikolov, 2014, Distributed representations of sentences and documents, in *International Conference on Machine Learning* pp. 1188–1196.

Leskovec, Jure, Anand Rajaraman, and Jeffrey David Ullman, 2014, *Mining of massive datasets* (Cambridge university press).

Loughran, Tim, and Bill McDonald, 2011, When is a liability not a liability? textual analysis, dictionaries, and 10-ks, *The Journal of Finance* 66, 35–65.

———, 2013, Ipo first-day returns, offer price revisions, volatility, and form s-1 language, *Journal of Financial Economics* 109, 307–326.

Lucca, David O, and Francesco Trebbi, 2009, Measuring central bank communication: an automated approach with application to fomc statements, Discussion paper, National Bureau of Economic Research.

Manela, Asaf, and Alan Moreira, 2017, News implied volatility and disaster concerns, *Journal of Financial Economics* 123, 137–162.

Mikolov, Tomas, Kai Chen, Greg Corrado, and Jeffrey Dean, 2013, Efficient estimation of word representations in vector space, *arXiv preprint arXiv:1301.3781.*

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* pp. 3111–3119.

Pennington, Jeffrey, Richard Socher, and Christopher Manning, 2014, Glove: Global vectors for word representation, in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* pp. 1532–1543.

Sontag, David, and Dan Roy, 2011, Complexity of inference in latent dirichlet allocation, in *Advances in neural information processing systems* pp. 1008–1016.

Taddy, Matt, 2015, Document classification by inversion of distributed language representations, *arXiv preprint arXiv:1504.07295.*

Tetlock, Paul C, 2007, Giving content to investor sentiment: The role of media in the stock market, *The Journal of finance* 62, 1139–1168.

Wallach, Hanna M, David M Mimno, and Andrew McCallum, 2009, Rethinking lda: Why priors matter, in *Advances in neural information processing systems* pp. 1973–1981.