Marketing Science Institute Working Paper Series 2021

Report No. 21-115

Focused Concept Miner (FCM): Interpretable Deep Learning for Text Exploration

Dokyun Lee, Emaad Ahmed Manzoor, and Zhaoqi Cheng

Focused Concept Miner (FCM): Interpretable Deep Learning for Text Exploration

Dokyun Lee, Emaad Ahmed Manzoor, Zhaoqi Cheng
{Dokyun, Emaad, Zhaoqi}@cmu.edu
Carnegie Mellon University*

Previous Versions May 2018, Oct 2018, Oct 2019, Dec 2019, May 2020.
This Version Sept 2020.

Abstract

We introduce the Focused Concept Miner (FCM), an interpretable deep learning text mining algorithm to (1) automatically extract coherent corpus-level *concepts* from text data, (2) *focus* the discovery of concepts so that they are highly correlated to the user-specified outcome, and (3) *quantify* the concept correlational importance to outcome. FCM is used to explore and potentially extract *apriori unknown* concepts from text that may explain business outcome. FCM is a custom neural network model explicitly configured to increase corpus-level insights and recovered-concept diversity without the need to provide any training data.

We evaluate FCM using a dataset of online purchases containing the reviews read by each consumer. Compared to 4 interpretable baselines, FCM attains *higher interpretability* as quantified by 2 human-judged metrics and 1 automated metric, and *higher recall* of unique concepts as supported by several experiments. In addition, FCM extracted constructs relating to product quality theorized to impact conversion in literature, without being explicitly trained to do so. FCM also achieves superior predictive performance compared to 4 interpretable benchmarks while maintaining superior or competitive predictive performance compared to 8 blackbox classifiers. In further experiments, we evaluate FCM on text data from online newsgroups and a crowdfunding platform, investigate the impact of *focusing* on concept discovery, and study the interpretability-accuracy trade-off. We present FCM as a complimentary technique to explore and understand text data before applying standard causal inference techniques. We conclude by discussing managerial implications, potential business applications, limitations, and ideas for future development.

Keywords: Interpretable Machine Learning, Deep Learning, Text Mining, Concept Extraction, Transparent Algorithm, XAI, Augmented Hypothesis Development.

*Equal contributions by all authors. We thank Olivier Toubia, Oded Netzer, Jey Han Lau, David Blei, Arun Rai, Gedas Adomavicius, Sudhir K., Carl Mela, Christophe Van Den Bulte, Raghu Iyengar, Eric Bradlow, Ryan Dew, Alex Burnap, Mingfeng Lin, Panos Ipeirotis, DJ Wu, Kunpeng Zhang, Daehwan Ahn, Alan Montgomery, Lan Luo, Dinesh Puranam, George Chen, Lizhen Xu, John McCoy, Eric Schwartz, Fred Feinberg, Anocha Aribarg, Puneet Manchanda, Elie Ofek, John A. Deighton, Doug Chung, and participants of Marketing Science Conf 2018 and 2019, CIST 2018, Conf on Digital Marketing and ML 2018, Advanced Computing in Social Sciences Symp 2019, Choice Symp 2019, INFORMS 2019, Conf on AI, ML, and Digital Analytics 2019, KRAIS ICIS 2019, Wharton Behavioral Insights through Text 2020, seminar audiences at McGill University, Korea Advanced Institute of Science and Technology, Seoul National University, University of Pittsburgh, University of Southern California, University of Minnesota, HEC Paris, University of Maryland, Georgia Institute of Technology, Harvard University, University of Michigan, The Wharton School, and Rutgers University for very helpful comments or conversations that shaped the paper. We gladly acknowledge generous grants by Nvidia for research, Marketing Science Institute Grant # 4000562. We thank Mao Chengfeng for command-line interface and Eric Zhou for demos. *www.fcminer.com* provides command line interface tool and python notebook demos.

1 Introduction

It is imperative for businesses to efficiently process and understand text data, given that more than 90% of data is estimated to be unstructured (Gantz and Reinsel, 2011), 68% of which is consumer generated (Mindtree, 2017). Content creators like Netflix and Amazon are collecting feedback reviews to create new shows tailored for success (Wernicke, 2015). Companies like C&A Marketing have teams of people who read through reviews on Amazon and eBay to identify consumer needs and use them to create new products (Feifer, 2013). After the decades-long surge of unstructured data, retailers and researchers are getting better at utilizing text data to obtain actionable insights such as measuring impact of user-generated content on business and extracting valuable information about customers and market (Abbasi et al., 2019; Yang et al., 2019; Gao et al., 2018; Netzer et al., 2019; Ananthakrishnan et al., 2020; Lysyakov et al., 2020).

Despite the deluge of potentially insightful data, studies show that nearly 80% of enterprises lack sufficient capabilities to manage unstructured data (Rizkallah, 2017), and $3 trillion in value goes uncaptured globally in text data alone (McKinsey, 2016). We believe this is due in part to the lack of scalable text exploration methodologies that (1) *automatically* extract corpus-level interpretable concepts[1] (unambiguous coherent construct), (2) *focus* the discovery of concepts so they are highly correlated to the user-specified outcome, (3) provide some degree of importance for mined concepts, (4) all without the need for costly training dataset or predefined constructs to mine. To explore and potentially extract *apriori unknown* concepts from text that may explain business outcome, all four features are desired.

This paper introduces the Focused Concept Miner (FCM), a novel deep learning-based text mining algorithm that (1) extracts highly interpretable concepts, as quantified by both human-judged and automated metrics, (2) is customized to filter out concepts uncorrelated to user-specified outcome while promoting concept diversity, (3) provides *correlational* measure of concepts to outcome, (4) without the user providing any training data or apriori defined content to mine. We outline the key features of FCM in Algorithm 1 and illustrate them in Figure 1. By applying FCM, managers and researchers can quickly make sense of and extract insights from a large amount of textual data tied to a business outcome for augmented hypothesis development and pattern recognition *before* launching a more involved causal study.

Our methodological contributions are twofold. FCM (1) extracts concepts that are more interpretable (coherent and unambiguous), and (2) extracts more unique concepts (higher recall) relevant to an outcome of business importance. We discuss existing approaches to achieve the partial output described in Algorithm 1 in Section 2, as well as the novel aspects of FCM. Figure 2 presents a flow chart for concept mining and when to use FCM over other methods.

[1] To be defined in Section 3.1

Algorithm 1 Focused Concept Miner: Algorithm Overview

Input	(1) *Optional* Structured X (e.g., numerical, categorial) (2) Textual X (corpus) (3) *Optional* Y of business importance (numerical, categorical) linked to corpus and X
Output	(1) Focus-mined corpus-level concepts predictive of Y (2) Correlational relative importance of concepts against one another and structured X (3) predictive model (optionally useful)
Features	(1) Improved interpretability of mined concepts (2) Potential new *apriori unknown* concept findings (3) Focused concepts relevant to Y (4) Joint estimation of structured X and text that is end-to-end (in one pipeline) (5) No need for pre-defined human-tagged training data (6) Inductive inference for new unseen data (7) Online learning via mini-batch gradient descent

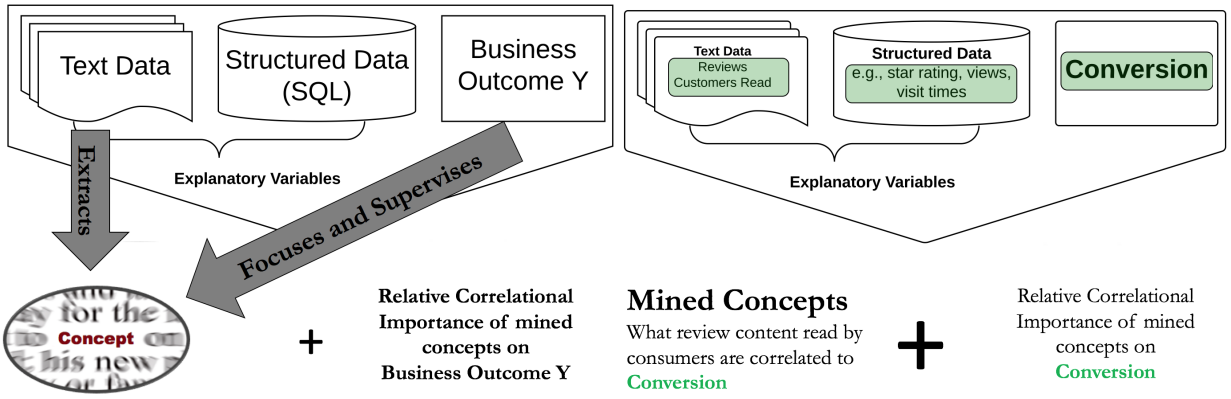


Figure 1: FCM Features Visualized with Use-case Example.

We evaluate FCM's performance on a dataset that tracks individual-level review-reading, searching, and purchasing behaviors on an e-commerce site. By applying FCM to this dataset, we demonstrate that FCM automatically extracts concepts from consumer-read product reviews that are known to influence product purchasing behavior—namely, dimensions of product price and quality—thereby providing an instance of external validity. Furthermore, compared to 4 interpretable baselines that partially achieve FCM's goals, FCM attains higher interpretability, as measured by three metrics (e.g., automated, 2 human-judged), and higher recall of unique concepts, as supported by several experiments. While it is not the main purpose of the algorithm, FCM achieves superior predictive performance compared to all interpretable benchmarks while maintaining superior or competitive performance even when compared to 8 prediction-focused blackbox algorithms.

To further demonstrate FCM's capabilities, we apply FCM to a crowdfunding dataset in Section 5.5.2, and an online newsgroup dataset in Appendix H. Additional experiments investigate the accuracy-interpretability relationship in FCM (Section 5.5.1) and the impact of focusing on extracted concepts (Section 5.5.3). Our contribution is thus a new method for text exploration with a focus on business use-cases.

FCM excels in extracting more interpretable concepts due to the following conceptual reasons:

1. **Focused Concept:** Concept mining is guided by Y of business importance to extract Y-relevant concepts. This seems counterintuitive, since it seems to add an additional constraint. However, text is high dimensional and providing Y effectively reduces hypotheses space. Focusing by Y refines the task more accurately for the algorithm.

2. **Semantic Similarity Knowledge**: FCM uses word representation that learns semantic relationships. It also uses both local and global contextual information to focus-mine concepts.

3. **Concept Diversity & Document Sparsity**: The model forces discovered concepts to be distinct from each other (diversity) and forces the document to be pithy (sparsity).

4. **End-to-End**: Focus-mined concepts and structured X are jointly estimated to predict Y in an end-to-end (one pipeline optimization) fashion. The model shares information from the beginning to the end and thus is more efficient.

FCM performs in one optimization step what may have taken managers and researchers many steps worth of text and data mining tasks, often filled with ad-hoc feature engineering and with methods ill-suited for extracting coherent and interpretable concepts. FCM demonstrates that deep learning approaches, normally associated with a lack of interpretability and often considered to be blackboxes, could be utilized to help businesses better understand textual data.

We end this introduction with a caveat. FCM is an exploratory and non-causal technique that is correlational in nature. FCM could serve as a useful tool to refine exploding volumes of

unstructured text data to recover Empirical Generalization[2] in management science, as discussed by Bass (1995)—in short, as a scalable text tool for pattern detecting, understanding, and precursor to theory building.

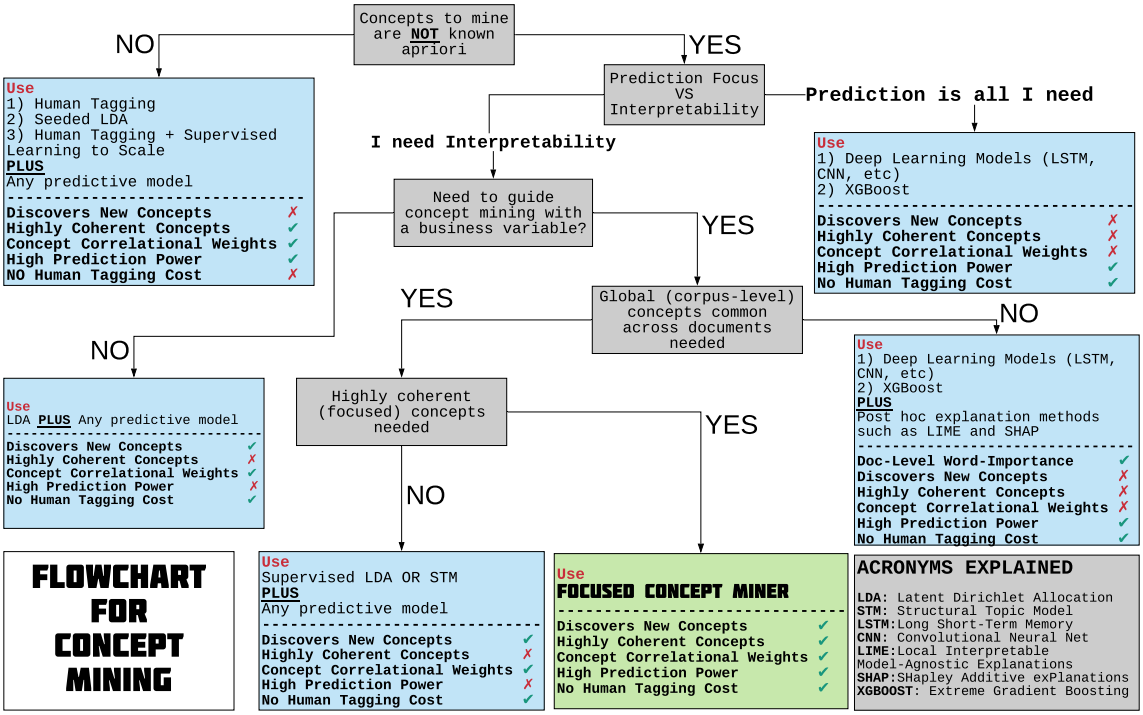


Figure 2: Flowchart for Concept Mining and When to Use FCM. See Section 2 for Literature.

2 Literature

Broadly, the task described in Algorithm 1 is applicable in many settings where text is linked to a business outcome. For clarity, take for example a consumer purchase (Y) related to consumer behavior on the web (X) and the consumer-read product reviews as text inputs. The manager may want to (1) predict conversion from user behavior, product attributes, and the reviews read, (2) investigate what X may predict conversion, and (3) delve deeper into the review text, to understand what content read by consumers may be highly correlated to conversion.

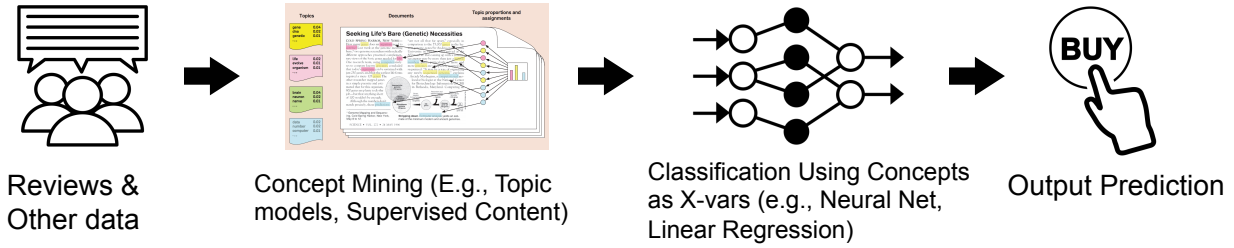
FCM can be used for all three tasks, but its focus is primarily on the third problem when concepts to mine are *unknown apriori*. Normally, deriving insights from text is tackled with a multi-step approach involving several different techniques. Procedures are different based on whether the concept is known apriori or not. Section 2.3 discusses a popular supervised learning framework as shown in Lee et al. (2018) and Liu et al. (2019) when the concept is *known apriori*.

When the concept is *unknown apriori*, figure 3 shows two approaches. Figure 3a is a multi-step approach. One would first apply unsupervised concept mining algorithms, such as Latent Dirichlet Allocation or aspect mining (discussed in Section 2.2). Mined concepts can then enter any classification or regression framework as X. This framework suffers from poor interpretability, low recall of unique concepts, lack of focus by the Y, and subpar prediction accuracy compared to

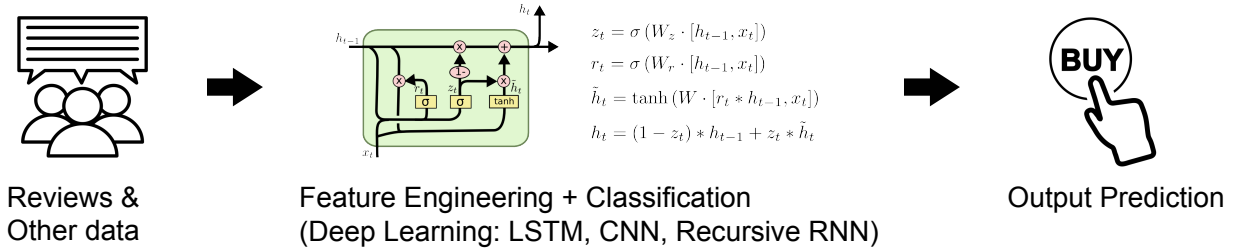
[2]Bass (1995) describe Empirical Generalization as "a pattern or regularity that repeats over different circumstances and that can be described simply by mathematical, graphic, or symbolic methods."

FCM, as Section 5 examines. Next, a manager that prioritizes high predictive performance may apply typical deep learning methods (Figure 3b), such as convolutional neural nets (CNN), which recover local-level feature patterns to aid prediction (for details, see Goldberg (2016)). However, this approach offers no insight from the text on why conversion might have happened. For further interpretable insights, post hoc algorithms like LIME (Ribeiro et al., 2016) or SHAP (Lundberg and Lee, 2017) must be applied. However, these approaches result in fragmented local-level concepts specific to individual data points (i.e., specific to a document and not corpus), unlike FCM—that is, users cannot get *high-level global concepts* (corpus level) from post hoc explanation method on deep learning (see Section 2.1).

Section 2.1 discusses what it means for an algorithm to be more "interpretable" by introducing the XAI (eXplainable Artificial Intelligence) literature. We discuss several definitions of interpretability and operationalize metrics to compare FCM against existing techniques. Next, we discuss two relevant literatures that provide competing techniques from computer science (pertaining to extracting content and concepts from text) and similar applications from business. We point out the conceptual differences of FCM.



(a) **Multi-step Framework to Predict and Extract Insight from Text:** A manager wishing to automatically extract concepts from text may utilize this multi-step machine learning framework.



(b) **Deep Learning Framework to Predict Conversion:** For high predictive performance, one may apply deep learning methods. However, this offers no insight from the text on why conversion might have happened. For further interpretable insights, post hoc processing to open up the blackbox must be applied (e.g., LIME, SHAP). However, this approach results in fragmented concepts, unlike FCM. LSTM graphic taken from https://colah.github.io/

Figure 3: **Conceptual Frameworks to Predict Outcome and Derive Insights from Text.**

2.1 Interpretable Machine Learning (ML)

The success of high-performing blackbox algorithms such as deep neural networks (LeCun et al., 2015) and boosted trees (Chen and Guestrin, 2016) is well documented.[3] However, these algorithms do not give any rationale for their predictions. This issue is amplified when deployed on

[3]Top-performing algorithms in data science competitions are usually one of the two mentioned, according to Kaggle.com co-founder Anthony Goldbloom. https://www.linkedin.com/pulse/lessons-from-2mm-machine-learning-models-kagglecom-data-harasymiv/

business intelligence systems that deal with consumer data, such as in automated credit approval and investments where auditability, liability, privacy, and other high-stake issues are entangled. Consequently, a global survey of more than 3,000 executives and managers show that "Developing Intuitive Understanding of AI" as number one challenge (Ransbotham et al., 2017). In fact, 2018 GDPR (General Data Protection Regulation) require algorithmic transparency among firms while DARPA also announced $2 billion initiatives for eXplainable Artificial Intelligence (XAI).

In most non-trivial settings, understanding why algorithms made certain predictions is critical to prevent egregious failures, justify usage, improve efficiency, and to use for decision making. Blackbox failures have been well documented. Angwin et al. (2016) and Wexler (2017) report cases of algorithmic racial bias in bailout decisions (stemming from biased training data obfuscated by the opaqueness of the algorithm), and even instances where the algorithms incorrectly denied parole. Zech et al. (2018) report training a deep vision net in the context of medical disease prediction based on x-rays. The system keyed on the meta-tagged word "portable"—reflective of where the samples came from—instead of a valid signal for disease. Additional consequences continue to pile up as blackbox algorithms are utilized without interpretability.

In response to the need for interpretability in machine learning algorithms, several sub-streams of research have arisen since the mid-2010s (Please see Guidotti et al. (2018); Gilpin et al. (2018) for surveys). The stream most related to our work is the XAI literature, which broadly defines (Rudin, 2019) two different algorithm families for interpretability.

Definition 1 (Explainable Machine Learning): Given a blackbox predictor B and a training dataset $D = \{X, Y\}$, the explainable machine learning algorithm takes as an input a blackbox B and a dataset D, and returns a transparent predictor T with requirements that 1) T replicates the prediction of blackbox predictor B with high fidelity, and 2) T offers human-understandable rationale for each prediction either at the instance-level or the model-average level. T may be a shallow tree, small set of rules, or linear regression with not too many explanatory variables.

Definition 2 (Interpretable Machine Learning): Interpretable machine learning algorithms refer to inherently transparent algorithms that provide human-understandable rationale for predictions yet still offer competitive performances compared to prediction-focused blackbox algorithms.

In this framework, FCM falls under the category of interpretable machine learning algorithm.

While the XAI literature has grown significantly and will continue to do so, the definition of "interpretability" still remains an illusive, fragmented, and domain-specific notion (Rudin, 2019; Lu et al., 2020) left to the researcher and user to define. Lipton (2016) states that "both the motives for interpretability and the technical descriptions of interpretable models are diverse and occasionally discordant, suggesting that interpretability refers to more than one concept." A recent survey of XAI, Guidotti et al. (2018), concludes by stating that "one of the most important open problems is that, until now, there is no agreement on what an explanation is." There have been several attempts to define this. To briefly paraphrase a few sampled works, Doshi-Velez and Kim (2017) state "interpretability is the degree to which a human can consistently predict the model's result," Miller (2018) state "interpretability is the degree to which a human can understand the cause of a decision," and Dhurandhar et al. (2017) state "AI is interpretable to the extent that the produced interpretation I is able to maximize a user's target performance." Few papers also tackle desiderata for interpretability conceptually, such as unambiguity (input and outputs are clear), selectiveness (a parsimonious explanation), contrastiveness (a "had input been x, output would have been y" type of explanation), factative (has to be highly truthful), etc (Lipton, 2016; Doshi-Velez and Kim, 2017; Miller, 2018; Lu et al., 2020).

In this paper, we incorporate insights from XAI literature, as well as interpretability in topic modeling literature, to propose to measure the "interpretability" by (1) tapping into existing interpretability measurements in topic modeling called *coherence*, (2) confirming the coherence measure

with human-judgment directly from mechanical turk, and (3) operationalizing the definition by Dhurandhar et al. (2017) into our problem-specific metric and measuring it directly with human subjects from mechanical turk. We discuss the measures in Section 3.5.

2.2 Finding Concepts in Text: Topic Modeling & Others

We discuss methods that *partially* achieve the goal outlined in Algorithm 1.

Within machine learning (ML) literature, one primary concern of natural language processing (NLP) is to extract meaningful concepts from a given text. NLP literature offers several ways to achieve the outcome described in Algorithm 1. An initial approach might be to apply any supervised machine learning algorithm to Input (Algorithm 1) to treat individual words or n-grams (n contiguous words) as X to predict the Y. Using any combination of feature selection methods (Chandrashekar and Sahin, 2014), one could extract several keywords or n-grams that could potentially explain the business outcome. However, these methods usually provide a fragmented list of words without enough coherence to be effective in extracting hypothesis-worthy concepts. Once they have a list of salient and informative words from these analyses, managers must drill down further to manually compose coherent and unambiguous concepts, which is a subjective rather than objective procedure. After this, it is still unclear how a manager may be able to gauge the economic impact of composed concepts that can consist of several distinct keywords.

Aspect-based sentiment analysis may partially achieve the goals of Algorithm 1. This analysis is concerned with mining opinions from text about 1) specific aspects of focal products or subjects and 2) sentiment valence of these aspects (Liu, 2012). Specifically, an aspect extraction sub-task can be used to reveal key concepts of the text (Pontiki et al., 2016). Briefly summarized, aspect extraction in product opinion mining utilizes a series of smaller techniques and heuristics to identify concepts that describe product aspects. For example, most algorithms first identify adjectives in text using part-of-speech tagging and then conduct additional filtering based on informativeness. This set of techniques lacks the features to achieve the goal of Algorithm 1 because these algorithms 1) discover narrow and simple aspects; 2) cannot be focused by Y; 3) require domain knowledge to feature engineer, which defeats the exploratory purpose of identifying unknown concepts; 4) focus on product reviews, which makes it unclear how they extend to non-review texts. FCM goes beyond product reviews and can be applied to any text data.

The most relevant and influential set of algorithms that focus on automatically extracting concepts from text data are topic models. Latent Dirichlet Allocation (LDA) (Blei et al., 2003) is a probabilistic generative model that seeks to identify latent topics that comprise a collection of documents (corpus). Briefly described, LDA assumes that a topic is a probabilistic distribution over a finite set of vocabularies and that a document consists of several different topics. Then, the method estimates a hierarchical Bayesian architecture with an expectation-maximization algorithm to converge on document-topic distribution and topic-word distribution. The end result is that a user gets a probability vector of topics for each document (document-topic vector) and a bag-of-words with loadings to describe topics (topic-word distribution).

While LDA is a seminal work, it cannot achieve the goal outlined in Algorithm 1 because it is an unsupervised algorithm. Given Y, a user cannot guide the algorithm to discover certain topics or provide a variable for algorithms to adjust the topics. Seeded LDA by Jagarlamudi et al. (2012) extends the original LDA to enable users to guide the topics based on user-input topic-words (thus cannot be used to mine apriori unknown concepts) while supervised versions of LDA (Blei and Mcauliffe, 2008; Zhu et al., 2012) modified the original model to guide topics discovered with Y. Lastly, the Structural Topic Model (STM (Roberts et al., 2014)) can incorporate both X and Y. The shortcomings of these LDA models are: (1) They struggle with extracting interpretable topics,

as discussed by Chang et al. (2009), i.e., topics discovered often suffer from *diffusion* (concepts appear in multiple topics) and *intrusion* (one topic contains many concepts and is ambiguous); (2) They do not jointly optimize prediction with X and discovered topics; and (3) They cannot provide the relative importance of topics without using an additional algorithm. In other literature streams, several papers tackle how to increase the semantic-coherence of topic models (see Mimno et al. (2011)). However, these models lack the ability to focus-mine topics from Y and relative importance metrics.

Finally, a handful of recent papers explore deep learning-based models that combine word embedding (Mikolov et al., 2013) and LDA to collaboratively improve the quality of latent topics and word embedding (Xun et al., 2017), to improve discovered topics and word embeddings via topic-specific word embeddings (Shi et al., 2017), and to jointly learn topic and word embeddings (Moody, 2016). However, these papers are again missing several features, such as (1) topic discovery guided by the Y, (2) the joint estimation of structured and unstructured data to predict the Y, and (3) inductive inference that enables prediction given new unseen data.

In summary, we are not aware of any methodologies that achieve the same output and features as our model.

2.3 Content Extraction Via NLP in Business

NLP has been used widely for business insights and applications. Some studies are dedicated to extracting and measuring brand perception, market trends, and adverse event from social media data (Netzer et al., 2012; Lee et al., 2018; Abbasi et al., 2019, 2018; Lysyakov et al., 2020), while many focus on extracting content and signals out of customer-generated product review data (Archak et al., 2011; Ransbotham et al., 2019; Liu et al., 2019; Chen et al., 2019; Choi et al., 2019).

Here, we mention papers concerned with methodology or the applications of *automatically* extracting concepts from textual business data. To the best of our knowledge, most papers in business research that seek to automatically extract concepts and topics from text data involve some variation of LDAs. For example, Buschken and Allenby (2016) extend the traditional LDA model to restrict to one topic per sentence and achieve better extraction of concepts from user-generated product reviews. Puranam et al. (2017) apply LDA in a restaurant review setting to see the impact of health regulation on what consumers talk about and on word-of-mouth. Liu and Toubia (2018) extend LDA and develop Hierarchically Dual Latent Dirichlet Allocation, then apply it in a search setting to infer consumers' content preferences based on their queries on the fly. Geva et al. (2019) apply LDA to examine how users in Twitter present themselves and shape their online presence.

The extant LDA approaches in the literature—as discussed in Section 2.2—have different goals or are missing certain features of FCM. More importantly, Section 5.1 and 5.2 show that FCM is (1) more interpretable and (2) recovers a higher number of relevant concepts compared to LDA families, resulting in better text understanding and greater managerial impact, as discussed in Section 6.

On the other hand, some papers report success with multi-stage supervised machine learning approaches in which key content is first defined and tagged by humans and then used to train NLP algorithms to scale to larger data. In this stream, Lee et al. (2018) utilize a traditional NLP approach to study what content (informative and brand-personality advertising) companies should post on social media to increase user-engagement (e.g., like, comments). Timoshenko and Hauser (2018) utilize a deep learning NLP approach to extractively reduce user review data and identify content (pre-defined and human-tagged on informativeness) related to consumer needs. Liu et al. (2019) utilize a deep learning NLP approach to investigate what content (dimensions of product price and quality) in user reviews influences consumer conversion. These papers require (1) apriori knowledge of what content to examine and (2) human-tagged labels on text data to answer particular business

questions. In contrast, FCM does not need training data and identifies concepts automatically. This is essential when managers want to discover unknown concepts in text highly correlated to Y for exploratory purposes.

3 Model

We formalize the notion of a *focused concept* in a neural network model and estimation details.

3.1 Definition of Focused Concept

We begin by defining focused concepts. Philosophically, many formulations of "concept" have been proposed, with a consensus that "concepts lack definitional structure" (Margolis and Laurence, 2019). In defining concepts, we adopt the view of *conceptual pluralism* (Margolis et al., 1999), where concepts have multiple types of structure. What FCM considers as "concepts" and extracts is merely one aspect of this pluralistic structure. To describe concepts discovered by FCM, we consider *the classical theory of concepts* (Carey, 2009), which proposes that a lexical concept $\mathcal{C}$ consists of simpler concepts that express necessary and sufficient conditions for falling under $\mathcal{C}$—i.e., a concept is described by a set of words. This is an abstraction on how FCM mathematically models concept (as a vector in semantic-similarity-preserving space), in which a lexical concept $\mathcal{C}$ can be characterized by nearby group of simpler concepts, which is more akin to *prototype theory of concepts* (Carey, 2009). In summary, given an outcome Y, FCM recovers prototypical concepts from the text that best predict Y, and we describe each concept with a set of words.

FCM optimizes the coherence of the extracted concepts by associating each concept with a collection of semantically similar keywords that describe the central idea of the concept; for example, a concept associated with the words "beautiful, large, lovely" is essentially one that embodies "aesthetics." Intuitively, individual words form the most basic concept—an atomic concept. Several words together form a more complex concept. The key idea is that a concept and a word, abstractly speaking, can live in the same vector space. This particular idea has been successfully utilized to quantify complex concepts such as gender and ethnic stereotypes (Garg et al., 2018) and cultural connotations (Kozlowski et al., 2018) directly from text data using a technique called Word2Vec, which comprises the first layer of FCM and will be elaborated on next. FCM builds on this proof-of-concept via a carefully constructed novel architecture, loss function, and regularization terms, to ensure that extracted concepts are both focused by the outcomes in the data and diverse in the sense that no two concepts have a significant overlap (i.e., the concepts are segregated and prototypical). In relation to the underlying mathematics of the model, concept is defined as:

Concept a vector in a semantic-similarity-preserving vector space. Similar vectors in this space have similar meaning in natural language. A concept can be described by a collection of words local to each other.

Focused-Concept a concept extracted to be highly correlated to Y.

Connection to "topics" in topic modeling is simple. Topics are defined as distribution over words and a topic consists of words that co-occur frequently in documents. A concept is a topic plus the additional constraint that all words describing this topic are semantically similar.

Next, we describe the model by introducing each of its components and their motivation, and finally tie them all together and discuss various forms of data that the model may be trained on. Our notation is summarized in Table 1.

Dimensions		Intermediate Elements	
D	Number of documents	d	Document index
T	Number of concepts	b_d	Bag-of-words document vector
V	Vocabulary size	w, c	Pivot, context word indices
E	Embedding size	v_w, v_c	Pivot, context word embeddings
k	Window size	$C_k(w)$	Set of context word indices
m	Number of negative samples	$N_m(w)$	Set of negative samples
Learned Parameters		**Loss Weights**	
$\mathbf{E_w}$	Word embedding matrix ($V \times E$)	λ	Dirichlet sparsity strengths
$\mathbf{E_t}$	Concept embedding matrix ($T \times E$)	η	Diversity regularizer strength
$\mathbf{W}$	Document-concept weights ($D \times T$)	ρ	Classification loss strength
θ	Concept-classification weights ($1 \times T$)		
CAN	Concept Allocator Network		

Table 1: Notation

3.2 Embedding Words, Documents, and Concepts

Embedding words. We begin by modeling the distribution of words in each document. We rely on the distributional hypothesis in linguistics (Sahlgren, 2008), which states that words used together in similar contexts tend to have similar meanings.[4] Recent models based on this hypothesis have demonstrated state-of-the-art performance on various natural language processing tasks (Mikolov et al., 2013; Pennington et al., 2014). We follow the example of Word2Vec (Mikolov et al., 2013), which encodes the distributional hypothesis as a neural network to find vector representations of words such that semantic-spatial relationships are preserved—that is, "similar" words lie nearby in the embedding space. For clarity, we adopt the model derivation and notation from Goldberg and Levy (2014).

To this end, we represent each word w by a real-valued vector v_w of length E, called its *embedding*. If we denote the vocabulary of all words in our corpus of documents by $\mathcal{V}$, the embeddings are stored as rows in a matrix $\mathbf{E_w}$ of dimension $|\mathcal{V}| \times E$. We denote by $\mathcal{C}_k(w)$ the set of *context* words around the *pivot* word w within a symmetric k window size. The *pivot* word w is the center word used to predict the surrounding *context* words $c \in \mathcal{C}_k(w)$. We define the likelihood of a corpus of documents in terms of their words and contexts. Given a corpus (a document d, or collection of documents), its likelihood is defined as:

$$\prod_{d \in \text{corpus}} \prod_{w \in \text{document d}} \prod_{c \in \mathcal{C}_k(w)} p(c|w; \mathbf{E_w}) \tag{1}$$

This is essentially a mathematical formulation of the distributional hypothesis in linguistics: the probability of a document is defined in terms of the conditional probability of each context word c given its corresponding pivot word w. Given a context word c and its pivot word w, we would like to capture the fact that words occurring in the same context often should have similar embeddings. Hence, we parameterize the conditional probability $p(c|w; \mathbf{E_w})$ as follows:

$$P(c|w; \mathbf{E_w}) \quad = \quad \frac{e^{v_c \cdot v_w}}{\sum_{c' \in \mathcal{V}} e^{v_{c'} \cdot v_w}} \tag{2}$$

[4] In the words of linguist John Rupert Firth: "You shall know a word by the company it keeps."

where v_c and v_w are the embeddings of c and w. Our goal is to learn the embeddings $\mathbf{E_w}$ that maximize the likelihood of the observed data in eq.(1).

However, computing the conditional probability term in eq. (2) involves a computationally expensive summation in the denominator over all possible words in the vocabulary, of which there may be hundreds of thousands. Hence, we approximate this objective via *skip-gram negative sampling* (Mikolov et al., 2013).

Let $\mathcal{D}$ be the set of all observed pivot-context word pairs in the corpus, and $\mathcal{D}'$ be the set of all pivot-context pairs that do not occur in the corpus. For a given a pivot-context word pair (w, c), let $P((w, c) \in \mathcal{D}|\mathbf{E_w})$ be the probability that this pair occurs in the corpus, and let $P((w, c) \notin \mathcal{D}|\mathbf{E_w}) = 1 - P((w, c) \in \mathcal{D}|\mathbf{E_w})$ be the probability that the pair does not occur in the training corpus. We would like to find $\mathbf{E_w}$ such that the likelihood of the observed pivot-context pairs is maximized, while the likelihood of the unobserved pivot-context pairs is minimized. This is captured by the following objective:

$$\max_{\mathbf{E_w}} \prod_{(w,c)\in\mathcal{D}} P((w, c) \in \mathcal{D}|\mathbf{E_w}) \prod_{(w,c)\in\mathcal{D}'} P((w, c) \notin \mathcal{D}|E_w) \tag{3}$$

We can parameterize the probability $P((w, c) \in \mathcal{D}|\mathbf{E_w})$ using the logistic-sigmoid function $\sigma(x) = (1 + \exp(-x))^{-1}$ that scales its argument to lie in $(0, 1)$:

$$P((w, c) \in \mathcal{D}|E_w) = \sigma(v_w \cdot v_c) = \frac{1}{1 + e^{-v_c \cdot v_w}} \tag{4}$$

Plugging eq. (4) into eq. (3) and taking logarithms leads to the following objective:

$$\begin{aligned}
&\max_{\mathbf{E_w}} \log\Big(\prod_{(w,c)\in\mathcal{D}} P((w, c) \in \mathcal{D}|E_w) \prod_{(w,c)\in\mathcal{D}'} P((w, c) \notin \mathcal{D}|\mathbf{E_w})\Big) \\
&= \max_{\mathbf{E_w}} \sum_{(w,c)\in\mathcal{D}} \log(P((w, c) \in \mathcal{D}|\mathbf{E_w})) + \sum_{(w,c)\in\mathcal{D}'} \log(P((w, c) \notin \mathcal{D}|\mathbf{E_w})) \\
&= \max_{\mathbf{E_w}} \sum_{(w,c)\in\mathcal{D}} \log\frac{1}{1 + e^{-v_c \cdot v_w}} + \sum_{(w,c)\in\mathcal{D}'} \log(1 - \frac{1}{1 + e^{-v_c \cdot v_w}}) \\
&= \max_{\mathbf{E_w}} \sum_{(w,c)\in\mathcal{D}} \log\frac{1}{1 + e^{-v_c \cdot v_w}} + \sum_{(w,c)\in\mathcal{D}'} \log\frac{1}{1 + e^{v_c \cdot v_w}} \\
&= \max_{\mathbf{E_w}} \sum_{(w,c)\in\mathcal{D}} \log(\sigma(v_c \cdot v_w)) + \sum_{(w,c)\in\mathcal{D}'} \log(\sigma(-v_c \cdot v_w))
\end{aligned} \tag{5}$$

The computationally expensive summation over all possible $(w, c) \in \mathcal{D}'$ in the second term can be approximated by summing over m *negatively-sampled* pivot-context pairs $\mathcal{D}_m$. The sampling is performed as follows for every $(w, c) \in \mathcal{D}$: sample $(w, c'_1), (w, c'_2), \ldots, (w, c'_m)$ such that $(w, c'_i) \notin \mathcal{D}$ and each c'_i is drawn with probability proportional to its frequency[5] in the corpus, and $P(c'_i) = n(c'_i)/N$ where $n(w)$ is the frequency of word w in the corpus and N is the number of words in the corpus.

[5]Note that Mikolov et al. (2013) use the frequency exponentiated to 3/4 which provided superior empirical performance.

Converting the maximization to a minimization problem yields the first component of the *loss function*[6] that our method seeks to minimize:

$$\mathcal{L}_{\text{neg}} = -\sum_{(w,c)\in\mathcal{D}} \log(\sigma(v_c \cdot v_w)) - \sum_{(w,c)\in\mathcal{D}'} \log(\sigma(-v_c \cdot v_w)) \tag{6}$$

where σ is the logistic sigmoid function $\sigma(x) = (1 + \exp(-x))^{-1}$ as defined earlier, m is the number of negative samples and k is the window-size. Taking a closer look at this loss function, we observe that the first summation operates over all pivot-context pairs in the corpus to ensure that words occurring often in the same context have similar embeddings. The second term operates over each pivot-context pair that does *not* occur in the corpus, to encourage them to have *dissimilar* embeddings.

Embedding documents and concepts. We now describe an extension of the model to capture concepts by combining ideas from Mikolov et al. (2013) and Moody (2016). We assume the existence of a fixed number of concepts T, and assume that the words of each document are drawn from a distribution over these T concepts. We store the unnormalized form of this distribution (the "concept weights" for each document) in matrix $\mathbf{W}$ of dimension $D \times T$, where D is the number of documents in the corpus. Each concept is represented by its own embedding of length E, stored in a matrix $\mathbf{E_t}$ of dimension $T \times E$; note that the concept embeddings lie in the same space as the word embeddings, which is crucial for each concept to be interpreted by a collection of keywords. Given the concept embeddings $\mathbf{E_t}$ and concept weights $\mathbf{W}$, the embedding of a document v_d can be derived as a weighted linear combination of its concept embeddings, in line with our earlier assumption. We first transform the document-concept weights $\mathbf{W}[d]$ to a probability distribution p_d, and then use this to weight each of the concept embeddings:

$$p_d[i] = \frac{e^{\mathbf{W}[d][i]}}{\sum_{j=1}^{T} e^{\mathbf{W}[d][j]}} \quad \forall i = 1, \ldots, T \tag{7}$$

$$v_d = \sum_{i=1}^{T} p_d[i] \times \mathbf{E_t}[i] = p_d \times \mathbf{E_t} \tag{8}$$

We now need a way to link concepts and words in order to jointly learn their embeddings in the same space, using the loss function given in eq. (6). We do this by linking words with concepts via the documents they occur in. Specifically, we define a "document-specific" word embedding v_{dw} that represents the word w as it appears in the context of document d. For example, the word "bank" in an article about finance could have a different meaning (and hence, a different embedding) than that in an article about a river. We define this document-specific word embedding as a *translation* of the original word embedding v_w by the document embedding v_d:

$$v_{dw} = v_d + v_w$$

While any general function $v_{dw} = f(v_d, v_w)$ could have been used to define a document-specific word embedding, simple translation ensures that the loss function remains differentiable and efficient to compute, which are necessary for efficient minimization. Multiplication did not make any difference. The skip-gram negative sampling loss in eq.(6) can now be modified as:

[6] An alternative method to obtain word embeddings is via factorization of the shifted pairwise mutual information (PMI) matrix Levy and Goldberg (2014). However, formulating the objective this way eliminates the flexibility to extend the model to incorporate several objectives such as concept supervision and diversity, to be discussed later in this section.

$$\mathcal{L}_{\text{neg}} = -\sum_{(w,c)\in\mathcal{D}} \log(\sigma(v_c \cdot v_{dw})) - \sum_{(w,c)\in\mathcal{D}'} \log(\sigma(-v_c \cdot v_{dw})) \tag{9}$$

where $d \in 1, \ldots, D$ is the index of the document containing w and c. Minimizing this loss now enables us to learn both v_w and v_d (and hence, $\mathbf{E_t}$) from the training corpus.

Predicting concepts for unseen documents. The concept weight matrix $\mathbf{W}$ defined thus far enables learning the concept weights for documents in the available corpus. In some scenarios, we foresee a trained FCM model being used to *predict* the concepts for unseen documents, which were previously unavailable in the corpus. Hence, we now propose an alternative to the fixed concept weight matrix $\mathbf{W}$ to predict concepts for unseen documents. A document d in its raw form can be represented as a bag-of-words vector $b_d \in \mathbb{R}^V$, containing its word-counts or TF-IDF scores (for example). We introduce a new FCM component, the Concept Allocator Network **(CAN),** that takes as input a bag-of-words vector b_d and generates its concept probability distribution p_d. **CAN** is a fully-connected multilayer neural network with H hidden layers, each of size h and *tanh* non-linear activations between its hidden-layers. A *softmax* activation after its final layer transforms its output to be valid probability distribution. Formally, **CAN** is defined in terms of the input layer matrix $\mathbf{M}^{(0)} \in \mathbb{R}^{V\times h}$, hidden layer matrices $\mathbf{M}^{(1)} \ldots \mathbf{M}^{(h-1)} \subset \mathbb{R}^{h\times h}$, and output layer matrix $\mathbf{M}^{(h)} \in \mathbb{R}^{h\times T}$. Each matrix is also associated with a bias vector, $m^{(0)}, \ldots, m^{(h-1)} \subset \mathbb{R}^h$ and $m^{(h)} \in \mathbb{R}^T$. The process of mapping a bag-of-words vector b_d to its concept probabilities p_d is given by the following:

$$x^{(0)} = \tanh(b_d\mathbf{M}^{(0)} + m^{(0)}) \tag{10}$$

$$x^{(j)} = \tanh(x^{(j-1)}\mathbf{M}^{(j)} + m^{(j)}) \quad \text{for}\sim j = 1, \ldots, h-1 \tag{11}$$

$$p^d = \text{softmax}(x^{(h-1)}\mathbf{M}^{(h)} + m^{(h)}) \tag{12}$$

where the *softmax* function transforms its vector-valued input into a probability distribution:

$$\text{softmax}(x)[i] = \frac{e^{x[i]}}{\sum_{j=1}^{T} e^{x[j]}} \quad \forall i = 1, \ldots, T$$

Thus, **CAN** replaces the concept weight matrix $\mathbf{W}$ to generate the concept probabilities p_d from the document's bag-of-words vector b_d. Like $\mathbf{W}$, **CAN** is jointly trained with the other FCM parameters. The hidden-layer size and number of hidden layers in **CAN** are hyperparameters that we tune via cross-validation.

Encouraging concept sparsity. In reality, we expect each document to embody only a few concepts, with the others being barely present or completely absent. This *sparsity* of the document-concept distribution is inspired by LDA and also easier to interpret. To enforce sparsity on the document-concept distribution, we append the product of the document-concept probabilities p_d to the loss function, transformed logarithmically to prevent numerical underflow (since the product of many probabilities will be a very small floating point number). This leads to the following "Dirichlet loss" term weighted by hyper-parameter λ, which approximately penalizes the document-concept distributions for having too many non-zero probability values:

$$\mathcal{L}_{dir} = \lambda\log(\prod_{d=1}^{D}\prod_{k=1}^{T} p_d[k]) = \lambda\sum_{d=1}^{D}\sum_{k=1}^{T}\log(p_d[k]) \tag{13}$$

Note that, while penalizing the L_0 norm $\|p_d\|_0\ \forall d = 1, \ldots, D$ would enforce sparsity exactly, it is non-differentiable, leading to issues when minimizing the loss function using gradient descent.

Penalizing the product of the probabilities (or the summation of the log-probabilities) as above serves to approximate the sparsity objective while remaining differentiable and efficient to compute.

Encouraging concept diversity. The model described so far tends to learn concepts that are highly overlapping, especially when a few concepts are significantly more prevalent in the corpus than others. To better capture less prominent but potentially important concepts, we introduce a novel extension that we term the "diversity regularizer". This regularizer encourages every pair of concept embeddings $\mathbf{E_t}[i], \mathbf{E_t}[j]$ to be dissimilar in terms of their dot-product. This is formulated as the following extension to the loss function:

$$\mathcal{L}_{\text{div}} = \eta \sum_{i=1}^{T} \sum_{j=i+1}^{T} \log \sigma(\mathbf{E_t}[i] \cdot \mathbf{E_t}[j]) \tag{14}$$

where η is a hyper-parameter that controls the strength of the prior, and the logσ log-sigmoid transformation ensures that this term and its gradient lie on the same scale as the other terms in the loss function. This regularization can be thought as finding diverse *prototypical* concepts that best correlate to Y.

3.3 Focusing Concepts on Target Outcomes

In practice, the concepts embodied by documents may fall into several different descriptive modes. For example, the set of concepts "furniture," "technology," and "kitchen" describe the *category* of product being sold, whereas the set of concepts "aesthetics," "functionality," and "reliability" describe *characteristics* of the product being sold; both these descriptive modes may exist simultaneously in the corpus, and our goal is to uncover the one that best explains the given *outcome, Y,* associated with each document.

Hence, we introduce a loss component that "focuses" the concepts toward explaining these outcomes. We assume that the target outcomes are binary, $y_d \in \{0, 1\} \forall d = 1, \ldots, D$, though extensions to real-valued outcomes are straightforward. We introduce a parameter vector $\theta \in \mathbb{R}^T$ that assigns an *explanation-weight* to each concept, and that is shared across all documents. Given the explanation weights θ and the document-concept distribution p_d, define $\hat{y_d}$ for a document d as a weighted combination of its concept probabilities:

$$\hat{y_d} = \theta \cdot p_d \tag{15}$$

Given the observed outcome y_d, we would like $\hat{y_d}$ to be large if $y = 1$ and small if $y = 0$. This requirement is captured by the following *cross-entropy loss* term that we append to the overall loss function weighted by hyper-parameter ρ:

$$\mathcal{L}_{\text{clf}} = \rho(y_d \log \sigma(\hat{y_d}) + (1 - y_d) \log(1 - \sigma(\hat{y_d}))) \tag{16}$$

Note that we could also add any user-specified X in Equation 15. This 1) increases prediction power and 2) allows managers to compare the correlational relative importance of mined concepts to key-referential X. We discuss this extension in Section 5.4 and in particular Equation 20.

3.4 Model Summary

In each training iteration, the input for the model is the pivot word w, the set of context words from the size-k window $C_k(w)$, the index d of the document, and the outcome y. The complete model incorporates all the losses defined in the previous sections, leading to the following loss function for each input iteration:

$$\mathcal{L}(w, C_k(w), d, y) = \mathcal{L}_{\text{neg}}(w, C_k(w)) + \mathcal{L}_{\text{dir}}(d) + \mathcal{L}_{\text{div}} + \mathcal{L}_{\text{clf}}(d, y) \tag{17}$$

To prevent longer documents (those with more words) from contributing disproportionately more to the loss, we also scale it by the length $l_d \in (0, 1)$ of each document which is inversely proportional:

$$\mathcal{L}_{\text{corpus}} = \sum_{d,y \in \text{Corpus}} \sum_{w \in \text{doc } d} \mathcal{L}(w, C_k(w), d, y) \times \frac{1}{l_d}$$

The model architecture is visualized in Figure 4. This diagram describes the raw input to the model, how the input constitutes the matrices to be estimated by the FCM, further processing by the model using the matrices, and the final output that enters the loss function to be minimized. This diagram describes how the data flows through the neural network model. Appendix F provides detailed step-by-step procedure of FCM training, inference, and implementation details.

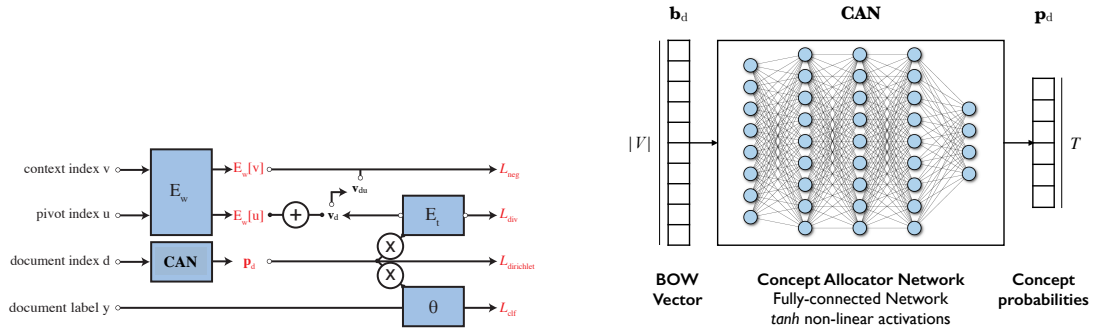


Figure 4: FCM Model Architecture

Describing Concepts Given the trained model, one way to describe concepts is to 1) get concept embedding vectors and 2) find the closest word vectors for each concept vectors. We take this approach and use the dot product distance.

3.5 Measure of Interpretability

We define three different metrics of interpretability to be used, as discussed in Section 2.1.

Coherence: This is a measure as defined by Mimno et al. (2011) from the topic modeling literature. It computes the sum of a pairwise score function on the top n words $w_1, w_2.., w_n$ used to describe each topic:

$$Coherence = \sum_{i<j} \log \frac{D(w_i, w_j) + 1}{D(w_i)} \tag{18}$$

where $D(w_i)$ is the count of documents containing the word w_i, and $D(w_i, w_j)$ is the count of documents containing both words w_i and w_j. Simply put, the coherence metric measures how well-focused the group of top topic key words are in describing a singular concept. A higher topic coherence means that the key words within one topic dimension are more coherent with each other in concept. Note that while the measure of coherence originates from topic modeling literature, it is directly applicable to any set of keywords. The topic modeling literature first came up with ways to detect "interpretability" by human judgment through mechanical turk (Chang et al., 2009), then the "coherence" construct was validated with domain expert taggers who "annotated [topics] as 'good'

if they contained words that could be grouped together as a single coherent concept" (Mimno et al., 2011). Next, automated measures were constructed that seem to perform as well as or better than humans (Mimno et al., 2011; Newman et al., 2010; Lau et al., 2014). For example, metrics based on word co-occurrences and mutual information based on an external corpus such as Wikipedia are more representative of how humans would evaluate a topic as interpretable (Newman et al., 2010). From the XAI literature perspective, this measure of interpretability fits the desiderata of measuring unambiguity (Ras et al., 2018) and selectivity of explanation (Lipton, 2016; Miller, 2018).

1st Human-judged - Number of Distinct Concepts Found: Using Amazon Mechanical Turk, we directly obtain the number of distinct concepts found in algorithm outputs. Each topic or concept would be more useful and interpretable if it describes fewer distinct concepts. This parallels the coherence measure but is more direct.

2nd Human-judged - Usefulness of Algorithm Output for the Particular Target Task: We use the definition of interpretability from Dhurandhar et al. (2017)—which states "AI is interpretable to the extent that the produced interpretation I is able to maximize a user's target performance"—and operationalize it in our case. As the next section elaborates, the data context is the consumer purchase journey and reviews read. Thus, the target goal here is making a purchase decision. Therefore, if we apply the definition of Dhurandhar et al. (2017), the algorithm output that extracts concepts from reviews that are more helpful for making a purchase decision should be considered more interpretable. We ask Amazon Mechanical Turkers to provide the usefulness of algorithm outputs for making a purchase decision.

3.6 Measure of Predictive Performance

To measure the performance of different models, we use the receiver operating characteristics (ROC) curve, which compares the true positive rate (TPR) against the false positive rate (FPR) at different discrimination thresholds. The Area Under the ROC curve is called AUC, which captures the probability that a model ranks a randomly chosen positive sample higher than a randomly chosen negative sample. In general, a classifier with higher AUC tends to have better predictive performance. For example, a random guess classifier yields an AUC of 0.5, while a perfect model yields an AUC of 1. Additionally, we show simple accuracy, precision, recall, and F1 score.[7]

4 Demonstration of FCM on a Novel Data

We apply FCM to a dataset from a top consumer-review platform to demonstrate efficacy.

4.1 Raw Data

Data comes from an online retailer in the UK through a top review platform company. They track 243,000 consumers over the course of two months in February and March of 2015 in the electronics and home & garden categories. There are 41 different subcategories, as shown in Appendix A. The data tracks consumer page views, review-reading behavior, clicks, and transactions. That is, the data includes the typical clickstream and conversion data plus consumers' review-reading behaviors, which is essential for FCM application. The data also records (1) when a user clicks on review pages, (2) whether each review has appeared on a user's browser, and (3) for how long the content was

[7] Measures are defined as accuracy (the total % correctly classified), precision (out of predicted positives, how many are actually positive), recall (out of actual positives, how many are predicted as positives), and $F1 = \frac{2*Precision*Recall}{Precision+Recall}$ (the harmonic average of precision and recall).

viewed on the user's browser, measured accurately down to milliseconds. With these data, we assume that if a review appeared on a user's browser, the user has read the review.

4.2 Processed Data for FCM

From the perspective of a user's shopping procedure, we can imagine a "decision-making journey" that characterizes how a user purchases or abandons a particular product. In such a journey, a user will first visit the product page to read the description, then read reviews of the product, and finally decide whether to buy the product or not. Accordingly, our dataset is at the "decision-making journey" or at the `UserID-ProductID` level. A data sample contains (1) the product review texts read by the user, (2) the explanatory variables shown to matter in predicting the purchase conversion in business literature (e.g., product price and review star ratings), and (3) a binary label indicating conversion. Next, we discuss selection criteria and data construction.

Selection Criteria & Data Frame Construction

We first define the scope of `UserID-ProductID` pairs ("journey"). These pairs are used to identify the journeys and serve as the primary key for our constructed data frame. We filter `UserID-ProductID` pairs to only include users who have read reviews, since FCM requires meaningful text linked to Y.

On the website, reviews are presented in groups of five. Consumers can read more reviews by clicking on the review page numbers. For each journey, we collect all the reviews that the consumer has read, sort them by browsing time, and concatenate them into a single document. This constitutes the text data of interest. As 88% of the journeys have fewer than 10 reviews read, we take the final 10 reviews read by consumers before they purchase or abandon.

Lastly, as the total conversion rate is 1.37%, there are many more journeys that ended in abandon (negative label) than conversion (positive label). Considering that the imbalance might negatively affect the performance of the trained FCM, we under-sample the negative pairs to reduce the class imbalance, achieving an abandon-to-purchase ratio of roughly 77:23. Finally, our constructed data frame is left with $58,229$ journeys of $30,218$ unique consumers and $6,612$ unique products. Of these journeys, $13,094$ yield user purchases.

Table 2 shows the summary statistics of the X at product, user, or journey levels.

Variable	**Var-Level**	**Definition**				**Count**
Product ID	Product	Total # of products				6612
User ID	User	Total # of unique users who have read reviews				30218
Content ID	Product	Total # of reviews				87593
			Mean	**SD**	**Min**	**Max**
Price	Product	Price of the product	63.28	84.41	0.24	1049
Rating Average	Product	Avg rating of reviews available for the product	4.28	0.49	1	5
Total # Pg views	Product	Total # of page views for a particular product	141.79	178.02	2	3280
Total # Reviews	Product	Total # of reviews available for the product	66.37	117.27	1	1463
Total Wallet Size	User	Total dollar value of transaction made by a user	104.68	171.76	0	3968.79
Total # of Rev-read	User	Total # of reviews read by a user	48.73	75.22	1	2675
User-Page views	Journey	Total # of page views for a user-product pair	3.25	2.64	1	100
User-Page # of Rev-read	Journey	Total # of reviews read in a user-product pair	10.90	13.85	1	376

Table 2: Variable Summary Statistics

5 Results

We first present FCM's interpretability performances against the benchmarks: 1) extraction of concepts with **higher coherency** (Section 5.1) and 2) **higher recall** (Section 5.2) of concepts. Here, we reiterate and provide brief descriptions of the benchmarks discussed in Section 2.2 with the citation for interested readers.

Interpretable Models

- **LDA + LR:** Plain LDA + logistic regression classifier. Plain LDA is the most widely used topic model and logistic regression makes this combination easy to interpret.
- **SLDA + LR:** Supervised LDA proposed by Blei and Mcauliffe (2008); Zhu et al. (2012) + logistic regression classifier. Supervision by Y could potentially improve the topic interpretability.
- **SeedLDA + LR:** Seeded LDA proposed by Jagarlamudi et al. (2012) + logistic regression classifier. We seed the topics based on dimensions of price and quality as defined in the literature and discussed in Table 3 to maximize its performance. These topics should perform better since it better captures the data generating process (of people reading the content that matters). Note that this is not a viable competitor to FCM as it requires apriori defined constructs.
- **Structural Topic Model + LR:** Structural Topic Model proposed by Roberts et al. (2014). Incorporates both X and natively handles Y.

5.1 Results on Higher Interpretability (Coherency) of Mined Concept

Higher interpretability of FCM-recovered concepts in comparison to LDA and variants are presented with 3 different metrics.

5.1.1 Higher Coherency: Comparison Based on Existing Metric *Coherence*

We first utilize the coherence measure in Equation 18. Figure 5 shows five topics (topic number decided by perplexity) recovered by best performing plain LDA out of 50 runs with average coherence of -2.25. Two experts then manually inspected the topics and color-coded similar conceptual words for ease of interpretation. Note the following observations: First, within a given topic, many different concepts appear—this is called *word intrusion* (Chang et al., 2009). For example, Topic 1 has concepts related to product names (orange), features (dark red), and aesthetics (green). Second, single concepts appear across many different topics—we call this *concept diffusion*. Many words related to singular concepts such as aesthetic (green), product name (orange), or features (dark red), appear in most if not all of the five topics. Presented with such outputs, it is unclear how managers may then utilize recovered topics as X for further insight.

Table 6 displays the concept-describing words for FCM-extracted concepts generated as described in Section 3.4. Compared to the topics found by the standard LDA, the FCM's concepts are more semantically coherent within each concept. Concepts are both well focused within (not intruded) as well as separated from one another (not diffused). FCM obtains an average coherence of -1.86, which is greater (higher interpretability) than the -2.25 obtained by LDA. Among the five concepts, the aesthetics concept represented by words such as "nice little big clean look design" achieved the highest coherence of -1.384. Even for the concept of serviceability, which has the lowest coherence score of -2.197, the words are semantically coherent. Based on 50 runs, the LDA coherence range lies in (-3.5, -2.25) while the FCM range lies in (-1.92, -1.68). The ranges do not overlap and are statistically significantly different, suggesting that FCM excels in extracting coherent (and thus more human-interpretable) concepts from review texts.

In fact, the concepts recovered by FCM closely coincide with the dimensions of product price and quality that are shown in the literature to influence consumer purchase. Garvin (Garvin, 1984, 1987) compiled and introduced a set of quality and price dimensions aimed at helping organizations think about product, as shown in Table 3. On multiple FCM runs, we were able to automatically extract most concepts compiled by Garvin. This serves as evidence of the external validity of FCM.

Table 6 provides the best coherence of 4 other interpretable baselines (50 runs and statistically significantly different from FCM's). FCM is superior to all of them. Appendix E presents actual examples of reviews along with computed concept composition while Appendix C shows topic words for all LDA variants (e.g., SLDA, STM, Seeded LDA).

Topic 0: phone + iron + lamp + **find** + **feature** + steam + **replace** + **old** + **design**
Topic 1: **sound** + picture + **old** + **brilliant** + battery + son + **feature** + **problem** + **smart** + **fantastic**
Topic 2: kettle + **clean** + microwave + **quick** + **size** + toaster + **design** + **heat** + **brilliant** + **quickly**
Topic 3: **assemble** + **sturdy** + room + curtain + **space** + **size** + **hold** + bin + **perfect** + **design**
Topic 4: bed + **comfortable** + cover + **lovely** + pillow + duvet + floor + mattress + **feel** + **thin**

Figure 5: **Topics Extracted by LDA:** Best coherence out of 50 runs. Average topic coherence is -2.25. We manually color-coded similar concepts for easy visualization. Appendix C has more details.

Concept-Describing Words	Concept Title
small nice space little use design clean size love look	Aesthetics
money quality cheap poor instruction price fine ok overall work	Price (Value)
come item easily perfect long expect fit feel job bit	Features
happy excellent definitely pleased far purchase need recommend worth time	Conformance
problem work lovely room day replace return job use far	Serviceability

Figure 6: **Concepts Extracted by FCM:** Concept representative words are found to compare to LDA. Average Concept Coherence is -1.86. Appendix C has more details.

Dimension	Description
Aesthetics	The review talks about how a product looks, feels, sounds, tastes, or smells.
Conformance	The review compares the performance of the product with pre-existing standards or set expectations.
Durability	The review describes the experience with durability, product malfunctions, or failure to work.
Feature & Performance	The review talks about the presence or absence of product features.
Brand	The review talks about indirect measures of the quality of the product, such as the reputation of the brand.
Price	The review contains content regarding the price of the product.
Serviceability	The speed, courtesy, competence, and ease of repair in case of any defects with the product.

Table 3: **Literature-defined Key Dimensions of Price and Quality**

5.1.2 Higher Coherency: Comparison Based on 2 Human-Judged Metrics

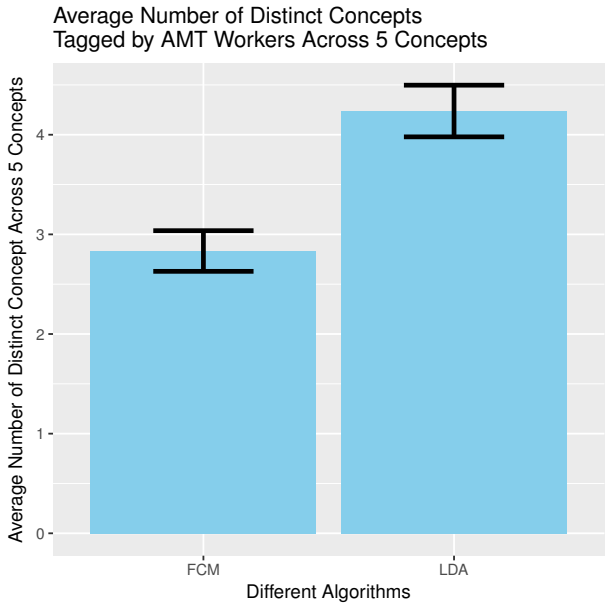
Two human-judged measures of interpretability (Section 3.5) are obtained from two distinct survey instruments (Appendix B). In both surveys, we show the outputs from both FCM and LDA to Amazon Mechanical Turkers.

For the *first* human-judged measure, we ask turkers how many distinct concepts they see in algorithm outputs at the topic or concept level. A smaller number signifies that concept or topic is more focused and concentrated in meaning, as well as less ambiguous, and thus more interpretable.

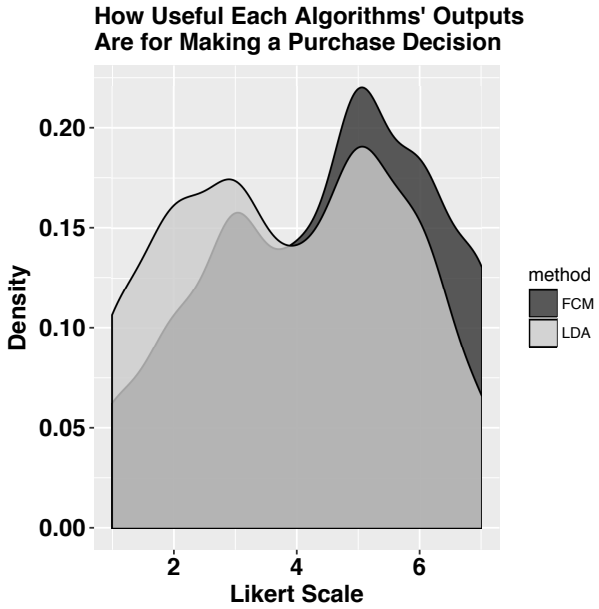
For the *second* measure, turkers are told to imagine a hypothetical situation in which they are shopping online and making a purchase decision. We show them the outputs of FCM and LDA and state that these are topic keywords of product reviews. Next, we ask them to rate on a Likert-like scale how useful these reviews would be in making a purchase decision.

Figure 7a shows that turkers on average found 2.83 distinct concepts in each FCM concept and 4.23 distinct concepts in each LDA topic. The t-test of difference in mean was statistically significant at the p-value $= 8.19 \times 10^{-8}$. This further confirms that FCM is able to produce concepts that are less ambiguous and more focused compared to LDA, and thus more interpretable.

Figure 7b shows the result of the second survey. The X-axis shows the value of the Likert scale where 1 means the algorithm's output was "extremely not useful" for making a purchase decision while 7 means "extremely useful." We first note that both distributions are bimodal, suggesting that some concepts and topics were useful while others were not. Second, FCM on average scored 4.43, while LDA scored 3.85, suggesting that FCM outputs were more helpful. T-test of difference in mean was statistically significant at the p-value $= 1.472 \times 10^{-5}$. Taken together, human judgment metrics find FCM more interpretable compared to LDA outputs. We repeat the experiments with all LDA variants and obtain qualitatively the same results.



(a) **Human-Judged Number of Concepts in FCM vs. LDA:** Standard errors are shown.



(b) **Human-Judged Usefulness of Concepts Found by FCM vs LDA for Making a Purchase Decision**

Figure 7: **Human-Judged Measures of Interpretability for FCM vs. LDA**

5.2 Results on Higher Recall (More Unique Concepts Recovered)

This section shows that FCM discovers a higher number of unique concepts compared to benchmarks.

5.2.1 Higher Recallability of Known Concepts from Main Dataset

We run an empirical experiment that compares the concept-recallability of FCM against LDA/SLDA on the main dataset in which the "ground truth concepts" are supplied from the existing literature. For e-commerce review corpus $\mathcal{D}$, Garvin (1984) provides dimensions of product price and quality (Table 3) known to influence consumer purchase decisions. While not exhaustive, Garvin's list is well-established and reasonable "ground truth concepts" for this experiment. Garvin's concepts,

$$\mathcal{C}(\mathcal{D}) := \{c_i\}|_{i=1,...,6} = \{\text{Aesthetics, Brand, Conformance, Features, Serviceability, Value}\}$$

are operationalized by manually assigning a set of 10 Garvin's concept words

$$W_i^{(\text{Garvin})} = \{w_{i1}^{(\text{Garvin})}, w_{i2}^{(\text{Garvin})}, ..., w_{i10}^{(\text{Garvin})}\}$$

to each of the Garvin's concept c_i. The concept words are chosen by 2 experts such that 1) the selected set of words in $W_i^{(\text{Garvin})}$ must be semantically relevant to Garvin's concept c_i, and 2) attain high term frequencies from data. Appendix C lists our operationalized concept words based on Garvin's theory. Our results are robust to different operationalizations.

We represent T concepts/topics produced by method $m \in \{FCM, LDA, SLDA\}$ by a set of its top 10 words (e.g. highest topic-word probability for LDA/SLDA, concept-word distance for FCM) as follows:

$$\hat{\mathcal{W}}^{(m)} = \{\hat{W}_1^{(m)}, \hat{W}_2^{(m)}, ..., \hat{W}_T^{(m)}\}$$

where

$$\hat{W}_t^{(m)} = \{\hat{w}_{t1}^{(m)}, \hat{w}_{t2}^{(m)}, ..., \hat{w}_{t10}^{(m)}\} \text{ for } t = 1, 2, ..., T.$$

Then, we say that method m successfully *discovers* the Garvin's concept c_i, if there exist concept/topic representing words $\hat{W}_t^{(m)}$ such that the number of overlapping words is greater than some threshold $h_{overlap}$:

$$\text{For the given } i\text{-th Garvin's concept, } \exists t \in \{1, 2, ..., T\} \text{ s.t. } |\hat{W}_t^{(m)} \cap W_i^{(\text{Garvin})}| \geq h_{\text{overlap}}.$$

Finally, recallability is the total number of Garvin's concepts recovered by different method m.

Table 4 shows the number of concepts discovered by models with different $h_{overlap}$ configurations. Runs refer to algorithm repetition with different initializations. With just 5 runs, FCM recovers more "ground truth concepts" than LDA/SLDA can in 50 or even 150 runs. The result is robust to different initializations of Garvin's concept words.

In summary, FCM's recallability of known concepts are superior to benchmarks.

Method	$h_{\text{overlap}} = 3$	$h_{\text{overlap}} = 4$	$h_{\text{overlap}} = 5$
LDA (Total Runs=50)	1	1	0
LDA (Total Runs=150)	3	2	0
SLDA (Total Runs=50)	3	2	0
SLDA (Total Runs=150)	3	3	1
FCM (Total Runs=5)	5	5	4

Table 4: **The Number of Garvin Concepts Discovered**

5.2.2 Higher Relative Recallability of Unknown Concepts

We present empirical studies quantifying the relative concept-recallability of any concept-extraction methods with respect to the generic corpus $\mathcal{D}$ and use this study to compare SLDA and LDA against FCM on our main dataset. Let A and B be two concept-extraction methods that operate on $\mathcal{D}$ and produce collections of concepts $\hat{\mathcal{C}}^{(A)}(\mathcal{D})$ and $\hat{\mathcal{C}}^{(B)}(\mathcal{D})$. Note that the corpus is generic in the sense that true concepts $\mathcal{C}(\mathcal{D})$ are *unknown apriori*. We approximate $\mathcal{C}(\mathcal{D})$ with the union of all the concepts recovered by the methods being compared:

$$\mathcal{C}(\mathcal{D}) \approx \hat{\mathcal{C}}^{(A)}(\mathcal{D}) \cup \hat{\mathcal{C}}^{(B)}(\mathcal{D}) = \hat{\mathcal{C}}(\mathcal{D})$$

We can then measure the *coverage* of each method $i \in \{A, B\}$ with respect to corpus $\mathcal{D}$ as the fraction of the concept space recovered by method i from that corpus:

$$\text{coverage}(m_i, \mathcal{D}) = \frac{|m_i(\mathcal{D})|}{|\hat{\mathcal{C}}(\mathcal{D})|} \tag{19}$$

Higher coverage corresponds to better relative recallability, which we operationalize as follows:

1. Run SLDA and FCM on the main dataset to extract $T \in \{5, 10\}$ topics/concepts.

   An example of the topics extracted by SLDA with $T = 5$ is:

   - **SLDA 1:** use clean time price excellent work simple small money far
   - **SLDA 2:** clean use feel like love price new old quality recommend
   - **SLDA 3:** look assemble fit sturdy nice quality room lovely small item
   - **SLDA 4:** use clean work small time love excellent little look job
   - **SLDA 5:** work use feature quality excellent love old price recommend time

   An example of the concepts extracted by FCM with $T = 5$ is:

   - **FCM 1:** problem work lovely room day replace return job use far
   - **FCM 2:** money quality cheap poor instruction price fine ok overall work
   - **FCM 3:** come item easily perfect long expect fit feel job bit
   - **FCM 4:** small nice space little use design clean size love look
   - **FCM 5:** happy excellent definitely pleased far purchase need recommend worth time

2. Take the union of the $2T$ topics and concepts extracted by SLDA and FCM and cluster them to create the approximate concepts $\hat{\mathcal{C}}(\mathcal{D})$. Clustering is performed using k-Means with 10 "k-Means++" initializations. The best value of k is chosen using the silhouette score. The "distance" between a pair of concepts is measured by the number of words they have in common. Continuing the example above, clustering results in the following $|\hat{\mathcal{C}}(\mathcal{D})| = 4$ concepts:

   - Concept Cluster A:
     - small nice space little use design clean size love look
     - look assemble fit sturdy nice quality room lovely small item
     - use clean work small time love excellent little look job
   - Concept Cluster B:
     - money quality cheap poor instruction price fine ok overall work
     - happy excellent definitely pleased far purchase need recommend worth time
     - use clean time price excellent work simple small money far
     - clean use feel like love price new old quality recommend

  - work use feature quality excellent love old price recommend time
- Concept Cluster C:
  - come item easily perfect long expect fit feel job bit
- Concept Cluster D:
  - problem work lovely room day replace return job use far

3. Compute the number of distinct concept clusters that contain the topics/concepts extracted by SLDA and FCM respectively. This corresponds to $|m_i(\mathcal{D})|$. Continuing the example above, the topics extracted by SLDA belong to 2 distinct concept clusters. Topics SLDA 1, 2, and 5 belong to concept cluster B while SLDA 3 and 4 belong to concept cluster A. On the other hand, FCM outputs belong to 4 distinct concept clusters: FCM 1 => cluster D, FCM 2, 5 => cluster B, FCM 3 => cluster C, and FCM 4 => cluster A. We then compute the coverage for each method using coverage equation 19. Thus, coverage(SLDA, $\mathcal{D}$) = 2/4 and coverage(FCM, $\mathcal{D}$) = 4/4.

We repeat the procedure above for 5 runs and report the mean coverage for SLDA and FCM for $T \in \{5, 10\}$ in Table 5. A paired two-sample t-test finds the difference in the mean coverage of SLDA and FCM to be significant at the 5% level. We repeat the experiment above for LDA and FCM without supervision and obtain the same result.

To summarize, FCM is superior in relative-recallability of concepts compared to benchmarks.

Method	$T = 5$	$T = 10$
SLDA	0.510	0.781
FCM	1.000	0.868
p-value	0.026	0.033

Method	$T = 5$	$T = 10$
LDA	0.493	0.698
FCM without supervision ($\rho = 0$)	1.000	0.763
p-value	0.023	0.031

Table 5: **Mean coverage for SLDA vs. FCM and LDA vs. FCM on Main Dataset**

5.3 Sanity Check on Predictive Performance

Focusing by user-specified variable is optional and provides user with filtering ability. But when Y is supplied to focus, we can also calculate the prediction performances to see if FCM passes the sanity check of achieving acceptable predictive performance. Using the main dataset, FCM takes the X and the read reviews as inputs to predict a purchase/abandon (y). To benchmark, in addition to the original 4 interpretable models, we run the following uninterpretable prediction-focused models[8].

Uninterpretable or Prediction-Focused Models

- **BOW+LR:** Bag-of-words approach + logistic regression classifier.
- **Sentiment + LR:** Review-level sentiment-labelled classification.
- **CNN with GLoVe:** Deep Learning models excel at prediction tasks. We also utilize GLoVe word embedding (Pennington et al., 2014) as the first layer.
- **XGB:** eXtreme Gradient Boosting (Chen and Guestrin, 2016) is a boosted tree model often known to achieve the best predictive performance in a wide variety of ML competitions off-the-shelf. Please see Footnote 3.

[8]For predictive performance measurement, data is split into 70% training, 15% validation, and 15% test sets. The model is trained on up to 500 different parameter configurations and our model gives stable results across these sets. All results hereafter are produced by the model trained under the configuration $\lambda = 10, \rho = 1000, \eta = 1000$.

We report test set accuracy, precision, recall, F1-score, and ROC AUC, in Table 6. Figure 8 shows the ROC curves. We compare FCM's predictive performance against two sets of baseline models—interpretable models vs. uninterpretable prediction-focused models.

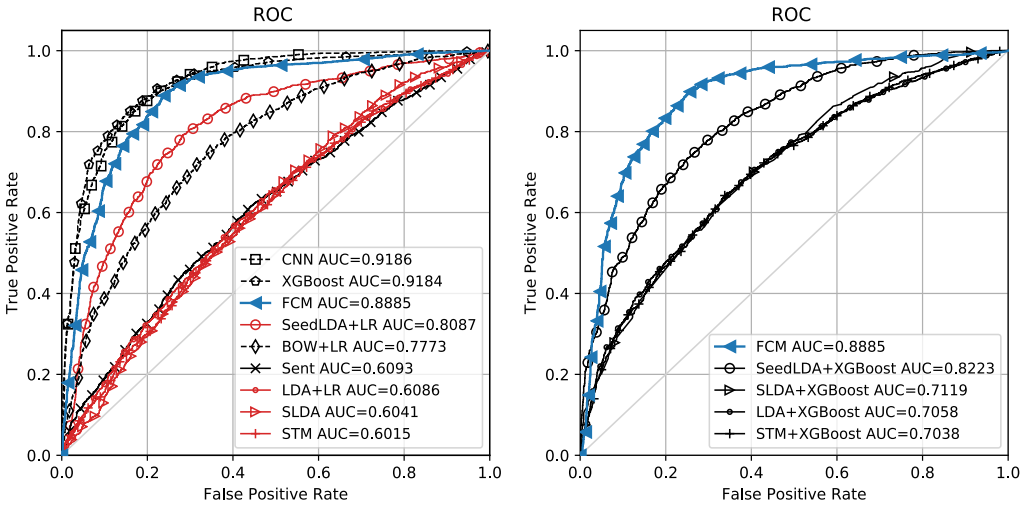


Figure 8: **Receiver Operating Characteristics Curve of FCM vs. Baselines:** Right ROC curve replaces all LR with XGBoost for increased AUC at the cost of interpretability. See Table 6 for full performance table.

Figure 8(left) presents the ROC curves and the AUC values for the FCM (blue), 4 interpretable baselines (red), and 4 prediction-focused baselines (black). First note that all interpretable models (red) fall significantly behind FCM (blue). While there are two uninterpretable algorithms (black) that surpass FCM, the difference is rather small at less than 0.03 in AUC. Two uninterpretable algorithms perform worse than FCM. We also provide ROC/AUC after replacing logistic regression with XGB for topic model based approaches in Figure 8(right). Note that performance increases (yet still falls behind FCM) at the cost of interpretability, since XGBoost does not provide coefficients like logistic regression.

As noted in Footnote 3, the top two performing algorithms are, unsurprisingly, deep learning and XGB. CNN with GLoVe embedding achieves the highest AUC at 0.9186 with XGB performing nearly the same at 0.9184. FCM follows closely at 0.8885. Given that we could also boost (reduce bias) and bootstrap aggregate (reduce variance) FCM predictions, albeit at a higher computational cost, FCM performance is competitive with the cutting-edge prediction-focused methods. In fact, we can turn FCM into an uninterpretable prediction-focused CNN by increasing classification hyperparameter ρ arbitrarily high, while setting interpretability-related parameters (λ, η) to zero. Contrastingly, FCM performs significantly better than the traditional bag-of-words approach (0.7773), the basic sentiment analysis (0.6093), and all topic model variants + XGB.

Among the interpretable competitors, the best is the seeded LDA (0.8087). Our seed words were driven by existing theory in consumer purchase behavior, as will be elaborated in Section 5.1.1 and Table 3. With these apriori known topics that better represent data-generating processes, it is unsurprising that seeded LDA excel over a naive bag-of-words approach. However, seeded LDA still falls short of FCM, suggesting that FCM extracts residual signals above and beyond theory-driven concepts. Note that seeded LDA approach can be useful when managers are equipped with domain-knowledge but not feasible for exploratory concept extraction. Other interpretable models, such as vanilla LDA (0.6086), supervised LDA (0.6041), and Structural Topic Models (0.6015), all

	Coherence	Accuracy	Precision	Recall	F1 score	AUC
FCM	-1.86	0.8228	0.8346	0.8475	0.8410	0.8885
Interpretable Benchmarks						
LDA + LR	-2.25	0.5653	0.5624	0.9630	0.7101	0.6086
SLDA + LR	-2.17	0.5857	0.6163	0.6641	0.6393	0.6041
STM + LR	-5.13	0.5872	0.5977	0.7745	0.6747	0.6015
Seeded LDA+LR	-1.95	0.7490	0.7945	0.7365	0.7644	0.8087
Uninterpretable Benchmarks						
BOW+LR	N/A	0.7212	0.7231	0.8033	0.7611	0.7773
Sentiment	N/A	0.5804	0.5896	0.7925	0.6762	0.6093
CNN with GloVe	N/A	0.8421	0.8307	0.8973	0.8627	0.9186
XGB	N/A	0.8475	0.8525	0.8757	0.8639	0.9184
LDA + XGB	N/A	0.5684	0.5623	0.9902	0.7173	0.7058
SLDA + XGB	N/A	0.5892	0.5740	0.9964	0.7284	0.7119
STM + XGB	N/A	0.6539	0.6637	0.7581	0.7077	0.7038
Seeded LDA + XGB	N/A	0.7408	0.7274	0.8495	0.7837	0.8223

Table 6: **Prediction Performance Against Competing Methods**

perform worse than FCM, even with good parameter tuning effort. Appendix H presents FCM's predictive performance against baselines for a well-known benchmark dataset called 20-Newsgroup, which shows that FCM excels even over XGB in some cases.

FCM excels in predictive performance over all interpretable baselines while staying competitive with the top uninterpretable prediction-focused algorithms.

5.4 Correlational Importance of Concepts for Gauging Economic Significance

Concepts	Coef	Explanatory variables	Coef
Aesthetics	0.107	Price	-0.003
Conformance	-0.076	Avg Rating (std)	0.288
Features	0.058	User-Page Views	-0.005
Value	0.054	Prod Tot # Reviews	0.002
Serviceability	-0.083	Prod Tot # Pg views	-0.0001
		Total Wallet Size	0.0048
		User-Pg # of Rev Read	0.012

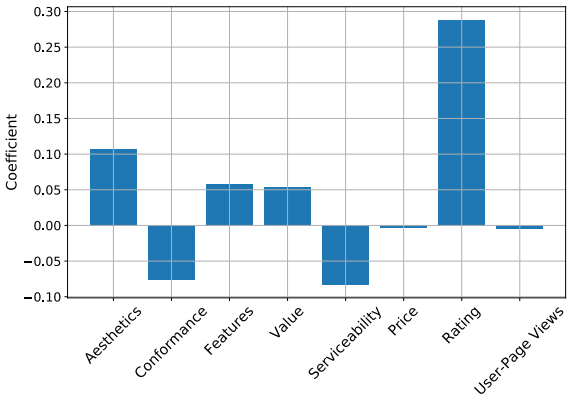


Table 7: **Estimated Coefficients**

To better utilize the extracted concepts, FCM provides correlational importance of mined concepts against the user-input X. The idea is to supply FCM with relatively well-understood X (e.g., price) along with texts to compare the correlational impact on the Y. In the last layer, FCM predicts the conversion (Equation 15) of a journey with the mined document-concept distribution, p_d. We modify this prediction layer to include user-specified X. We rename p_d as $DocConceptD$ for clarity

and add the explanatory variables $ExpVar$ with a sigmoid function:

$$\hat{Conversion} = \sigma(\theta_0 + \sum_i \theta_i DocConceptD_i + \sum_j \theta_j ExpVar_j) \tag{20}$$

where $DocConceptD$ is a probability vector of different concepts that sums up to 1. For this exposition, concepts are named according to Figure 6. The coefficients here speak to the impact of concept volume present in the review documents and the trained weights, θ, characterize how much the predicted conversion will respond to the change in X. Although the sigmoid layer of FCM follows the formula of a generalized linear regression, we are not aware of any work that could provide the confidence interval of a deep learning-based model. Thus, we do not provide the confidence interval.

Table 7 shows the trained coefficients. We standardized the average rating for easier interpretation. The results pass the sanity check: a negative coefficient for price and a high positive coefficient on average ratings. Aesthetic concepts had the highest positive correlation with conversion while the serviceability and return-related concepts had the lowest. Calculating the average correlation across each journeys shows that a 10% concept increase is associated with a -20% to 15% change in predicted conversion probability. Given the FCM results, a manager may then launch a more focused causal study to investigate how to prioritize certain concepts and information in customer reviews. Other ideas are discussed in Section 6.

5.5 Experiments on {Interpretability-Accuracy, Additional Data, Role of Y-focusing}

5.5.1 Interpretability-Accuracy Relationship

This section explores the interpretability-accuracy relationship in FCM. In the topic modeling literature, Chang et al. (2009) show that the model fit (perplexity) is negatively correlated to interpretability (coherence). In the context of object classification in computer vision via CNN, Bau et al. (2017) report experimental findings that the interpretability of a model can be decreased without changing its discrimination ability. The authors conclude that the interpretability of a model is not required for high accuracy. Zhang et al. (2017) develop an interpretable convolution neural network and in the process show that there is a trade-off between interpretability and classification accuracy.

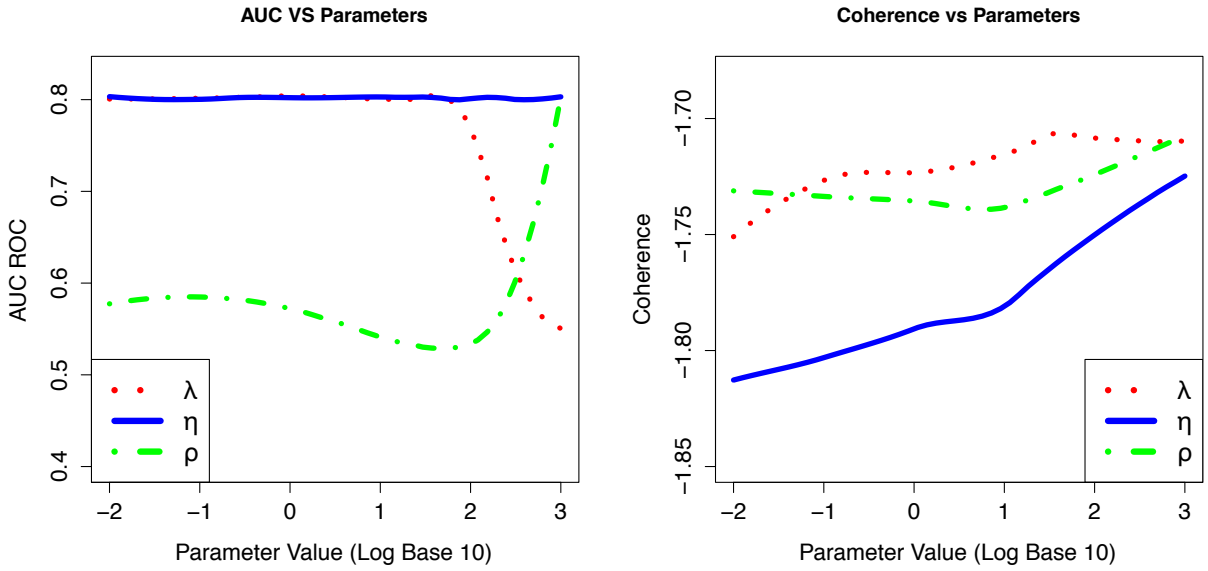
As FCM's objective function consists of different components, we can directly see how increasing certain weight on the objective function changes the accuracy vs. interpretability. For example, we explore how increasing ρ influences accuracy and coherence. Increasing ρ should also increase the accuracy, but it is not clear if it will have the same impact on coherence. In particular, we examine ρ (classification), η (concept diversity), and λ (concept sparsity). The first plot in Figure 9 shows the impact on AUC as we vary ρ, η, and λ, and the second plot shows the impact on coherence. For each parameter, we vary the parameter from 10^{-2} to 10^3 in 20 equally spaced points while repeating each point three times, giving us a total of 60 points per parameter. While one parameter is varied, the other two are fixed. The results are then smoothed with a LOESS (locally estimated scatterplot smoothing) plot.

For AUC (predictive power), ρ has an expected trend. As ρ increases, AUC increases. On the other hand, λ shows the opposite pattern. As we force sparsity of concept, the predictive accuracy decreases—a clear loss in signal. Interestingly, concept diversity, η, does not seem to influence the AUC. From the geometric point of view, in the concept vector space, there may be clusters of several different concepts. Increasing η does not seem to decrease predictive information, since it forces

each concept to cover different regions in the concept space, as opposed to increasing sparsity, which does decrease predictive signals.

For coherence (interpretability), neither λ nor ρ seem to have a clear trend. Both have a slight upward trend, but the range is not so large nor the patterns so clear. However, in comparison, η seems to have a clear upward pattern. As we increase the concept diversity, coherence and interpretability increases.

We document several interesting findings: 1) The interpretability-accuracy trade-off correlation depends on different parameters of the model; 2) Increasing ρ increases AUC as intended and slightly increases interpretability; 3) Increasing concept sparsity, λ, decreases AUC but doesn't influence interpretability; and 4) Increasing concept diversity, η, does not influence AUC while increasing interpretability.



(a) **Variation in ROC-AUC**

(b) **Variation in Coherence**

Figure 9: Variation in ROC-AUC and Coherence for different hyperparameter settings. Hyperparameters were varied for each while the other two were fixed.

5.5.2 Results on Different Dataset For Robustness - DonorsChoose.org

We apply FCM to another dataset. The open data is from DonorsChoose.org, a nonprofit crowdsourcing site that allows individuals to donate directly to public school classroom projects. This dataset spans over 6.2 million donations and 1.2 million fundraising projects from 2002 to 2016 at the project level. Since the project page usually provides little structured information to the individual donors other than the full text of teacher-written essays, the textual data should yield a significant insight in predicting the fulfillment of the donation goal.

We label the projects as either a success/fail based on fundraising goal reached. Training data consist of randomly sampled 10,000 positives and 10,000 negatives. Each row contains (1) the project essay text and (2) the binary project status as the label. We also control for the project funding amount. As with our main results, we set five concepts to be discovered. Extracted concept-describing words and estimated relative importance coefficients are found in Table 8. Once again, the

	Concept-Describing Words	Concept Title	Coef
1	camera history photography animal world trip experience picture video life	Photography & Outdoor Learning	0.0311
2	art material child supply color easel chair time pencil center	Art Supplies	0.0868
3	music play drum instrument equipment song sensory fitness physical keyboard	Music & Physical Ed	0.0188
4	book read reading novel text library reader level love language	Reading & Literature	0.0597
5	technology math computer ipad lab project able science allow laptop	Tech Equipment	-0.0689

Table 8: **Concepts Extracted by FCM for the DonorsChoose Dataset**

concepts are semantically highly coherent and well separated from one another. We manually name each concept with descriptive titles for convenience, and find that the extracted concepts consist of essentially five different curriculum types. The estimated coefficients imply that art supply-related (0.0868) or literature-related (0.0597) concepts are more likely to be successfully funded. FCM also suggests that technology-equipment funding requests are less likely to be successful. Appendix D provides full results including interpretability and prediction performance benchmarks.

5.5.3 The Role of Y-Focusing - Examples from DonorsChoose.org

This section investigates the impact of Y-focusing on FCM output—i.e., what concepts are extracted if a different Y is selected to guide the concept discovery? For this, DonorsChoose data is used again due to the availability of a different Y.

Y-Variable: $NotFunded$

	Concept-Describing Words	Concept Title	Coef
1	teach time new allow day tool board lesson special	Special Teaching Tool	-0.0402
2	child create project material provide come like activity education	Project Materials	0.0441
3	music play math science kit experience stem language opportunity	Music & STEM	-0.0224
4	book read reading text novel level library reader love english	Reading & Literature	-0.0275
5	technology skill ipad computer program able grade tablet app access	Tech Equipment	0.0461

Table 9: **DonorsChoose Result Guided by Y:**$NotFunded$**.**

One way to obtain a different Y is to simply flip the Y from Section 5.5.2 to predict $NotFunded$. FCM will extract out concepts that are highly correlated to the project being unsuccessfully funded. Intuition suggests that FCM will extract concepts that are repeated and perhaps will also yield new concepts. Table 9 presents the results. Comparing the results for the original (Table 8), we find that concepts "Tech Equipment" and "Reading & Literature" are extracted yet again, with the coefficients' directions flipped from the original dataset results. This shows FCM's consistency in both concept extraction and correlational coefficient estimation.

However, FCM also recovers slightly modified concepts or even new concepts not originally found in Table 8. Compare Concept 3 in both tables. It was originally "Music & Physical Education" and had positive correlation with successful funding. Now it is changed slightly into "Music & STEM" with a negative value for $NotFunded$. While the directions tell a consistent story, extracted concepts are slightly modified. Concept 1 is a new concept seeking to fund some sort of special teaching tool that will assist the teachers. The coefficient is negative, suggesting that it is likely to get funded.

Concept 2 refers to student project material requests and is positive, suggesting a lower chance of getting successfully funded.

Y-Variable: *Exciting*

	Concept-Describing Words	Concept Title	Coef
1	life opportunity experience world garden society culture grow hope	Experiential Learning	-0.0842
2	camera ipad technology video class projector able use tablet allow	Tech Equipment	0.0588
3	drum math calculator science music hand instrument game play teach	Music & STEM	-0.0972
4	art ball provide new material child writing like supply	Art Supplies	0.0049
5	book read reading center love time remember level listen player	Reading & Literature	0.1177

Table 10: **DonorsChoose Result Guided by Y:** *Exciting*.

The 2014 KDD (Knowledge Discovery and Data mining) conference hosted a data science competition with Kaggle.com (https://www.kaggle.com/c/kdd-cup-2014-predicting-excitement-at-donors-choose/data) using DonorsChoose data. The competition created a customized binary Y linked to each funding request called *Exciting* as defined and deemed important by DonorsChoose. *Exciting* goes beyond $SuccessfullyFunded$ used in Section 5.5.2 in that the project must (1) be $SuccessfullyFunded$, (2) have at least one teacher-acquired donor, (3) have a higher than average percentage of donors leaving an original message, plus more. For more details, please refer to the Kaggle link. In summary, *Exciting* is a domain-expert defined Y that is sufficiently different from $SuccessfullyFunded$ with correlation at 0.1484 – there are many $SuccessfullyFunded$ projects that are not *Exciting*.

Table 10 shows the results. Two concepts, 4 ("Art Supplies") and 5 (Reading & Literature), are recovered once again and the directions match the original result. Concept 2 ("Tech Equipment") is recovered once again but this time with reversed direction. While the tech equipment was negatively correlated with an successfully funded project, it is positive for the *Exciting* project. Concept 3 ("Music & STEM") is slightly modified from "Music & Physical Education" with reversed direction. Lastly, a new concept related to "Experiential Learning" is discovered to be negatively correlated to *Exciting*.

Summary

Even on the same data, FCM recovers both repeated and new concepts depending on the Y used to focus the concept. This enables creative use of FCM. For example, if a manager had return data connected to review-reading, FCM might recover review content that might help reduce product returns.

6 Managerial Implications & Envisioned Usage

Economic Value of Text and FCM The economic value of insights obtainable from text is well documented.

With respect to our primary dataset on consumer purchases and review-reading behavior, prior literature documents the economic value of identifying Garvin's dimensions in user-generated text. For example, Abrahams et al. (2015) report that textual data from social media can be used to discover product defects to inform quality management (serviceability), Archak et al. (2011) reveal that review text can be used to learn consumers' relative preferences for different product features

(features), and Netzer et al. (2012) use text from online forums to mine the market structure and uncover characteristics of the competitive landscape (brand). Lastly, Liu et al. (2019) extract Garvin's concept in review text to quantify the causal impact on consumer purchase behavior.

Connecting FCM method claims (higher interpretability and recall - Sections 5.1 and 5.2) to managerial impact, *assume* that we did not know much about dimensions of product price and quality that influence consumers. Running LDA models 50 or even 150 times failed to recover many Garvin's concepts. In comparison, FCM recovers most concepts in just 5 runs. This demonstrates the potential value of FCM for text data with unknown apriori insights. FCM's high interpretability provides values akin to XAI literature while high recall serves to discover potentially unseen insights.

In research papers that discuss the value of text, researchers either quantify (1) textual value without describing the actual content or (2) the values of concepts that are already proven to matter. The economic value of (1) describing coherent concepts from text data known to provide significant signals and (2) recovering previously unknown concepts in text, and thus the value of FCM, is evident.

Envisioned Usage Examples Other FCM applications are brainstormed. Applications arise due to the ability of FCM to 1) focus the mined concepts in "one-click," 2) provide predictions and extract concepts for previously unseen documents, and 3) retrain the learned model dynamically via stochastic gradient descent updates. Specifically, FCM can be used to dynamically monitor consumer feedback and complaints (i.e., *dynamic resonance marketing tool* (Clemons et al., 2006)) on social media and websites as a more exploratory version of techniques shown in Netzer et al. (2012) and Abbasi et al. (2019). Managers can benefit from a dashboard of daily or weekly summarization of consumer chatters using FCM in place and quickly see different aspects by focusing with appropriate outcomes. Similarly, when applied to reviews of a specific product, FCM could extract feature importance, which might inform product design. This would be a quicker (though less accurate) alternative to the method described by Timoshenko and Hauser (2018).

With respect to bias in algorithm literature (briefly discussed in Section 2.1), there is a dearth of literature on unstructured data. FCM could serve to detect content bias in text. For example, say a news site is using recommender systems to suggest articles to read. If the recommender (which is based on consumer history) repeatedly recommends particular content, this may lead to *filter bubbles* (Pariser, 2011; Lee and Hosanagar, 2019), whereby consumers only consume content they like or agree with. This could be harmful to general consumer welfare and to the platform (Sunstein, 2018). FCM could serve as an exploratory algorithm to detect content biases. Similar applications exist in social media and search engines.

7 Conclusions

We introduced a new deep learning-based text mining method, the Focused Concept Miner (FCM), to explore, organize, and extract information from textual data guided by any Y of business importance.

We envision FCM as an *exploratory tool* to make sense of the severely untapped textual data in the business world, and as a jumping off point for further focused causal studies that may lead to prescriptive policies. There are two broad use-cases for machine learning algorithms for managers and business researchers: (1) scale hypotheses testing, and (2) discovering hypotheses from empirical data. Extant papers already utilize ML to scale theory testing (see end of Section 2.3). Additionally, ML can serve as a navigator to point out interesting patterns and empirical generalizations that could potentially generate worthwhile hypotheses to be causally explored in more depth. Using ML

to augment hypothesis generation is ripe for serious consideration. Interpretable Machine Learning algorithms such as FCM can be instrumental. Ultimately, however, FCM is only as good as the user. For example, in our data, the online reviewers are self-selected and heterogeneous. FCM captures insight from the text *as is*, albeit while controlling for any variables. Only researchers who practice sound logic through domain knowledge can be good judges of what is spurious and what is worth further investigation. Appendix G discusses FCM's limitations and potential future extensions.

We hope managers and researchers can use FCM creatively with any combination of text, structured, and business outcome variables to glean insights and build out new hypotheses from rich text data.

References

Abbasi, A., J. Li, D. Adjeroh, M. Abate, and W. Zheng: 2019, 'Donť Mention It? Analyzing User-Generated Content Signals for Early Adverse Event Warnings'. Information Systems Research **30***(3), 1007–1028.*

Abbasi, A., Y. Zhou, S. Deng, and P. Zhang: 2018, 'Text analytics to support sense-making in social media: A language-action perspective'. MIS Quarterly **42***(2).*

Abrahams, A. S., W. Fan, G. A. Wang, Z. Zhang, and J. Jiao: 2015, 'An integrated text analytic framework for product defect discovery'. Production and Operations Management **24***(6), 975–990.*

Ananthakrishnan, U. M., B. Li, and M. D. Smith: 2020, 'A Tangled Web: Should Online Review Portals Display Fraudulent Reviews?'. Information Systems Research.

Angwin, J., L. Kirchner, S. Mattu, and J. Larson: 2016, 'Machine Bias: There's Software Used Across the Country to Predict Future Criminals. And it's Biased Against Blacks., May 2016'.

Archak, N., A. Ghose, and P. G. Ipeirotis: 2011, 'Deriving the pricing power of product features by mining consumer reviews'. Management Science **57***(8), 1485–1509.*

Bass, F. M.: 1995, 'Empirical generalizations and marketing science: A personal view'. Marketing Science **14***(3_ supplement), G6–G19.*

Bau, D., B. Zhou, A. Khosla, A. Oliva, and A. Torralba: 2017, 'Network dissection: Quantifying interpretability of deep visual representations'. arXiv preprint arXiv:1704.05796.

Blei, D. M. and J. D. Mcauliffe: 2008, 'Supervised topic models'. In: Advances in neural information processing systems. *pp. 121–128.*

Blei, D. M., A. Y. Ng, and M. I. Jordan: 2003, 'Latent dirichlet allocation'. Journal of machine Learning research **3***(Jan), 993–1022.*

Buschken, J. and G. M. Allenby: 2016, 'Sentence-based text analysis for customer reviews'. Marketing Science **35***(6), 953–975.*

Carey, S.: 2009, The origin of concepts. *Oxford university press.*

Chandrashekar, G. and F. Sahin: 2014, 'A survey on feature selection methods'. Computers & Electrical Engineering **40***(1), 16–28.*

Chang, J., S. Gerrish, C. Wang, J. L. Boyd-Graber, and D. M. Blei: 2009, 'Reading tea leaves: How humans interpret topic models'. In: Advances in neural information processing systems. *pp. 288–296.*

Chen, T. and C. Guestrin: 2016, 'Xgboost: A scalable tree boosting system'. In: Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. *pp. 785–794.*

Chen, W., B. Gu, Q. Ye, and K. X. Zhu: 2019, 'Measuring and managing the externality of managerial responses to online customer reviews'. Information Systems Research **30***(1), 81–96.*

Choi, A. A., D. Cho, D. Yim, J. Y. Moon, and W. Oh: 2019, 'When Seeing Helps Believing: The Interactive Effects of Previews and Reviews on E-Book Purchases'. Information Systems Research **30***(4), 1164–1183.*

Clemons, E. K., G. G. Gao, and L. M. Hitt: 2006, 'When online reviews meet hyperdifferentiation: A study of the craft beer industry'. Journal of management information systems **23***(2), 149–171.*

Dhurandhar, A., V. Iyengar, R. Luss, and K. Shanmugam: 2017, 'TIP: Typifying the interpretability of procedures'. arXiv preprint arXiv:1706.02952.

Doshi-Velez, F. and B. Kim: 2017, 'Towards a rigorous science of interpretable machine learning'. arXiv preprint arXiv:1702.08608.

Feifer, J.: 2013, 'The Amazon Whisperer'.

Gantz, J. and D. Reinsel: 2011, 'Extracting value from chaos'. IDC iview **1142***(2011), 1–12.*

Gao, Q., M. Lin, and R. W. Sias: 2018, 'Words matter: The role of texts in online credit markets'. Available at SSRN 2446114.

Garg, N., L. Schiebinger, D. Jurafsky, and J. Zou: 2018, 'Word embeddings quantify 100 years of gender and ethnic stereotypes'. Proceedings of the National Academy of Sciences **115***(16), E3635–E3644.*

Garvin, D. A.: 1984, 'What Does Product Quality Really Mean?'. Sloan management review *p. 25.*

Garvin, D. A.: 1987, 'Competing on the 8 dimensions of quality'. Harvard business review **65***(6), 101–109.*

Geva, H., G. Oestreicher-Singe, and M. Saar-Tsechansky: 2019, 'Using retweets when shaping our online persona: Topic Modeling Approach'. MIS Quarterly **43***(2), 501–524.*

Gilpin, L. H., D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal: 2018, 'Explaining Explanations: An Overview of

Interpretability of Machine Learning'. In: 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA). *pp. 80–89.*

Goldberg, Y.: 2016, 'A primer on neural network models for natural language processing'. Journal of Artificial Intelligence Research **57**, *345–420.*

Goldberg, Y. and O. Levy: 2014, 'word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method'. arXiv preprint arXiv:1402.3722.

Guidotti, R., A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi: 2018, 'A survey of methods for explaining black box models'. ACM Computing Surveys (CSUR) **51***(5), 93.*

Jagarlamudi, J., H. Daumé III, and R. Udupa: 2012, 'Incorporating lexical priors into topic models'. In: Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics. *pp. 204–213.*

Kozlowski, A. C., M. Taddy, and J. A. Evans: 2018, 'The geometry of culture: Analyzing meaning through word embeddings'. arXiv preprint arXiv:1803.09288.

Lau, J. H., D. Newman, and T. Baldwin: 2014, 'Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality'. In: Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics. *pp. 530–539.*

LeCun, Y., Y. Bengio, and G. Hinton: 2015, 'Deep learning'. nature **521***(7553), 436.*

Lee, D. and K. Hosanagar: 2019, 'How do recommender systems affect sales diversity? A cross-category investigation via randomized field experiment'. Information Systems Research **30***(1), 239–259.*

Lee, D., K. Hosanagar, and H. Nair: 2018, 'Advertising Content and Consumer Engagement on Social Media: Evidence from Facebook'. Management Science.

Levy, O. and Y. Goldberg: 2014, 'Neural word embedding as implicit matrix factorization'. In: Advances in neural information processing systems. *pp. 2177–2185.*

Lipton, Z. C.: 2016, 'The mythos of model interpretability'. arXiv preprint arXiv:1606.03490.

Liu, B.: 2012, 'Sentiment analysis and opinion mining'. Synthesis lectures on human language technologies **5***(1), 1–167.*

Liu, J. and O. Toubia: 2018, 'A Semantic Approach for Estimating Consumer Content Preferences from Online Search Queries'. Marketing Science.

Liu, X., D. Lee, and K. Srinivasan: 2019, 'Large Scale Cross Category Analysis of Consumer Review Content on Sales Conversion Leveraging Deep Learning'. Journal of Marketing Research - Forthcoming.

Lu, J., D. D. Lee, T. W. Kim, and D. Danks: 2020, 'Good Explanation for Algorithmic Transparency'. In: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society. *New York, NY, USA, p. 93, Association for Computing Machinery.*

Lundberg, S. M. and S.-I. Lee: 2017, 'A unified approach to interpreting model predictions'. In: Advances in neural information processing systems. *pp. 4765–4774.*

Lysyakov, M., S. Viswanathan, and K. Zhang: 2020, 'Retailers' Content Strategies on Social Media: Insights from Analysis of Large-scale Twitter Data'.

Margolis, E. and S. Laurence: 2019, 'Concepts'. In: E. N. Zalta (ed.): The Stanford Encyclopedia of Philosophy. *Metaphysics Research Lab, Stanford University, summer 2019 edition.*

Margolis, E., S. Laurence, et al.: 1999, Concepts: core readings. *Mit Press.*

McKinsey: 2016, 'The Age of Analytics: Competing in a Data-Driven World'. Technical report, McKinsey Global Institute.

Mikolov, T., I. Sutskever, K. Chen, G. S. Corrado, and J. Dean: 2013, 'Distributed representations of words and phrases and their compositionality'. In: Advances in neural information processing systems. *pp. 3111–3119.*

Miller, T.: 2018, 'Explanation in artificial intelligence: Insights from the social sciences'. Artificial Intelligence.

Mimno, D., H. M. Wallach, E. Talley, M. Leenders, and A. McCallum: 2011, 'Optimizing semantic coherence in topic models'. In: Proceedings of the conference on empirical methods in natural language processing. *pp. 262–272.*

Mindtree: 2017, 'Integrated Customer Insights'. Technical report, Mindtree.

Moody, C. E.: 2016, 'Mixing dirichlet topic models and word embeddings to make lda2vec'. arXiv preprint arXiv:1605.02019.

Netzer, O., R. Feldman, J. Goldenberg, and M. Fresko: 2012, 'Mine your own business: Market-structure surveillance through text mining'. Marketing Science **31***(3), 521–543.*

Netzer, O., A. Lemaire, and M. Herzenstein: 2019, 'When Words Sweat: Identifying Signals for Loan Default in the Text of Loan Applications'. Journal of Marketing Research **Forthcoming**.

Newman, D., J. H. Lau, K. Grieser, and T. Baldwin: 2010, 'Automatic evaluation of topic coherence'. In: Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. *pp. 100–108.*

Pariser, E.: 2011, The filter bubble: How the new personalized web is changing what we read and how we think. *Penguin.*

Pennington, J., R. Socher, and C. 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.*

Pontiki, M., D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, A.-S. Mohammad, M. Al-Ayyoub, Y. Zhao, B. Qin, O. De Clercq, et al.: 2016, 'SemEval-2016 task 5: Aspect based sentiment analysis'. In: Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016). *pp. 19–30.*

Puranam, D., V. Narayan, and V. Kadiyali: 2017, 'The Effect of Calorie Posting Regulation on Consumer Opinion: A Flexible Latent Dirichlet Allocation Model with Informative Priors'. Marketing Science.

Ransbotham, S., D. Kiron, P. Gerbert, and M. Reeves: 2017, 'Reshaping business with artificial intelligence: Closing the gap between ambition and action'. MIT Sloan Management Review **59***(1).*

Ransbotham, S., N. H. Lurie, and H. Liu: 2019, 'Creation and consumption of mobile word of mouth: How are mobile reviews different?'. Marketing Science **38***(5), 773–792.*

Ras, G., M. van Gerven, and P. Haselager: 2018, 'Explanation methods in deep learning: Users, values, concerns and challenges'. In: Explainable and Interpretable Models in Computer Vision and Machine Learning. *Springer, pp. 19–36.*

Ribeiro, M. T., S. Singh, and C. Guestrin: 2016, '" Why should i trust you?" Explaining the predictions of any classifier'. In: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. *pp. 1135–1144.*

Rizkallah, J.: 2017, 'The Big (Unstructured) Data Problem'.

Roberts, M. E., B. M. Stewart, D. Tingley, C. Lucas, J. Leder-Luis, S. K. Gadarian, B. Albertson, and D. G. Rand: 2014, 'Structural topic models for open-ended survey responses'. American Journal of Political Science **58***(4), 1064–1082.*

Rudin, C.: 2019, 'Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead'. Nature Machine Intelligence **1***(5), 206.*

Sahlgren, M.: 2008, 'The distributional hypothesis'. Italian Journal of Disability Studies **20***, 33–53.*

Shi, B., W. Lam, S. Jameel, S. Schockaert, and K. P. Lai: 2017, 'Jointly Learning Word Embeddings and Latent Topics'. In: Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval. *pp. 375–384.*

Sunstein, C. R.: 2018, # Republic: Divided democracy in the age of social media. *Princeton University Press.*

Timoshenko, A. and J. R. Hauser: 2018, 'Identifying customer needs from user-generated content'. Forthcoming at Marketing Science.

Wernicke, S.: 2015, 'How to use data to make a hit tv show'.

Wexler, R.: 2017, 'When a computer program keeps you in jail: How computers are harming criminal justice'. New York Times.

Xun, G., Y. Li, J. Gao, and A. Zhang: 2017, 'Collaboratively improving topic discovery and word embeddings by coordinating global and local contexts'. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. *pp. 535–543.*

Yang, M., Y. Ren, and G. Adomavicius: 2019, 'Understanding user-generated content and customer engagement on Facebook business pages'. Information Systems Research **30***(3), 839–855.*

Zech, J. R., M. A. Badgeley, M. Liu, A. B. Costa, J. J. Titano, and E. K. Oermann: 2018, 'Confounding variables can degrade generalization performance of radiological deep learning models'. arXiv preprint arXiv:1807.00431.

Zhang, Q., Y. N. Wu, and S.-C. Zhu: 2017, 'Interpretable convolutional neural networks'. arXiv preprint arXiv:1710.00935 **2***(3), 5.*

Zhu, J., A. Ahmed, and E. P. Xing: 2012, 'MedLDA: maximum margin supervised topic models'. Journal of Machine Learning Research **13***(Aug), 2237–2278.*

APPENDIX FOR Focused Concept Miner (FCM): Interpretable Deep Learning for Text Exploration

A Product Categories in Data

Table 1 shows the summary statistics of the X at product, user, or journey levels.

Variable	Var-Level	Definition	Count			
Product ID	Product	Total # of products				6612
User ID	User	Total # of unique users who have read reviews				30218
Content ID	Product	Total # of reviews				87593
			Mean	**SD**	**Min**	**Max**
Price	Product	Price of the product	63.28	84.41	0.24	1049
Rating Average	Product	Avg rating of reviews available for the product	4.28	0.49	1	5
Total # Pg views	Product	Total # of page views for a particular product	141.79	178.02	2	3280
Total # Reviews	Product	Total # of reviews available for the product	66.37	117.27	1	1463
Total Wallet Size	User	Total dollar value of transaction made by a user	104.68	171.76	0	3968.79
Total # of Rev-read	User	Total # of reviews read by a user	48.73	75.22	1	2675
User-Page views	Journey	Total # of page views for a user-product pair	3.25	2.64	1	100
User-Page # of Rev-read	Journey	Total # of reviews read in a user-product pair	10.90	13.85	1	376

Table 1: Variable Summary Statistics

Figure 1: **Word Cloud of Product Categories**

B Survey Instrument Used to Measure Interpretability

For each survey, we asked 100 turkers who have previously completed at least 100 tasks with 98% or greater accuracy. We embedded a couple of attention questions and also filtered the results to prevent bots. The ordering of questions and topic presentations were randomized.

Survey 1
Given a set of words, please identify the number of distinct high-level concepts or topics described by these set of words.
For example, if the set of words are as follows,
Example Set: Game, Team, Year, Play, Good, Player, Win, Season, Fan, Hockey, Baseball
This example set of words describe one concept or topic "Sports" and # of concept is 1

Here are few more examples

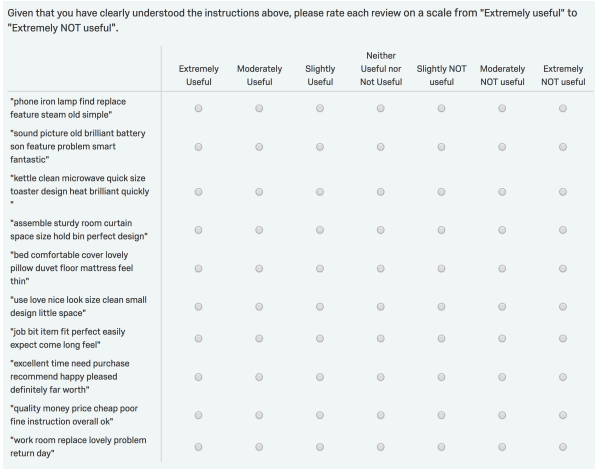
- ai algorithm machine automation robot self-driving [the topic is about artificial intelligence - # of concept =1]
- canon dslr mirrorless 48pixel fullframe fuji lens film tripod [the topic is about DSLR camera - # of concept =1]
- gate earbuds fence speaker pasta tomato keyboard [# of concepts = 3, "Gate Fence", "Earbuds Speaker Keyboard", "pasta tomato"]
- good iron printer buy bin mattress great price board fridge [# of concepts=5, "board iron", "good great buy price", "printer bin", "mattress", "fridge"]
- globe photos frank mass bear mountain cell group area york [# of concepts = 7, "Bear mountain", "cell group area", all other words go by themselves]

==========

Given that you have understood the examples above, please take a look at the following set of words and identify the number of distinct high-level concepts in each set.
Topic 1: phone iron lamp find replace feature steam old simple
Topic 2: kettle clean microwave quick size toaster design heat brilliant quickly
Topic 3: use love nice look size clean small design little space
Topic 4: quality money price cheap poor fine instruction overall ok
Topic 5: assemble sturdy room curtain space size hold bin perfect design
Topic 6: job bit item fit perfect easily expect come long feel
Topic 7: excellent time need purchase recommend happy pleased definitely far worth
Topic 8: sound picture old brilliant battery son feature problem smart fantastic
Topic 9: bed comfortable cover lovely pillow duvet floor mattress feel thin
Topic 10: work problem return replace lovely room day

Survey 2
We are interested in studying how useful certain product reviews are for making a purchase decision on an e-commerce site. Imagine that you are shopping for a particular product on Amazon.com. You are already decided on a product to purchase and are comparing several different options. To make a better decision, you decide to read customer generated reviews for more information regarding all different aspects of products.

For hypothetical 10 reviews, we provide few topic keywords about the review. Please first look at all 10 topic description of product reviews. Please rate them on a scale from **"Extremely useful"** to "**Extremely NOT useful**" on whether you would choose to read the reviews to make a purchase decision.



C Topic Words of Benchmark Topic Models Run on Main Dataset

We provide topic words for all the LDA family benchmarks run on the main e-commerce dataset. Coherence range of LDA and other LDA variants do not overlap with FCM's coherence range.

	Topic words	Avg Coherence
LDA	phone, iron, lamp, find, feature, steam, replace, old, design sound, picture, old, brilliant, battery, son, feature, problem, smart, fantastic kettle, clean, microwave, quick, size, toaster, design, heat, brilliant, quickly assemble, sturdy, room, curtain, space, size, hold, bin, perfect, design bed, comfortable, cover, lovely, pillow, duvet, floor, mattress, feel, thin	-2.25
SLDA	feature, picture, quality, need, bin, battery, time, money, purchase, problem excellent, comfortable, purchase, like, son, pillow, room, lamp, daughter, duvet money, store, steam, large, bit, break, great, size, work, long buy, sturdy, need, excellent, recommend, printer, pleased, cheap, space, instruction price, time, microwave, work, little, carpet, floor, hoover, brilliant, vacuum	-2.17
Seed LDA	curtain, item, perfect, need, happy, cheap, little, bit, comfortable, design nice, pleased, find, laptop, problem, screen, brilliant, tablet, clock, son love, purchase, water, pleased, brilliant, powerful, bit, find, old, long size, pleased, work, space, little, find, door, time, instruction, want cook, clean, cheap, happy, love, boil, cooker, want, feature, size	-1.95
STM	son, small, long, lot, want, fit, cheap, new, nice, design boil, argos, curtain, old, love, simple, print, pole, design, item toaster, lovely, love, cheap, shelf, picture, stylish, find, hold, bathroom thin, little, sleep, size, bedroom, poor, wash, worth, soft, blind dyson, come, powerful, happy, old, long, size, want, oven, item	-5.13
FCM	small, nice, space, little, use, design, clean, size, love, look money, quality, cheap, poor, instruction, price, fine, ok, overall, work come, item, easily, perfect, long, expect, fit, feel, job, bit happy, excellent, definitely, pleased, far, purchase, need, recommend, worth, time problem, work, lovely, room, day, replace, return, job, use, far	-1.86

Table 2: **Topic Words and Coherence Generated by Benchmark Topic Models on Main Dataset.**

	Concept	
1	Aesthetics	bright, clean, design, gorgeous, look, lovely, nice, pretty, simple, stylish
2	Conformance	decent, different, disappointed, excellent, expect, faulty, feature, refund, use, work
3	Feature	assemble, compact, fit, function, handy, lightweight, mount, powerful, strong, versatile
4	Brand	dyson, htc, ipad, iphone, lg, microsoft, samsung, sony, windows, xbox
5	Price (Value)	bargain, cheap, expensive, fit, inexpensive, money, price, purchase, reasonable, worst
6	Serviceability	break, durable, flimsy, poor, problem, quality, size, solid, strong, sturdy

Table 3: **Operationalized concept words for Garvin's concepts.**

D Full Results on DonorChoose Dataset

We provide full results (predictive performance and coherence interpretability) on the DonorChoose dataset here. Note that we only provide interpretable benchmarks that are viable FCM competitors. For more details see the flowchart in Main Manuscript Figure 2.

D.1 Prediction Performance - Successfully Funded as Y

	Accuracy	Precision	Recall	F1	AUC
LDA + LR	0.5026	0.5030	0.4402	0.4695	0.5005
STM + LR	0.5025	0.5032	0.3914	0.4403	0.5019
LDA + XGBoost	0.5081	0.5080	0.5164	0.5121	0.5074
STM + XGBoost	0.5071	0.5071	0.5043	0.5057	0.5127
FCM	0.5256	0.5253	0.5425	0.5338	0.5231

Table 4: **Prediction Performance on DonorsChoose** $Y = SuccessfullyFunded$

D.2 Topic Words and Coherence - Successfully Funded as Y

	Topic words	Coherence
LDA	owl, marching, drum, sharpener, composer, yearbook, gum, biome, sound, peacemaker tablet, kindle, choir, educational, scholar, music, violin, elmo, article, camera motor, ball, seating, fine, loom, animal, cell, coordination, chromebook, tricycle yoga, americans, chess, bee, garden, meditation, hot, costume, globe, nutrient calculator, robot, ap, makerspace, robotic, english, esol, overcome, advanced, steam	-1.81
STM	drum, guitar, choir, ensemble, percussion, ukulele, fitness, volleyball, instrument, tennis subtraction, erase, multiplication, ipad, app, sensory, letter, ipads, disability, instruction fiction, novel, reader, book, nonfiction, aloud, reread, read, library, text camera, edit, artwork, photography, pastel, editing, watercolor, portfolio, printer, film plant, microscope, soil, ecosystem, dissection, organism, dna, specimen, sensor, forensic	-1.70
FCM	camera, history, photography, animal, world, trip, experience, picture, video, life art, material, child, supply, color, easel, chair, time, pencil, center music, play, drum, instrument, equipment, song, sensory, fitness, physical, keyboard book, read, reading, novel, text, library, reader, level, love, language technology, math, computer, ipad, lab, project, able, science, allow, laptop	-1.65

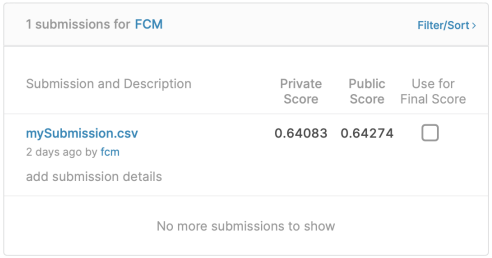
Table 5: **Topic/Concept Words on DonorsChoose** $Y = SuccessfullyFunded$

D.3 Prediction Performance On Kaggle Leaderboard- Exciting as Y (From Kaggle Competition Dataset)

Kaggle, an online platform for data science, held the KDD Cup in 2014 for predicting the excitement (custom Y as discussed in Section 5.5.2) of projects at DonorsChoose.org. Although the formal submission for competition has been closed, Kaggle still accepts late submissions and calculates the AUC score of each submission.

To better illustrate the prediction performance of FCM, we uploaded our predicted results using FCM to Kaggle. Since the competition itself doesn't require model interpretability, for the purpose of maximizing the predictive power of FCM, we replaced the last sigmoid layer in FCM with XGBoost.

Figure 2 shows the screenshots of our model result. With integration of XGBoost, FCM model gets an AUC score of 0.64083 for our first submission. Among 472 teams on the final leaderboard, our FCM model beats 97% of all the submissions and is only narrowly beaten by the 14th-place



(a) **FCM Record**

#	△pub	Team Name	Score	Entries
1	▲1	'STRAYA	0.67813	213
2	▼1	DataRobot	0.67319	220
3	▲22	ChaoticExperiments (KIR...	0.67297	69
4	▲10	dkay & bmax & James King	0.66472	239
5	▲6	Triskelion,Yan, KazAnova...	0.65948	225
6	▲3	Giulio, orchid, Luca & Ben	0.65918	264
7	▲5	:-)	0.65372	123
8	▲30	柳景明	0.65366	53
9	▼2	ミ^・.・^彡SPN	0.64896	296
10	▼5	Abhishek & Silogram	0.64878	143
11	▼1	FAndy	0.64720	78
12	▼4	Delicious Food	0.64455	306
13	▲14	Yun G	0.64365	103
14	▼8	Pacific Rim	0.64343	162
15	—	Owen	0.63995	52

(b) **Leaderboard**

Figure 2: The private leaderboard is calculated with approximately 75% of the test data and reflects the final standings of each team.

team. We could further feature engineer to achieve higher predictive power but that is *not* the point of FCM.

As per the rule of the competition, the top three teams on the leaderboard are required to publish their model to the platform.[1] This allows us to have a deeper comparison between the architectures of FCM and the leading prediction models.

The top three models on the leaderboard rely on the textual features handcrafted from methods including part-of-speech and tf-idf. Since the competition does not require interpretability of the prediction, all top ranking algorithms are blackbox algorithms. Most are heavily engineered by assembling a collection of highly nonlinear learning algorithms.

D.4 Predictive Performance & Topic Words and Coherence - Exciting as Y (From Kaggle Competition Dataset)

Note that since Kaggle **does not release ground-truth labels of the test set, we produce the benchmark results in Table 6 using the hold-out set sampled from the provided data.** As a result, there is a difference between the AUC score of FCM in Table 6 vs. the one computed by Kaggle in Figure 2.

[1] See `https://www.kaggle.com/c/kdd-cup-2014-predicting-excitement-at-donors-choose/overview/winners` for the code submitted by the top three teams.

	Accuracy	Precision	Recall	F1	AUC
LDA + LR	0.8765	0.3125	0.0045	0.0088	0.6181
STM + LR	0.8765	0.3125	0.0045	0.0088	0.6325
LDA + XGBoost	0.8769	0.2000	0.0009	0.0018	0.6975
STM + XGBoost	0.8770	0.2500	0.0009	0.0018	0.6990
FCM	0.9142	0.3958	0.0219	0.0415	0.6615
FCM + XGBoost	0.9150	0.4000	0.0046	0.0091	0.7109

Table 6: **Prediction Performance on DonorsChoose** $Y = Exciting$ **(results on hold-out set)**

	Topic words	Coherence
LDA	level, technology, reader, story, language, home, center, group, library, english game, writing, problem, hand, concept, board, parent, free, pencil, paper science, computer, ipad, explore, live, develop, look, give, change, projector instrument, lab, middle, state, diverse, real, large, interest, background, focus money, budget, digital, lot, cut, club, fund, population, team, design	-2.02
STM	ipad, tablet, app, computer, ipads, projector, technology, laptop, camera, internet instrument, band, drum, guitar, musical, bully, bullying, fitness, ball, music plant, garden, microscope, clay, butterfly, dissection, organism, painting, scientist, habitat erase, pencil, manipulative, sharpener, letter, calculator, math, marker, easel, alphabet reader, book, novel, read, fiction, nonfiction, listening, library, literature, reading	-1.98
FCM	life, opportunity, experience, world, garden, society, culture, grow, hope camera, ipad, technology, video, class, projector, able, use, tablet, allow drum, math, calculator, science, music, hand, instrument, game, play, teach art, ball, provide, new, material, child, writing, like, supply book, read, reading, center, love, time, remember, level, listen, player	-1.77

Table 7: **Topic/Concept Words on DonorsChoose** $Y = Exciting$

E Review Examples for Case Study and Visualization

We present review examples for case study and visualization. The last sigmoid layer ($\sigma(.)$) of FCM enables us to measure the correlation of documents on the business outcome. To do this, we first calculate the predicted conversion rate $\hat{Conversion}$ using the true document-concept distribution $DocConceptD$. Then, holding explanatory variables the same, we compare the predicted conversion rate using the average document-concept distribution $\overline{DocConceptD}$, which is calculated by averaging the document-concept across all the available documents in the training set.

$$\Delta Conversion = \sigma(\theta_0 + \sum_i \theta_i DocConceptD_i + \sum_j \theta_j ExpVar_j) - \sigma(\theta_0 + \sum_i \theta_i \overline{DocConceptD}_i + \sum_j \theta_j ExpVar_j) \quad (1)$$

The differences in the two predicted conversion rates are calculated across all documents in the dataset to gauge the correlation of document on conversion.

Table 8 shows six reviews along with the product information, predicted conversion correlation, and concept distribution assessed by FCM. For diversity of examples, we randomly choose two reviews with the most positive and the most negative correlation, and two neutral reviews. For each phrase that matches a dimension of concept, we manually apply a tag that indicates the next corresponding concept. Comparing the tagged reviews and the estimated concept distribution shows that the concept distributions assessed by FCM are generally well aligned with the actual semantics of the reviews.

Product	Title	Review with Manually Tagged Concepts	Predicted Conversion Correlation	Assessed Concept Distribution
Single Glass Door Display Cabinet - Black	Useful storage	I needed a replacement for an old corner unit which used to house small momentoes we collect from our travels, both in the UK and abroad.This little unit is perfect. It has7 glass shelves and of course the floor of the cabinet to display items <FEATURE>. I needed something with more shelving rather than a unit with 3 larger spaced shelves.Fits nicely <CONFORMANCE> in the same corner and has a light in the top.My husband put it together fairly easily <FEATURE>, with my son helping when putting glass in place <CONFORMANCE>. Atlhough good value, it is lightweight and the glass is fairly thin<AESTHETICS>. Comes with fixing strap to hold against the wall <FEATURE> if required.Quick delivery.	99.34%	Aesthetics: 10.27% Conformance: 34.01% Features: 35.18% Value: 10.27% Serviceability: 10.27%
Hoover Turbo Power Bagless Upright Vacuum Cleaner	replacement hoover	Easy to assemble <FEATURE> and light weight<AESTHETIC>. The extension <FEATURE> hose for the stairs is a great. That's the good points. The suction <FEATURE> is not that great, especially if you have pet hair to remove. Difficult to see <AESTHETIC> through the cylinder to see if it needs emptying. On the 3rd time I used it the belt snapped. I returned it to Argos and got a full refund <SERVICEABILITY>.	˜ 0	Aesthetics: 10.18% Conformance: 6.13% Features: 43.76% Value: 18.88% Serviceability: 21.04%
Challenge Xtreme 70 Piece Titanium Drill Driver Set	good variety but breaks quickly	We bought the drill driver set to utilize <FEATURE> some of the parts for building <FEATURE> flat pack furniture as well as outdoor decking <FEATURE> . The variety and amount of bits is great <CONFORMANCE> but unfortunately the pieces break <SERVICEABILITY> very quickly and easily. The screwdriver heads wear out <AESTHETICS> rapidly and the drill bits break even when drilling into soft woods.	˜ 0	Aesthetics: 12.95% Conformance:12.95% Features: 48.19% Value: 12.95% Serviceability: 12.95%
2m Heavy-weight PVC Curtain Track - White	Has a drawback	I bought this to hang curtains with 2 sets of linings fitted (thermal and blackout) when these linings are fitted <CONFORMANCE> you have to use the top row of the tape on the curtains to hang them. This makes the curtain sit low <AESTHETICS> on the rail and causes <SERVICEABILITY> a gap between the curtain and rail which allows light in. I got round this by using a moulded skirting board fitted to the rail a bit like a pelmet, it works for me. They rail itself is really easy <FEATURE> to adjust to the correct size and to fit .	-79.48%	Aesthetics: 20.92% Conformance: 28.17% Features: 24.46% Value: 10.92% Serviceability: 15.53%
Eve 3 Light Ceiling Fitting - Clear	lovely light but...	I bought 2 eve lights for my narrow hall and was pleased <CONFORMANCE> with them so much I bought another 2 for my living room. However, I am so disappointed <SERVICEABILITY> that although the sun ray effects on the ceiling is lovely <AESTHETIC> -the rest of the ceiling is very dark(room size 12ftx15ft) They also cast <FEATURES> massive gloomy shadows on the walls which are driving me mad and I am going to replace <SERVICEABILITY> them. In themselves - the lights are lovely and a bargain <VALUE> but they are only good enough for narrow spaces like landings and halls.	-75.25%	Aesthetics: 12.21% Conformance:29.50% Features: 17.99% Value: 17.67% Serviceability: 22.63%

Table 8: **Example Reviews and Concept Tagged by FCM.**

F FCM Training Procedure Details

Overview of Training Procedure The following is a description of the FCM training algorithm corresponding to Figure 4. Training consists of 2 stages: a preprocessing stage that constructs the training samples from the raw document corpus, and the optimization stage that learns the model parameters by minimizing the loss function derived in Section 3. We detail each of these stages below.

Stage I: Construct Training Samples

The following procedure is applied to extract pivot-context word pairs from each document d in the corpus that are labeled y. Let k be the desired window-size (a tunable hyperparameter).

For each word u in the document d:

1. Collect the words $v \in C_k(u)$ in a window of size k centered at u. Note that $|C_k(u)| = k$ always, since if u lies at the extreme ends of the document, we extend the edges of the window to include a total of k words.

2. Construct the training sample as the tuple: $(d, u, v_1, \ldots, v_k, y)$ where $C_k(u) = \{v_1, \ldots, v_k\}$.

This procedure, when run on all documents and all words therein, gives us N training samples. These training samples are then provided to the optimization stage detailed below.

Stage II: Optimize Loss Function

The following procedure operates by iterating over each training sample $(d, u, v_1, \ldots, v_k, y)$, computing the value of the loss function derived in Section 3, computing the gradient of this loss function, and adjusting the values of the model parameters in the direction of this gradient. The procedure itself is repeated several times (each of which is called an epoch), and the overall loss is inspected to ensure it decreases with each epoch.

For each training sample $(d, u, v_1, \ldots, v_k, y)$:

1. Using the current concept embeddings $\mathbf{E_t}$ and the document-concept probabilities $\mathbf{p_d}$ from the **CAN** network, construct the document vector $\mathbf{v_d}$ (see eq. 7 and 8).

2. Retrieve m negative samples (words) $w_1, \ldots, w_m$ uniformly at random from the document corpus.

3. Using the current word embeddings $\mathbf{E_w}$, obtain the embeddings of the words $u, v, v_1, \ldots, v_k$ and $w_1, \ldots, w_m$, compute the document-specific word embedding $\mathbf{v_{dw}}$ and compute the skip-gram negative sampling loss $\mathcal{L}_{\text{neg}}$ (see eq. 9).

4. Using the document-concept probabilities $\mathbf{p_d}$, compute the Dirichlet loss $\mathcal{L}_{\text{dir}}$ (see eq. 13).

5. Using the current concept embeddings $\mathbf{E_t}$, compute the diversity loss $\mathcal{L}_{\text{div}}$ (see eq. 14).

6. Using the current prediction weights θ, the document-concept probabilities $\mathbf{p_d}$ and the document label y, compute the prediction loss $\mathcal{L}_{\text{clf}}$ (see eq. 16).

7. Combine the losses as in eq. 17 to construct the overall loss, compute its gradient via automatic differentiation, and update the model parameters $\mathbf{E_w}, \mathbf{E_t}, \theta$ and the **CAN** network in the direction of this gradient with the Adam (Kingma and Ba, 2014) adaptive momentum-based optimization method.

Making Predictions

Predictions for an unseen document d' are made by first obtaining its document-concept probabilities $\mathbf{p'_d}$ using the trained **CAN** network, and then computing the prediction $\hat{y}_{d'} = \theta \cdot \mathbf{p'_d}$ using the trained prediction weights θ.

Implementation Choice Specifics We construct train, validation, and test sets using 70%, 15%, and 15% of the full data, respectively. To improve generalizability, we regularize $\mathbf{W}$ and θ with their L_2 norm, and perform dropout on $\mathbf{E}$ and gradient clipping to prevent exploding gradients. We initialize our algorithm with pre-trained word2vec word-vectors, trained in an unsupervised fashion on a corpus of 100 billion words from Google News. We train the model using mini-batch stochastic gradient descent with a batch-size of 10,240 on an Nvidia Titan X Pascal with 12GB of GPU memory. The estimation roughly took 2 hours on this hardware specification to get to 200 epochs.

G Limitations and Future Extensions

We share several short idea overviews for extending the basic FCM models and post-processing FCM outputs for future papers. The scope of difficulties range from feasible to very uncertain. Specifically, four extension ideas are shared in order of increasing difficulty: Automatic Naming of Concepts, Adding Valence to Concepts, Zooming into Concepts, and finally FCM and Causality.

Automatic Naming of Concepts In demonstrating FCM with review data, we have manually interpreted and titled each extracted concept. Titling concepts can be further automated post-FCM for a faster and more "objective" presentation of the results. One method involves adopting existing techniques from topic modeling literature (e.g., Lau et al. (2011)), which reduces down to generating label candidate sets by tapping into external text—such as Wikipedia—or the corpus itself and ranking the label set by its similarity to topics. However, given that FCM architecture includes semantic-spatial relationship-aware word embedding at its basis, geometric methods within the concept (and word) vector space may be more appropriate and powerful.

Adding Valence to Concepts The current FCM only deals with the volume of concepts present in text. Valence (positive or negative sentiment) could be added to enhance the model and interpretation. Within the model pipeline, before the last softmax layer, the document-concept vector could be further processed. However, it is unclear how to modify the end-to-end architecture to inject valence of the document-concept vector without the need for training datasets or breaking the end-to-end framework.

Zooming into Concepts Model architecture could be extended to be hierarchical in concept representation relations. Given the hierarchical nature of discovered concepts, the model could also include a lever to zoom in or out on concept hierarchy for different levels of abstraction. One potential way might be to tap into existing semantic networks and knowledge graph databases that are aware of concept hierarchy, such as ConceptNet (http://conceptnet.io/) or WordNet (https://wordnet.princeton.edu/).

FCM and Causality While FCM is envisioned and presented as an exploratory tool (and *not causal*) with many caveats, some users may still want to extend it for causal uses. Using deep learning for causal inference is still a nascent field with only a handful of papers (e.g., Hartford et al. (2016), Louizos et al. 2017, Kallus 2018, etc.) and is theoretically undeveloped due to the fact that there are no theoretical asymptotic results on generic deep learning models, which makes it difficult to draw robust inferences.

To speculatively suggest, as the last layer of the FCM architecture resembles classical logistic regression, perhaps it can be extended to inject characteristics of extant causal models. We are unsure where to begin, however. For now, a quick and robust way to utilize FCM is to simply use it as a representation learning algorithm (Bengio et al., 2013) to extract non-trivial representation of input text data by ablating the FCM and using the inner-layer data representations. This could be anything from simple document-concept vectors to a complicated nonlinear combination of document-elements. Given that document-concept vectors are the most interpretable, we suggest starting with these vectors as X inputs to other traditional causal techniques.

Subject	Newsgroups
Computers (Comp)	*comp.graphics*, comp.os.ms-windows.misc, comp.sys.ibm.pc.hardware comp.sys.mac.hardware, comp.windows.x
Recreation (Rec)	*rec.autos*, rec.motorcycles, rec.sport.baseball, rec.sport.hockey
Science (Sci)	*sci.med*, sci.crypt, sci.electronics, sci.space
Politics (Pol)	*talk.politics.mideast*, talk.politics.guns, talk.politics.misc
Religion (Rel)	*talk.religion.misc*, alt.atheism, soc.religion.christian

Table 9: Newsgroups in the 20newsgroups dataset grouped by category. Selected representative dataset is in italics.

	ROC-AUC		F1-Score		Average Precision	
Dataset	FCM	XGB	FCM	XGB	FCM	XGB
Comp + Rec	**0.966**	0.915	**0.903**	0.810	**0.952**	0.922
Comp + Sci	**0.965**	0.937	**0.891**	0.852	**0.961**	0.937
Comp + Pol	**0.991**	0.948	**0.956**	0.838	**0.989**	0.946
Comp + Rel	**0.972**	0.891	**0.904**	0.780	**0.959**	0.899
Rec + Sci	**0.944**	0.866	**0.878**	0.796	**0.945**	0.853
Rec + Pol	**0.983**	0.944	**0.932**	0.868	**0.982**	0.933
Rec + Rel	**0.952**	0.908	**0.841**	0.809	**0.941**	0.913
Sci + Pol	**0.978**	0.932	**0.925**	0.819	**0.975**	0.934
Sci + Rel	0.841	**0.958**	0.684	**0.878**	0.809	**0.961**
Pol + Rel	0.942	**0.970**	0.833	**0.892**	0.920	**0.973**

Table 10: (Classification Metrics) Area under the ROC curve (AUC), average precision (AP) and F1-score (F1) for each dataset and method. 1.000 is the best score for all metrics. Best method for each dataset is in bold.

H 20-Newsgroup Data Performance

To demonstrate the predictive performance of FCM on a publicly available dataset, we train and evaluate FCM on the 20newsgroups dataset. The 20newsgroups dataset consists of 20 collections of documents, each of which contains 1,000 emails from a single newsgroup. Each newsgroup is associated with some topic (such as science, politics, computer graphics, etc.), which is also used as the label for all the documents within the newsgroup. The newsgroups may be broadly categorized as in Table 9. We evaluate FCM on the binary classification task of distinguishing between emails from a pair of different newsgroups. Instead of evaluating on every pair of newsgroups (which is a total of 190 pairs), we select a single newsgroup from each of the 5 broad categories in Table 9 (selected newsgroups emphasized in italics), and evaluate on all 10 pairs derived from the selected newsgroups. For comparison, we use the XGBoost classifier. We tuned the XGBoost hyperparameters to perform the best on the test data, setting the maximum depth to 1 and η to 0.1 for 1,000 training iterations. For both FCM and the XGBoost, we report area-under-the-curve classification metrics (ROC-AUC and average-precision), as well as thresholded classification metrics (precision, recall, F1-score and accuracy) in Tables 10 and 11. We find that FCM performs on-par with or better than XGBoost for a majority of the newsgroup pairs on all metrics.

	Accuracy		Precision		Recall	
Dataset	FCM	XGB	FCM	XGB	FCM	XGB
Comp + Rec	**0.903**	0.826	0.885	**0.886**	**0.922**	0.746
Comp + Sci	**0.892**	0.856	**0.892**	0.885	**0.890**	0.822
Comp + Pol	**0.957**	0.849	**0.956**	0.892	**0.956**	0.790
Comp + Rel	**0.927**	0.800	**0.926**	0.865	**0.883**	0.711
Rec + Sci	**0.876**	0.787	**0.879**	0.770	**0.877**	0.825
Rec + Pol	**0.932**	0.865	**0.936**	0.839	**0.928**	0.899
Rec + Rel	**0.878**	0.820	**0.881**	0.869	**0.804**	0.756
Sci + Pol	**0.927**	0.832	**0.933**	0.864	**0.997**	0.779
Sci + Rel	0.800	**0.884**	0.893	**0.916**	**0.918**	0.844
Pol + Rel	0.869	**0.895**	0.845	**0.936**	0.821	**0.851**

Table 11: (Classification Metrics) Accuracy, precision, and recall for each dataset and method with the prediction threshold fixed at 0.5. The threshold is not tuned for any metric. The best score for all metrics is 1.000. The best method for each dataset and metric is in bold.

References

Bengio, Y., A. Courville, and P. Vincent: 2013, 'Representation learning: A review and new perspectives'. *IEEE transactions on pattern analysis and machine intelligence* **35**(8), 1798–1828.

Hartford, J., G. Lewis, K. Leyton-Brown, and M. Taddy: 2016, 'Counterfactual Prediction with Deep Instrumental Variables Networks'. *arXiv preprint arXiv:1612.09596*.

Kallus, N.: 2018, 'DeepMatch: Balancing Deep Covariate Representations for Causal Inference Using Adversarial Training'. *arXiv preprint arXiv:1802.05664*.

Kingma, D. P. and J. Ba: 2014, 'Adam: A method for stochastic optimization'. *arXiv preprint arXiv:1412.6980*.

Lau, J. H., K. Grieser, D. Newman, and T. Baldwin: 2011, 'Automatic labelling of topic models'. In: *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. pp. 1536–1545.

Louizos, C., U. Shalit, J. M. Mooij, D. Sontag, R. Zemel, and M. Welling: 2017, 'Causal effect inference with deep latent-variable models'. In: *Advances in Neural Information Processing Systems*. pp. 6446–6456.