Yale University
EliScholar – A Digital Platform for Scholarly Publishing at Yale

Cowles Foundation Discussion Papers | Cowles Foundation

5-19-2019

Attribute Sentiment Scoring with Online Text Reviews: Accounting for Language Structure and Missing Attributes

Ishita Chakraborty
Yale University

Minkyung Kim
Yale University

K. Sudhir
Yale University

Follow this and additional works at: https://elischolar.library.yale.edu/cowles-discussion-paper-series

 Part of the Economics Commons

Recommended Citation

Chakraborty, Ishita; Kim, Minkyung; and Sudhir, K., "Attribute Sentiment Scoring with Online Text Reviews: Accounting for Language Structure and Missing Attributes" (2019). *Cowles Foundation Discussion Papers*. 2592.
https://elischolar.library.yale.edu/cowles-discussion-paper-series/2592

This Discussion Paper is brought to you for free and open access by the Cowles Foundation at EliScholar – A Digital Platform for Scholarly Publishing at Yale. It has been accepted for inclusion in Cowles Foundation Discussion Papers by an authorized administrator of EliScholar – A Digital Platform for Scholarly Publishing at Yale. For more information, please contact elischolar@yale.edu.

ATTRIBUTE SENTIMENT SCORING WITH ONLINE TEXT REVIEWS: ACCOUNTING FOR LANGUAGE STRUCTURE AND MISSING ATTRIBUTES

By

Ishita Chakraborty, Minkyung Kim, and K. Sudhir

May 2019

Revised September 2020

COWLES FOUNDATION DISCUSSION PAPER NO. 2176R



COWLES FOUNDATION FOR RESEARCH IN ECONOMICS
YALE UNIVERSITY
Box 208281
New Haven, Connecticut 06520-8281

http://cowles.yale.edu/

Attribute Sentiment Scoring with Online Text Reviews: Accounting for Language Structure and Missing Attributes

Ishita Chakraborty, Minkyung Kim, K. Sudhir

Yale School of Management

September 2020

We thank the participants in the marketing seminars at Duke, HKUST, ISB, Kellogg, Penn State, NITIE, UBC, UNC Charlotte, University of Texas, UW Marketing Camp, the Yale SOM Lunch, the 2019 ISMS Doctoral Consortium, the 2018 CMU-Temple Conference on Big Data and Machine Learning, the 2019 SICS, the 2019 Management Science Workshop at University of Chile, 2019 IMRC Conference in Houston teh 2019 SAS AI Symposium.

Electronic copy available at: https://ssrn.com/abstract=3395012

Attribute Sentiment Scoring with Online Text Reviews: Accounting for Language Structure and Missing Attributes

The authors address two significant challenges in using online text reviews to obtain fine-grained attribute level sentiment ratings. First, they develop a deep learning convolutional-LSTM hybrid model to account for language structure, in contrast to methods that rely on word frequency. The convolutional layer accounts for the spatial structure (adjacent word groups or phrases) and LSTM accounts for the sequential structure of language (sentiment distributed and modified across non-adjacent phrases). Second, they address the problem of missing attributes in text in constructing attribute sentiment scores—as reviewers write only about a subset of attributes and remain silent on others. They develop a model-based imputation strategy using a structural model of heterogeneous rating behavior. Using Yelp restaurant review data, they show superior accuracy in converting text to numerical attribute sentiment scores with their model. The structural model finds three reviewer segments with different motivations: status seeking, altruism/want voice, and need to vent/praise. Interestingly, our results show that reviewers write to inform and vent/praise, but not based on attribute importance. Our heterogeneous model-based imputation performs better than other common imputations; and importantly leads to managerially significant corrections in restaurant attribute ratings.

Keywords: text mining, natural language processing (NLP), convolutional neural networks (CNN), long-short term memory (LSTM) Networks, deep learning, lexicons, endogeneity, self-selection, online reviews, online ratings, customer satisfaction

Electronic copy available at: https://ssrn.com/abstract=3395012

INTRODUCTION

Many firms conduct routine tracking surveys on product/service performance on selected attributes chosen by managers that they believe drive overall customer satisfaction (Mittal et al. 1999, Mittal et al. 2001). The summary scores from these surveys are used as dashboard metrics of overall satisfaction and attribute performance by managers. In many industries offering "experience goods", such as restaurants, hotels and (even) nursing homes, crowd-sourced online review platforms have emerged as an alternative and less expensive source of scalable, real-time feedback for businesses to *listen in* on their markets for both performance tracking as well as competitive benchmarking (e.g., Xu 2019, Li et al. 2019). Even when not used as a replacement for tracking surveys of performance, such quantitative summary metrics are valuable for managers because consumers use review platforms when making choices (e.g., Zhu and Zhang 2010, Luca and Vats 2013).

This paper develops a scalable text analysis method by which online review platforms that only collect open-ended text reviews can produce attribute level summary ratings[1] similar to those who use quantitative attribute level surveys. This involves solving two novel and challenging subproblems. First, it requires developing a text mining framework that can convert the rich texture of attribute level sentiment expressed in the text to a fine-grained quantitative rating scale, that not only captures the *valence* of the sentiment, but also the *degree* of positivity or negativity in sentiment. The second problem is that since reviewers self-select which attributes to write about in open-ended text, many attributes will be missing in unprompted reviews. The challenge is to correctly interpret "silence," when a reviewer does not mention an attribute in the review text and impute the correct sentiment to obtain the aggregate attribute level rating. Our results show that the magnitude of corrections can be large enough to be managerially significant.[2] Further, behavioral research has long recognized the importance of the right imputation for missing values

[1] Some review platforms such as Zagat, OpenTable and TripAdvisor ask for numerical attribute ratings from reviewers before open-ended text. This may obviate the need to convert text to numerical attribute sentiment scores; but a key disadvantage is that attribute level questions vastly reduce response rates and quality because of the additional time and cognitive costs on the reviewers (Krosnick 1991, Huang et al. 2015). Therefore many large review platforms such as Yelp, Google and Facebook only obtain an overall rating and free-flowing, open-ended text feedback. Our approach can provide attribute level ratings on such platforms.

[2] Luca (2016) finds that a 1 point change in restaurant ratings leads to a 5-9% change in revenues.

Electronic copy available at: https://ssrn.com/abstract=3395012

because people do not ignore missing attributes and often make complex and imperfect inferences from missing data in evaluations. For example, Slovic and MacPhillamy (1974) and Peloza et al. (2015) discuss some common types of wrong inferences—higher weights on common attributes (i.e. attributes for which information is available for all options) or simply proxy missing attribute score with some unrelated attribute score (extra-attribute mis-estimation). Gurney and Loewenstein (2019) provides an excellent review of this topic. While the nature of these inferences may vary, the general takeaway is that missingness usually worsens choice and decision making. This justifies our interest in obtaining corrected attribute ratings.[3] We next describe the key challenges involved in tackling these two problems and how we address them.

Challenges in Attribute Level Sentiment Scoring from Text

Attribute level sentiment scoring from text involves connecting a specific product attribute (e.g., food, service) to an associated satisfaction rating. With fine-grained sentiment scoring, we need to convert text to more than just valence (positive, negative, neutral), but also represent the degree of positivity and negativity (in say a 1-5 point scale). While there has been some work on sentiment scoring of attribute valence (e.g., Archak et al. 2011), there has been little work on fine-grained attribute scoring—the focus of our paper. We now describe the challenges involved relative to extant work in the literature. We note that the computer science literature in fine-grained sentiment scoring is still evolving and it remains an open problem in natural language processing (Schouten and Frasincar 2015).

Over the last decade, marketing scholars have extensively used text analysis to identify topics, customer needs and mentions of product attributes. These papers typically have used "bag-of-words" approaches such as LDA and lexicons—where the identification of attributes and sentiments is based on the frequency of sentiment words. LDA applications include Tirunillai and

[3]To assess its value in our specific context, we show using an mTurk experiment (see Online Appendix Table A1), that consumer choices are more consistent with their true preferences when attribute level ratings are available. Managers also clearly would prefer their ratings to be valid—to the extent imputations help obtain valid estimates, they would clearly prefer it. We show that our imputations indeed work better than other common imputations on a holdout sample, and that the corrections are large enough to be managerially significant.

Electronic copy available at: https://ssrn.com/abstract=3395012

Tellis (2014), Hollenbeck (2018), Puranam et al. (2017) and Büschken and Allenby (2016). Archak et al. (2011) use a lexicon method to identify attributes and sentiment valence; but do not address fine-grained sentiment scoring.[4] But bag-of-words based approaches are limited in their ability to adequately score attribute sentiments. Consider the following examples where *sentiment degree* is modified, as in (i) *"horrible," "not horrible," "not that horrible"* and (ii) *"delight, "just missed being a delight"*. When words are just counted as in bag-of-words, making the connections between the key sentiment words "horrible" and "delight" with their degree modifiers will be difficult, without considering how they are grouped adjacently to form phrases—i.e., spatial structure.

More generally, in NLP, certain types of sentences are considered "hard" for sentiment scoring (Socher et al. 2013). Table 1 provides a typology of such "hard" sentences with examples. Like the examples above which modified sentiment degree, *negations* often require accounting for adjacent words, i.e., spatial structure to correctly interpret both valence and sentiment degree. Further, other types of sentences such as *long and scattered sentences* and *contrastive conjunctions* require accounting for both the spatial the sequential structure of language, as the sentiment is distributed and modified across *non-adjacent words* in a sentence. When there are long sentences with sentiments scattered across attributes, being able to make the right association of the sentiment with the attribute becomes a challenge; further sentiments get modified along different parts of a long sentence, and therefore one has to consider these sequences together in inferring sentiment. Contrastive conjunctions–words/phrases like "but," "despite," and "inspite of" can reverse the sentiment of a sentence—on either side of the conjunction. Implied sentiments are challenging because the meaning/sentiment associated with a word lies within a richer context of its usage.

[Insert Table 1 here]

These examples motivate the need to go beyond frequency-based "bag-of-words" approaches and model the structure of language (in terms of phrases and sequences). In our deep learning

[4]Timoshenko and Hauser (2018) and Liu et al. (2019) use deep learning models that are not based on word frequencies, but their focus is on attribute and valence identification respectively and hence do not need to account for language structure issues that need to be addressed in fine-grained attribute sentiment scoring.

Electronic copy available at: https://ssrn.com/abstract=3395012

model, a convolutional layer captures the spatial structure (grouping of adjacent words), and a Long Short Term Memory (LSTM) layer captures the sequential structure (sequence of adjacent and non-adjacent phrases). This allows us to improve our sentiment classification not only in the aggregate on "easy" sentences, but also on the "hard" sentences.

Accounting for Attribute Silence in Attribute Sentiment Scoring

As described earlier, the current literature on topic identification focuses on the frequency of mentions across reviews (e.g., Büschken and Allenby 2016) to identify the most common or novel needs/benefits, attributes desired by consumers/user. The implicit assumption is that topics or attributes that are not mentioned are not important and can be ignored.

We question the premise that *importance* is the primary reason for why an attribute is mentioned or not. There can be other reasons for why a reviewer is silent on an attribute. Some may write only if it can *influence* or be *informative* to readers. For example, if there is high variance among current raters, one's rating can be influential and informative. Or if one's own rating is different from the consensus based on current reviews, one may be motivated to write a different point of view. There could of course be asymmetry in this motivation depending on whether the deviation from consensus is positive or negative. Finally, some raters may choose not to write when the product meets expectations (and rating would have been a three), but only to praise/vent when they are very satisfied or dissatisfied.

We develop a model-based strategy that imputes *missing* sentiment based on observable restaurant characteristics and observable/unobservable reviewer characteristics. We consider and exploit three key features of the available data in this context in developing and identifying the structural model: (1) the same restaurant is visited and experienced by multiple reviewers; given that a restaurant provides similar services to all patrons, we assume that all reviewers receive a common latent utility plus idiosyncratic shocks. (2) the same reviewer visits multiple restaurants, this allows us to identify observable reviewer heterogeneity and unobserved heterogeneity in rating styles—i.e. how they map experienced utility to attribute level ratings. (3) all reviewers provide an

Electronic copy available at: https://ssrn.com/abstract=3395012

overall rating, so given multiple observations from a reviewer, we can infer heterogeneous weights of attributes on overall ratings.

We allow the structural model of rating behavior to account for heterogeneity in rating styles and weights on attributes driving overall ratings. Specifically, we allow for a nonlinear and heterogeneous mapping from experienced utility to attribute ratings using an ordinal logit and a heterogeneous weighting of different attributes to explain the observed overall rating as a regression. The heterogeneity is modeled within a latent class framework. We estimate the model using an EM algorithm, where the missing data on attribute ratings are imputed based on the model parameters during an iteration and iterated till the parameters converge.

Since the structural model provides insights on reviewer segments and their behavior, it not only helps with imputation but also enables us to assess the above conjectured "drivers of silence" in reviews. We find that there are multiple reviewer segments with different motivations to write reviews—one segment seeks status, another seeks to vent/praise and a third is altruistic or wants to voice their opinion. Interestingly, we find that informativeness and need to vent/praise drives what attributes are mentioned; not attribute importance. We then validate the imputations from our structural model by showing superior performance relative to simpler homogeneous models and other ad-hoc imputation rules on holdout data. Finally, we demonstrate that corrections for attribute silence based on observable and unobservable heterogeneity leads to significant corrections in average attribute ratings for a business.

We note that our problem definition for attribute level ratings abstracts away from issues of (1) selection in *who* chooses to review (e.g., Li and Hitt 2008) and (2) strategic review shading by reviewers and/or fake reviews (e.g., Mayzlin et al. 2014, Luca and Zervas 2016) when aggregating ratings. Reviewer selection/review shading issues are relevant not just for attribute level ratings, but also for overall ratings; as such any approaches to address these issues for overall ratings should also be applicable for attribute level ratings.

Summarizing, our key contributions are as follows: The paper is the first to do *fine-grained* attribute sentiment scoring using text reviews in marketing; i.e., we not only capture attribute sen-

Electronic copy available at: https://ssrn.com/abstract=3395012

timent valence, but also the *degree* of positivity or negativity in sentiment. For this, we highlight the need to move beyond word frequency based approaches (lexicon and LDA) to a deep learning approach that accounts for language structure. Specifically, we account for the spatial and sequential structure of language using a convolutional-LSTM model. Second, we find that attribute silence in reviews is driven by need to inform and need to praise/vent, but not based on the importance that the reviewer itself places on the attribute. Using a structural model of rating behavior, we develop a model-based imputation for missing attribute ratings. Overall, we note that though the paper is motivated in the empirical context of online reviews, the problems of generating fine-grained attribute sentiment scoring from text and the interpretation/correction of attribute silence has broad application across many settings.

The rest of the paper is organized as follows. §2 discusses the related literature. §3 describes the problem of attribute sentiment scoring, the challenges and how our model addresses these challenges. §4 describes the structural model of rating behavior, the estimation strategy, and how the model is used for imputing missing attribute scores. §5 describes our data. §6 summarizes the results. §7 concludes.

RELATED LITERATURE

This paper is related to multiple strands of literature in marketing and computer science. We organize our discussion in two parts.

Text Analytics on UGC and Online Reviews

Table 2 positions our paper with respect to the most relevant literature on online reviews and user generated content in marketing. Some of the early research on user-generated (UGC) content in marketing (e.g., Chevalier and Mayzlin 2006, Dhar and Chang 2009, Duan et al. 2008, Ghose and Ipeirotis 2007, Onishi and Manchanda 2012) uses quantitative metrics like review ratings, volume and word count to infer the impact of UGC on business outcomes like sales and stock prices. Though these papers established the importance of studying UGC and its specific role in

Electronic copy available at: https://ssrn.com/abstract=3395012

experience goods markets, they did not investigate content in review text.

Another research stream focused on using UGC content in blogs and review forums to extract insights around customer needs and brand positioning (e.g., Lee and Bradlow 2011, Netzer et al. 2012, Tirunillai and Tellis 2014, Büschken and Allenby 2016). Archak et al. (2011) use UGC to measure sentiment valence (not fine-grained sentiment) on specific product attributes using a lexicon approach and its impact on demand.

[Insert Table 2 here]

Fine-grained sentiment analysis for individual attributes is one of the more challenging variants of the sentiment analysis problem (Feldman 2013, Wang et al. 2010). Figure 1 shows the evolution of sentiment analysis literature, highlighting the trade-offs of the different approaches. Lexicon or dictionary based methods (Wang et al. 2010, Taboada et al. 2011) are highly interpretable, but rely on carefully hand-crafted features. They are therefore not scalable. They under-perform in detecting sentiments in "hard" sentences. Early supervised text classification methods like SVM (Joachims 2002) do not need hand-crafting and are scalable but they need large amounts of labeled training data (tagged by humans) to reach desired levels of accuracy. Hence deep learning models (Kim 2014, Socher et al. 2013, Zhou et al. 2015) combined with meaning-infused word vectors (Pennington et al. 2014, Mikolov et al. 2013) have revolutionized the field of text mining — they do extremely well on text classification tasks, yet require only much smaller volume of training data to attain high levels of accuracy. Thus they overcome the shortcomings of both traditional supervised as well as unsupervised algorithms. A limitation is that they lack interpretability and so it is hard to understand what is driving the performance of deep learning models. Recently, marketing scholars have used deep learning models for text analysis to answer important questions such as need identification (Timoshenko and Hauser 2018) and the impact of reading reviews on particular attributes on purchasing decisions (Liu et al. 2019), but their focus is not on fine-grained sentiment and hence language structure is less important.

[Insert Figure 1 here]

Electronic copy available at: https://ssrn.com/abstract=3395012

We advance the marketing literature on sentiment analysis in two ways: (i) considering fine-grained attribute sentiment scoring and (ii) moving from "bag-of-words" methods like LDA and lexicons to deep learning models that account for structural aspects of language. Hybrid models that combine features of different deep learning architectures can improve performance on hard tasks (Wang et al. 2016); in that spirit, we motivate and construct a hybrid convolutional-LSTM model. Further, to understand the key drivers of model performance, we test our model on various types of hard sentences. In our corpus, nearly half of the sentences are 'hard," justifying the need to account for language structure. By reporting performance metrics not just overall, but on types of "hard" sentences, we offer new benchmarks for performance evaluation in future research.

Missing Attributes (Attribute Silence) in Reviews

Our study of attribute silence, i.e. missing attributes in text reviews is primarily related to the statistics literature on missing data and imputations. Rubin (1976) laid the seminal framework for analysis of missing data, in which every data point has some likelihood of being missing. Rubin classifies missing data problems into three groups: "Missing Completely at Random" (MCAR), "Missing at Random" (MAR), and "Missing Not at Random" (MNAR). MCAR occurs if the probability of missing is the same for all cases, i.e., causes of the missing data are unrelated to the data. This assumption is likely violated in most settings. Most modern imputation models for missing data are based on the MAR assumption; i.e., the probability of being missing is the same within groups defined by the observed data. For this strategy to be successful, rich behavioral models (including those with unobserved heterogeneity) are modeled on the behaviors of interest, such that the MAR assumption becomes reasonable. While many MAR models are based on observed heterogeneity, in our setting given the potential unobserved heterogeneity in rating styles across reviewers, the MAR strategy will be unsuccessful without unobserved heterogeneity. Fortunately, in our setting given multiple observations across restaurants and reviewers, unobserved reviewer heterogeneity can be estimated and the MAR approach can be applied. If not, then we have an MNAR setting potentially due to unobserved heterogeneity. The most common approach is to then intro-

Electronic copy available at: https://ssrn.com/abstract=3395012

duce new identifying restrictions by explicitly justifying a model of missingness for the context at hand, and estimate the joint model of missingness with the behavioral model (Little and Rubin 2019, Mohan and Pearl 2018). Overall, missing data models are often estimated using Multiple Imputation, or by likelihood methods. Likelihood based approaches either use Bayesian methods or the EM algorithm for estimation. Recently, Athey et al. (2018) proposed matrix completion methods for imputation in big data settings.

In this paper, we develop a structural model of heterogeneous reviewer rating behavior that allows for both observable and unobservable heterogeneity taking into account the data generation process. We allow for a rich nonlinear mapping from experienced utility to five-level rating and weighted mapping of attribute ratings to overall rating behavior. We use an EM algorithm to estimate the model, with model-based imputation to fill in for missing attribute ratings during the EM iterations. Ex-post, we use the parameters of the structural model to assess various conjectures of attribute missingness. We also use the estimates of the structural model to impute for missing attribute ratings to construct aggregate corrected metrics of restaurant ratings, conditional on the observed characteristics of restaurants and the observed and unobserved characteristics of reviewers.

CONVERTING TEXT INTO NUMERIC ATTRIBUTE SENTIMENT SCORES

We first describe the attribute level sentiment analysis problem of converting unstructured text data in reviews into attribute level sentiment scores. We then describe two methods of attribute scoring models with text data: (1) the lexicon model and (2) the deep learning model. Along the way, we also describe various implementation issues and choices that needs to be made.[5]

The problem of attribute level sentiment analysis is to take a document d as input (in our empirical example, a Yelp review) and identify the various attributes $k \in K$ that are described in d, where K is the full set of attributes. Having identified the attributes k, the problem requires associating a

[5]For completeness, we also estimate some bag-of-words based supervised machine learning models e.g., Support-Vector-Machine (SVM), Naive Bayes and Logistic Regression as baseline models as they have been used for text classification in the past. We also estimated a supervised topic model S-LDA, but do not report the results as it does not separate attribute and sentiment classes well; a primary requirement for this task.

Electronic copy available at: https://ssrn.com/abstract=3395012

sentiment score s with every attribute. In solving the attribute level sentiment problem, we make two simplifying assumptions. First, we assume that each sentence is associated with one attribute. Occasionally, sentences may be associated with more than one attribute; in that case, we consider the dominant attribute associated with the sentence. Like Büschken and Allenby (2016), we find that in our empirical setting, multiple attribute sentences account for less than 2% of sentences in our review data, and thus have very little impact on our results. Second, we assume that the attribute-level sentiment score of a review is the mean of the sentiment scores of all sentences that mention that attribute. We outline the steps involved in obtaining attribute level sentiment ratings from text reviews in Table 3.

[Insert Table 3 here]

Most of the steps in table 3 are clear, except for the choice of attribute/sentiment classes (Step 0) and the attribute sentiment classifier used (Step 5). We begin by describing how we choose the relevant attributes and the sentiment scale in Step 0. We use a 1-5 scale for sentiment granularity (1: extremely negative, 3: neutral and 5: extremely positive) as this is comparable to the 5 point rating scale in many review platforms. Also, human taggers fail in practice to differentiate well between classes when the sentiment granularity is higher than 5 levels (Socher et al. 2013).

To obtain comparable fine-grained sentiment scores on a managerially relevant set of attributes across restaurants, we first need to choose a set of attributes on which restaurants should be scored on. This is similar to an exploratory phase before conducting a quantitative survey. For this, we conducted (i) a review of the literature; (ii) an analysis of the most frequent attribute words in the corpus; and (iii) topic modeling using LDA. The literature on restaurant evaluation and industry customer satisfaction surveys identified food quality, employee behavior and wait time (service), basic hygiene, look and feel (ambiance) and value for money as the most common attributes (Ganu et al. 2009). We then did frequent word categorization of our review corpus by associating the most high frequency nouns, noun phrases and select verbs to restaurant-relevant attributes. Beyond the four attributes identified from past literature and industry surveys, we found a fifth attribute "location" that has words pertaining to parking, convenience and safety of the restaurant location.

Electronic copy available at: https://ssrn.com/abstract=3395012

Finally, we conducted topic modeling of our review corpus using LDA. As is common with LDA, these topics combined both restaurant attributes and consumer sentiments, and given the very high frequency of food related comments, the topics were disproportionately around food.[6] Overall, we concluded that the five attributes—food, service, ambiance, value and location captured the most relevant attributes for a restaurant rating platform. (Socher et al. 2013).

Figure 2 illustrates and clarifies the major steps in attribute sentiment scoring using an example review. These steps above are the same irrespective of the Attribute Sentiment Classifier (*AS*) used in Step 5 of Table 3. We next describe the two types of attribute sentiment classifiers we consider.

[Insert Figure 2 here]

Attribute Sentiment Classifier: The Lexicon Method

We begin with the lexicon-based method because it is highly interpretable, transparent and very widely used and thus serves as useful benchmark relative to more complicated models. The method consists of lexicon construction followed by attribute sentiment classification of text based on dictionary look-ups; i.e. sentences are classified into an attribute and sentiment class by locating word matches in attribute and sentiment class-specific dictionaries. We explain the method below and discuss its limitations.

1. Lexicon building. Lexicon construction involves creating a dictionary of attribute words with corresponding attribute labels (e.g., *waiter*–"service") and sentiment words with sentiment class labels (e.g., *excellent*–"extremely positive"). We first identify the high-frequency attribute and sentiment words in our corpus to create our *vocabulary*. We construct attribute and sentiment class-specific dictionaries, by asking human taggers on Amazon Mechanical Turk to classify all attribute words into one of the five attributes we identified in Step 1—food, service, value, ambiance and location and all sentiment words into one of the five sentiment classes–given we decided to use a

[6]Büschken and Allenby (2016) note that by initializing the LDA model with seed-words for a wider range of attributes, one could obtain more balanced topics. Since we only needed to identify relevant topics and not gain greater balance, using seed-words did not help with identifying additional attributes that were relevant for a large enough set of restaurants to be used on a platform

Electronic copy available at: https://ssrn.com/abstract=3395012

5 point rating scale. Every word is labeled by 3 distinct human taggers and we retain only those words for which at least 2 out of 3 taggers agree on the labeling.[7]

2. Attribute Level Sentiment Scoring. Each review is split into sentences. Using the lexicon, each attribute word in the sentence is classified into one of the pre-specified attributes (or none) and each sentiment word is classified into a 1-5 sentiment rating scale using a "look-up" or search of the pre-created lexicons. Following this, the steps are similar to those listed in Table 3

Despite its simplicity, interpretability and transparency, the method has several limitations. First, lexicon construction is costly in both time and effort, and scales linearly with number of words. Second and more importantly, the method treats language as simply a bag-of-words or "fixed phrases" and does not account for various aspects of language structure. In practice, lexicon methods therefore work fairly well for sentiment identification in simple sentences, but perform poorly on "hard" sentences. (Liu et al. 2010).

Why the Lexicon Method Fares Poorly with "Hard" sentences. We elaborate further on why lexicon methods fail to classify *hard* sentences that we had mentioned in the introduction in Table 1 (Socher et al. 2013). This is problematic because "hard" sentences are close to 50% of sentences in our review corpus. We now explain each of these types.

1. Negations and Sentiment Degree. Sentences which have different degrees of negative sentiment can be hard to classify without accounting for variable size n-grams. Lexicon methods typically look at one word at a time and will not be able to obtain sentiment valence or degree; Even if ad hoc approaches may be used to address standard negations with bi-grams or tri-grams by hard-coding negation phrases, examples like "*Pizza is not that good*," "*Pizza is not at all great*," illustrate that such ad hoc approaches are unlikely to be effective overall in capturing degree of sentiment. This motivates the use of the convolutional layer, which handles the spatial structure.

2. Long Sentences and Scattered sentiments. In long sentences consisting of more than 20 words,

[7]While it is possible to use a previously constructed generic lexicon to label attributes and to assign sentiment scores, a domain and task specific lexicon improves classification/labeling accuracy. Moreover, we could not find any existing lexicon that is well-suited for fine-grained sentiment analysis of restaurant reviews. For e.g., AFINN lexicon (Nielsen 2011) and Stanford Sentiment Treebank (Socher et al. 2013) have words and phrases with 5-levels of sentiment classification, however, they are built on Twitter and rotten.tomatoes.com movie review dataset respectively and have limited overlap of words and attributes with our restaurant domain.

Electronic copy available at: https://ssrn.com/abstract=3395012

the degree of sentiment (and even polarity) can change multiple times. As an example,“ *OK, in fact good, to start with but kept getting worse and wait staff were unapologetic but manager saved the night*.” In this sentence, the sentiment flows from being good to bad to extremely bad and then back to positive. Yelp reviews tend to have a significant percentage of long sentences. Without sequence history, the classifier cannot capture sentiment shifts and will classify most of these sentences as *neutral* due to the mix of positive and negative sentiment words. More importantly, immediate sentiment modifiers may be changed by sentiment words that are farther away, so having a “long term memory” of what was said before and whether recent sentiment (short-term memory) should take precedence needs to be considered. The LSTM layer helps with both the sequencing and the immediate and distant sentiment modifiers, while the convolutional layer still helps group words into phrases within the long sentence before being fed into the LSTM layer.

3. Contrastive conjunctions. Sentences which have an *X but Y* structure often get misclassified by sentiment classifiers as the model needs to take into account both the clauses before and after the conjunction and weigh their relative importance to decide the final sentiment. An example sentence includes “*Despite the creativity in the menu, execution was a disappointment*.” The first half here is extremely positive due to the word *creativity*, but the second half moderates it significantly. A good classifier should be able to learn from both parts of the sentence to arrive at the correct classification. While the convolutional layer identifies phrases before and after the conjunction, the LSTM layer helps with interpreting the change of meaning after the conjunction.

4. Implied sentiments (sarcasm and subtle negations). These sentences do not have explicit positive or negative sentiment words but the context implies the underlying sentiment. This makes the task of sentiment identification extremely hard for all classes of models and especially for models relying on a specific set of positive or negative words. An example sentence includes “*The place is a treasure if only you are lucky to be there on the right day*.” This is an example of sarcasm, the reviewer uses a positive word like “treasure” but hints at the extreme variance in the type of experience one can have. There could also be subtle negations, for example, “*The girl managing the bar had to be the waitress for everyone*.” Here the reviewer is complaining about lack of

Electronic copy available at: https://ssrn.com/abstract=3395012

service arising out of shortage of staff without using any explicit negative word. Given the meaning/sentiment associated with the work lies in the richer context of its usage, we will empirically assess how much the spatial and sequential structure helps with accurate classification.

Attribute Sentiment Classifier: A Deep Learning Hybrid Convolutional-LSTM Model

Lexicon methods use a constructive algorithm based on pre-coded attributes and sentiment words in a lexicon to score attribute level sentiment. In contrast, deep learning models are a type of supervised learning model, where the model is trained using a training dataset by minimizing a loss function (e.g., the distance between the model's predictions and the true labels). The trained model is then used to score attribute level sentiment on the full dataset. Like deep learning, regression and support vector machines (SVM) are also variations of supervised learning.

What distinguishes deep learning from regression and support vector machines is that deep learning seeks to model high-level abstractions in data by using multiple processing layers (the multiple layers give the name "deep"), composed of linear and non-linear transformations (Goodfellow et al. 2016). Deep learning algorithms are useful in scenarios where feature (variable) engineering is complex and it is hard to select the most relevant features for a classification or regression task. For instance, in our task of fine-grained sentiment analysis, it is not clear which features (combination of variable length *n*-grams) is most informative in order to classify a sentence into "good food" or "great service". The two key ingredients behind the success of deep learning models for NLP are meaningful word representations as input and the ability to extract contiguous variable size n-grams (spatial structure) with ease while retaining sequential structure in terms of word order and associated meaning.

In this section, we outline the architecture of the model and its intuition and discuss critical modeling/implementation choices.[8] Figure 3 shows the general architecture of a neural network used for text classification. Following pre-processing of text, the first layer is the embedding layer, where words are converted to numerical vectors by making use of word embeddings. These

[8]The technical description is provided in a self-contained online appendix for the interested reader.

Electronic copy available at: https://ssrn.com/abstract=3395012

embedded numerical vectors are then fed to the succeeding feature generating layers, which are the core of the deep learning model. In contrast to older supervised learning methods like SVM which work with the raw data directly as inputs, these feature generating layers, i.e., the convolutional layer and long short term memory network (LSTM) layer in our model, extract higher level features important for classification. The extracted feature vectors are then passed into a logit classifier (soft-max) that classifies the sentence to the class with highest probability of association.

Embedding Layer and Word Representation. Neural network layers work by performing a series of arithmetic operations on inputs and weights of the edges that connect neurons. Hence, words need to be converted into a numerical vector before being fed into a neural network.[9] These vectors are called *embedding* and most well-known embedding algorithms (e.g., word2vec, GloVe) are based on the distributional hypothesis— words with similar meanings tend to co-occur more frequently (Harris 1954) and hence have vectors that are close in the embedding space. The efficiency of the neural network improves manifold if these initial inputs carry meaningful information about the relationships between words. Hence the choice of embedding is an important one — we experiment with both embeddings trained from scratch on our Yelp review corpus as well as a range of pre-trained word embeddings like Word2Vec (Mikolov et al. 2013) and GloVe (Pennington et al. 2014) that are available for all words in our *vocabulary*.[10] There are pros and cons for both approaches — pre-trained embeddings is a form of transfer learning that eliminates embedding generation time, but self-trained embeddings may result in higher classification accuracy due to a more context-relevant vocabulary.

Feature Generating Layers (Convolutional-LSTM). The macro architecture of the neural network comprises of layers to be included (e.g., *feed-forward* or *convolutional*) and type of interconnections between them. As discussed above, the most challenging aspect of our task is dealing

[9]The simplest method to form numeric vectors from words is a one-hot representation which means that if there are V words in the vocabulary; each word is represented as a $V \times 1$ dimensional vector where exactly one of the bits is 1 and rest are zero. Such a representation is not scalable for large vocabularies and also stores no semantic information about words. Another option is to only take into account word frequency and simply convert words into numbers based on some normalized frequency score like *tf-idf*.

[10]These embeddings have been trained on different corpus like Wikipedia dumps, Gigaword news dataset and web data from Common Crawl and have more than 5 billion unique tokens.

Electronic copy available at: https://ssrn.com/abstract=3395012

with different types of hard negations resulting from variable-size n-grams (e.g., *not good, not that great*) and shifting polarities (*started off well but ended in a sorry surprise*). In many challenging text and image classification problems (Wang et al. 2016), hybrid models that combine the strengths and mitigate the shortcomings of each individual model have been found to improve performance. In that spirit, we build a network consisting of a single convolutional layer with variable-size filters followed by a Long Short Term Memory (LSTM) layer.

Convolutional layers with different filter sizes specialize in extracting variable-length *n*-grams (phrases) associated with relevant attributes and sentiments and have recently been used successfully in various text analysis applications (Kim 2014, Timoshenko and Hauser 2018). To improve granular sentiment detection where sequence information is critical, we follow the convolutional layer with an LSTM layer that processes the features (phrases) identified from the convolutional layer. LSTM is a variant of the recurrent neural networks (RNN) that specializes in handling longer contextual information (Hochreiter and Schmidhuber 1997). An LSTM employs a cell state (long-term memory) and a combination of gates that are like "regulators" of information to constantly evaluate what parts of the history (in this case n-grams from earlier part of the sentence) need to be forgotten and what needs to be retained to improve the accuracy of the attribute and sentiment classification task.[11] As we motivated in our discussion of "hard" sentences, by taking advantage of the properties of the convolutional layer and LSTM, we expect the hybrid to improve classification accuracy while keeping training time low.

[Insert Figure 3 here]

Classifier. The loss function choice depends on the nature of the classification task. Since our tasks involve the classification of text into 5 attribute classes and 5 sentiment classes, it is a multi-class classification problem. We use the standard loss function for multi-class classification called Categorical Cross Entropy. Say s_i represents the convolutional-LSTM model classification for sentence i and t_i represents the ground truth classification, then the cross entropy loss function

[11] For more details on this architecture, see online appendix.

Electronic copy available at: https://ssrn.com/abstract=3395012

can be defined in the following manner :

$$Categorical\ Cross\ Entropy\ Loss\ (CCE) = -\sum_{i}^{C} t_i log(s_i)$$

Deep Learning Implementation: Important Choices

Word Embeddings. We tested pre-trained embeddings based on word2vec and GloVe with different numbers of embedding dimensions (e.g., 100, 300) for attributes and sentiment classification. Further, we evaluated whether self-trained embeddings from the specific text corpus can produce superior classification relative to the pre-trained embeddings.

Micro Architecture. The micro architectural decisions in a neural network involve the number of neurons in each of the layers, the size and number of filters for the convolutional layer and dimensions of the max pooling function (that concatenates variable-size feature vectors generated from variable-size convolutional filters). Many of these decisions are empirically driven but some factors that inform these choices are: sentiment classification would rely on presence of long-range n-grams, so we would typically chose a mix of filter sizes for this task ranging from 1-6 grams. In contrast, the attribute classification task often needs only unigrams and bi-grams (*chicken, cola drink, wait time*) and hence simple unigram and bigram filters would be sufficient. Also, since the sequence of *n*-grams matters for sentiment classification, ideally we should not use a max pooling layer after the convolutional layer as the aggregation loses sequential information before being passed to the LSTM layer. However, a pooling layer is needed to merge variable-size feature maps generated from the convolutional filters. We balance this tradeoff by max-pooling on the smallest possible pooling dimension so that we can preserve as much of the sequence information as feasible in sending input into the LSTM layer.

Model Training. As is standard for deep learning models, the model parameters are optimized jointly by training the model iteratively on smaller sub-samples of the training data (mini-batches) and then using the estimation error to improve the model (i.e. change the weights and biases in small increments) through a feedback loop. We experimented with mini-batch sizes of 5, 10, 25,

Electronic copy available at: https://ssrn.com/abstract=3395012

30, 50 and different optimizers. We chose the RMSProp (Bengio and CA 2015) optimizer because it uses an adaptive learning rate.

Performance Measures for Model Comparison

The primary metric on which we compare our models is accuracy or hit rate. This metric is formally defined as:

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{1}$$

where tp, tn, fp, fn stand for true positives, true negatives, false positives and false negatives respectively. Accuracy is the most common metric that is used for evaluating granular text classification problems and is a fairly good metric unless there is a class imbalance issue (i.e. some classes are not well-represented in the training or test dataset). While we try to maintain class balance in our data sets, equal representation of all classes is difficult as some classes like food, service appear much more often in Yelp reviews than other classes. Likewise, moderately positive sentiments are more common than extremely positive or negative sentiments.

Among the models that do equally well on accuracy, we further evaluate them based on two types of accuracy metrics that capture not just error-rate but also the type of errors that occur.

Simple Confusion Matrix for Attribute Classification Accuracy: This confusion matrix helps to evaluate class-wise accuracy—doing so allows us to assess whether overall higher accuracy comes only from superior performance in high high-occurrence classes like food or a class like location that has few attribute words. We can assess whether the model is able to capture more complex classes like ambiance and service which manifest with a varied set of attribute words.

Polarity Reversal Confusion Matrix for Sentiment Accuracy: Though the CS literature typically uses accuracy as a performance metric for the fine-grained (multi-class) sentiment classification (Socher et al. 2013, Kim 2014), there can be other useful metrics of performance. For example, it may be useful to construct a polarity based coarse class: positive, neutral, negative and assess

Electronic copy available at: https://ssrn.com/abstract=3395012

accuracy on the coarse classes classification because confusing sentiment class 1 with 4 or 5 (a polarity reversal) is worse than confusing 1 with 2 (same polarity). With this thought, we construct and report a polarity reversal confusion matrix for the models that have the best overall accuracy.

We also evaluate model performance on specific hard sentence types (e.g., long and scattered sentiments, contrastive conjunctions and implied sentiments) that we discussed earlier in motivating why we account for the spatial and sequential structure of language . Finally we also assess qualitative factors like model building effort, scalability and interpretability for the various models.

ANALYSIS OF STRUCTURED RATINGS ACCOUNTING FOR MISSING ATTRIBUTES

In the first part of the paper, we converted review text into numerical attribute scores on a 1-5 scale and attributes were coded as "missing" when reviewer is silent on an attribute. For every review, we also have an overall rating on the restaurant. The challenge is how to impute the missing attribute ratings to obtain the correct aggregate attribute rating. We now outline our model-based imputation strategy to correct for attribute silence, before providing specific details.

We first develop a structural model of rating behavior that allows for (1) nonlinear mapping from experienced quality to attribute ratings; (2) heterogeneity in rating styles; and (3) heterogeneity in weights of attribute ratings on the overall restaurant rating. We then use an iterative two step EM algorithm to estimate the model, where the mapping from experienced quality to the nonlinear, heterogeneous attribute rating is estimated in the first step, and the heterogeneous weights that link attribute rating and overall rating is estimated in the second step. In each iteration, when attribute rating is missing in the review, we impute the attribute rating based on the model estimates in the current iteration. We iterate till the model converges.

The structural model estimates give us insights into reviewer segments and reviewer rating behaviors. We use the estimates to assess whether our conjectures on attribute silence has support in the data. We then assess the validity of the model-based imputations on a holdout sample. Finally, we illustrate that corrections for attribute rating using the imputations can be substantial.

Electronic copy available at: https://ssrn.com/abstract=3395012

A Structural Model of Rating Behavior

Every reviewer who writes a review has an experience with the restaurant. Let A^*_{jk} be the experienced latent quality at restaurant j on the attribute k. The experienced latent utility is a function of observable restaurant characteristics associated with the attribute X_{jk} and an idiosyncratic shock that varies across visits.[12] Specifically, let

$$A^*_{ijk} = \alpha_k X_{jk} + \nu_{ijk}$$

where ν_{ijk} follows a Type I extreme value distribution (TIEV).

The mapping from underlying latent utility A^*_{jk} to the 5 point rating scale A_{jk} can be nonlinear and heterogeneous across reviewers in terms of both observable and unobservables. Specifically, we formulate the nonlinear mapping from latent experienced utility A^*_{ijk} to an ordinal rating A_{ijk} (1-5 scale) as an ordinal logit model, given that we assume ν_{ijk} to be TIEV:

$$A_{ijk} = s, \text{ if } C^g_{k(s-1)} < A^*_{ijk} + \beta^g_k X_i \leq C^g_{ks} \tag{2}$$

where $C^g_{k(s-1)}$ and $C^g_{k(s-1)}$ are the cutoffs of reviewer segment g for attribute k, score s ($C^g_{k0} = -\infty$, $C^g_{k5} = \infty$). The thresholds C^g_{ks} increase monotonically over s. While X_i captures the effect of observable characteristics on thresholds, the cut thresholds C^g_{ks} can capture the unobserved heterogeneity in reviewer's attribute rating style (for high and low scores) and differences with respect to attribute expectations, which determine satisfaction.

Further, we observe the overall rating of the restaurant for all reviews. It is natural to treat the overall restaurant rating as arising from a weighted sum of the ratings on attributes, allowing for both observable and unobservable reviewer heterogeneity (by same latent class as for attribute ratings). Specifically, we formulate the ratings equation as

[12] This experienced quality can vary over time t as a function of observable restaurant characteristics that vary over time, but for simplicity of notation, we suppress the t subscript in the exposition.

Electronic copy available at: https://ssrn.com/abstract=3395012

$$R_{ij} = \gamma_0^g + \sum_k \gamma_k^g A_{ijk} + \varepsilon_{ij} \tag{3}$$

Model Estimation and Missing Attribute Ratings

To the extent that there are no shared parameters across equations (2) and (3), the two equations can be estimated independently. However, given the unobserved heterogeneity, the model needs to be estimated using an iterative two step EM algorithm. Each equation is estimated in a separate step, then the posterior of the heterogeneity distribution is obtained using Bayes rule, and the iterations continue conditional on the posterior heterogeneity from the previous step until there is convergence in the heterogeneity classification of the reviewers.

In our setting, where reviewers are silent on several attributes, A_{ijk} is missing in many reviews for many attributes. In the absence of unobserved heterogeneity, imputation would be a straightforward prediction based on observable restaurant and reviewer characteristics. However since A_{ijk} is also a function of unobserved characteristics of reviewer i, the imputation needs to condition on the unobserved heterogeneity and iterated through the EM algorithm. Specifically, we use the prediction from the first step (ordinal logit), conditional on the unobserved heterogeneity as the imputation of attribute rating in the second step and iterate till convergence.

The predicted probability of the attribute score being s ($s \in 1,2,3,4,5$) from the ordinal logit model of first step is

$$p_{ijks}^g \equiv Pr(A_{ijk} = s) = Pr(C_{k(s-1)}^g < A_{ijk}^* + \beta_k^g X_i \leq C_{ks}^g) \tag{4}$$

Based on the estimated β_k^g and C_s, we compute probabilities of each attribute rating for each k and s for each latent segment g when attribute rating is missing.

We then estimate the rating equation with imputation when attribute ratings are missing:

Electronic copy available at: https://ssrn.com/abstract=3395012

$$R_{ij} = \sum_g q_i^g \left[\gamma_0^g + \sum_k \gamma_k^g \left[\underbrace{(1 - M_{ijk}) A_{ijk}}_{\text{if present}} + \underbrace{M_{ijk}}_{\text{if missing}} \underbrace{\sum_{s=1}^{5} (s p_{ijks}^g)}_{\text{Expected}} \right] \right] + \varepsilon_{ij} \tag{5}$$

where R_{ij} is reviewer i's star rating for restaurant j, M_{ijk} is whether the rating for attribute k is missing. If the attribute rating is present (i.e., $M_{ijk} = 0$), we use observed attribute rating A_{ijk}, and otherwise, we use expected attribute rating $\sum_{s=1}^{5}(sP_{ijk}^s)$ as the input. We estimate intercept γ_0^g, attribute importance γ_k^g and probability of reviewer i belonging to segment g, q_i^g. Thus, the parameters to be estimated are $\Theta = \{\alpha_k, C_{ks}^g, \gamma_0^g, \gamma_k^g, q_i^g\}$

To be specific, the EM estimation procedure is the iteration between E (Expectation) and M (Maximization) steps below.

1. Initialization: Determine initial value of parameters $\Theta^{(1)}$ through MLE by assuming no unobserved heterogeneity across reviewers. Assume that each reviewer is equally likely to be in each segment (i.e., $q_i^g = \frac{1}{N_g}$, where N_g is the number of segments).

2. E step: For reviewer $i = 1, 2, ..., m$, given the n^{th} parameter $\Theta^{(n)}$, compute $p_{ijks}^{g(n)}$, the predicted probability of the attribute score of each review.

3. M step: Estimate $(n+1)^{th}$ parameters $\Theta^{(n+1)}$ by iteratively maximizing the likelihoods in step 1 and step 2.

   (a) Step 1: Attribute Rating

$$\sum_i^m log(L(\alpha, C_s, \pi)) = \sum_i^m log(\sum_g \pi_g L_{ig})$$

   where individual reviewer's likelihood $L_{ig} \equiv L_i(\alpha, C_s^g) = \prod_j \prod_l^4 \left[p_{ijk,l-1}^g - p_{ijkl}^g \right]^{1(s_{ij}=l)}$ and π_g is the segment size.

Electronic copy available at: https://ssrn.com/abstract=3395012

(b) Step 2: Overall Rating

$$\sum_{i}^{m} log(L(\alpha, C_s, \pi)) = \sum_{i}^{m} log(\sum_{g} \pi_g L_{ig})$$

where $L_{ig} \equiv L_i(\gamma_0^g, \gamma_k^g) = \prod_j \phi(\varepsilon_{ijk} | \gamma_0^g, \gamma_k^g)$

4. Iterate between E step and M step until convergence.

The E step is internally consistent because the imputation is based on observed and unobserved heterogeneity conditional on estimates of every iteration. Our imputation strategy works because we have multiple observations on attribute ratings and overall reviewer ratings—even if some of the attribute ratings are missing. We are able to identify the heterogeneity in attribute rating styles (the unobserved thresholds C_{ks}^g) as long as we have variation in attribute ratings on a subset of reviews from every reviewer, conditional on latent experience which are identical across reviewers and vary only by restaurant observables.

What Drives Attribute Silence?

We conjecture several possibilities for why a reviewer may be silent about some attributes in a review: Informativeness, Importance, and Praise/Vent need.

1. *Informativeness*: Reviewers write on review platforms to share their experience with others, so one of the major motivations could be to inform or add new information (Berger 2014). For instance, price and location may be written about less because they are not only search attributes, but prices are usually described categorically on review platforms. Location information may be often obtained through the address and other information. Hence such attributes may be described less overall. But even with experience attributes like food, service and ambiance, there may be variations in motivations across restaurants and time. For example, a review may be informative if there is high variance in predicted utility for a restaurant, i.e., there is high variance in past reviewer ratings. Controlling for variance, if the restaurant's average attribute rating is very different from the reviewer's corresponding rating, the reviewer may also consider it informative to

Electronic copy available at: https://ssrn.com/abstract=3395012

write a review. Empirically, we assess the conjecture testing whether attribute presence (silence) is positively (negatively) related to (i) variance in past reviewer ratings, and (ii) difference between the restaurant's average attribute rating and the reviewer's rating. Further, we test whether there is heterogeneity for positive and negative deviations.

2. *Importance*: A reviewer may be silent about an attribute if it is unimportant for the reviewer. To assess if attribute silence may vary by its importance on overall ratings across unobserved segments, we empirically assess whether attribute presence (silence) is correlated with attribute weights (γ_k^g) derived from the structural model controlling for attribute type.

3. *Praise/Vent Need*: Some reviewers may feel the need to praise/vent, when highly satisfied or dissatisfied, but not write when the rating is average (when it is a three). For this we assess whether silence varied by attribute rating level. To assess this conjecture, we compare the probability of the attribute score being s when the attribute is missing ($Pr(A_{ijk} = s|M_{ijk} = 1)$ versus when it is present ($Pr(A_{ijk} = s|M_{ijk} = 0)$. Let us define a ratio $\pi_{gk}^s = \frac{Pr(A_{ijk}=s|M_{ijk}=0)}{Pr(A_{ijk}=s|M_{ijk}=1)}$. If the ratio is larger than 1 (i.e., the probability is larger in the case of missing), a reviewer who evaluates the attribute as score s is more likely to miss the attribute. In other words, π_{gk}^s captures how likely the attribute's true score is s when it is missing vs. present. For segment g and attribute k, π_{gk}^l should be larger than 1 when score l represents satisfaction level that is more likely to be missing.

EMPIRICAL APPLICATION

Data

Yelp is a crowd-sourced review platform where reviewers can review a range of local businesses e.g., restaurants, spas & salons, dentists, mechanics and home services to name a few. The website was officially launched in a few U.S west coast cities in August of 2005 and subsequently expanded to other U.S cities and countries over the next few years. As of Q1 2017, Yelp is present in 31 countries, with 177 million reviews and over 5 million unique businesses listed (Yelp Investor Relations Q4 2018). Given our empirical application, we focus on restaurant reviews. Since 2008, Yelp has shared review, reviewer and business information for select U.S and international

Electronic copy available at: https://ssrn.com/abstract=3395012

cities as part of its annual challenge. Unique reviewer and business identification numbers in the data helps create a two-way panel of reviews at reviewer and business level. For each review, we observe overall rating, textual evaluation and date of posting as well as information about business characteristics (e.g., cuisine, price range, address, name) and reviewer characteristics (e.g., experience with Yelp, Elite membership). Table 4 summarizes the various data sets we use for different types of analysis. A discussion on each dataset follows.

[Insert Table 4 here]

1. Exploratory Analysis. We use the full dataset of 1.2 million restaurant reviews for the exploratory analysis to identify attribute and sentiment classes that we described in the model section. We created a *vocabulary* of 8458 words consisting of both sentiment and attribute words.[13] We then did a Parts of Speech tagging of our word list i.e. we classified our word list into adjectives, adverbs, nouns and verbs so as to separate attribute and sentiment words. Attribute words are mainly nouns whereas sentiment words are adjectives and adverbs with some important exceptions: for instance, some verbs are strong indicators of an attribute. e.g, "greeting", "seated", "served" refer to *service* and "spent" refers to *value*.[14] Finally human taggers classified the attribute and sentiment words into attribute and sentiment classes. In our dictionaries, we only retain those words that have been labeled into a particular class by at least 2 out of 3 taggers.[15]

2. Training and Test Data for Supervised Learning . For supervised learning, we constructed another data set at the sentence level. Human taggers classify the sentences into its primary attribute and sentiment level. We ensured this dataset of sentences is balanced in its representation of all attribute and sentiment classes. 75% of this data was used for training and the remainder for model validation and testing. See Table 5 for the composition of training and test data sets.

[13] We excluded stop-words, meaningless phrases and the long tail of words with occurrence frequency less than 1500 in our corpus.

[14] Some adjectives are good indicators of both attribute and sentiment for e.g. the word "cheap" invariably refers to price attribute in a negative way whereas some descriptive adjectives strongly refer to an attribute for e.g., decorated refers to ambiance.

[15] Our attribute and sentiment dictionaries are available upon request. These are more detailed relative to previous studies (Pak and Paroubek 2010, Berger et al. 2010) that focus on two (i.e. positive and negative) or three levels (i.e. positive, neutral and negative) of sentiments.

Electronic copy available at: https://ssrn.com/abstract=3395012

[Insert Table 5 here]

An discussed in §3, lexicon methods cannot deal with hard sentence types. Table 6 shows the distribution of different sentence types in a randomly sampled subset of sentences from our corpus. 48% of all sentences and 66% of the negative sentences belong to one of the complex types. Long sentences account for 27% of our data. Given their empirical importance, we created a special test dataset of hard sentence types to assess model performance specifically on such sentence types.

[Insert Table 6 here]

3. Restaurant and Reviewer Stratified Sample. To estimate the linkages between attribute level sentiment and overall ratings, we focus on a stratified sample of reviews. We ensure that we have multiple reviews by individuals so that we can account for unobserved heterogeneity in reviewer rating styles. We want multiple reviews on restaurants to ensure that there are multiple reviewers who obtained similar latent utilities up to a random shock. We therefore restricted our sample to only individuals that posted at least 5 reviews and restaurants that have at least 20 reviews.[16]

We then used stratified sampling by restaurant and reviewer types to ensure that various groups of restaurant types (high and low end; chain and independent) and different types of reviewers (elite and non-elite; by experience on Yelp) are represented in the data. This allows us to study how ratings and missing attributes differ by the types.

The sampling leaves us with 45,652 reviews from 2,704 businesses and 19,583 reviewers. As past restaurant reviews might impact current reviews, we incorporate restaurants' time-varying features (e.g., variance and mean of past reviews) by extracting all past reviews for the restaurants in our stratified sample. The full dataset (including all past reviews for restaurants in our sample) contains 250K reviews. We generate each review's time varying variables, including number of past reviews; mean and variance of past star rating; and mean and variance of past attribute ratings.

[16]The restriction of 5 or more reviews also allows us to eliminate human or bot-generated fake reviews, which are mostly generated by users with one or only a few number of reviews. Luca and Zervas (2016) document that a larger number of reviews by a Yelp user is negatively correlated to the probability of his reviews getting filtered as spam by Yelp.

Electronic copy available at: https://ssrn.com/abstract=3395012

Table 7a compares the descriptive characteristics of the full data and our final sample consisting of 45,652 reviews. The mean and median number of reviews per reviewer in our sample is slightly higher than the population (due to stratification). However, the reviewers in our sample are fairly similar to the population in terms of average star rating, experience and length of reviews. Table 7b provides the number of businesses, reviews and the summary of star rating by a restaurant's price range, chain/independent Our sample has almost an equal mix of chain and independent restaurants but independent restaurants get more reviews with higher ratings on average. Low-end and high-end restaurants do not show much difference in terms of average star rating.

[Insert Tables 7a and 7b here]

Descriptive Evidence on Attribute Rating Behavior

We now present descriptive evidence on potential drivers of reviewer's rating behavior to motivate the choices and assumptions we make in the structural model. We first look at the impact of observable reviewer (e.g., Elite status[17]) and restaurant characteristics (e.g., price range, chain restaurant[18]) on the distribution of attribute ratings and attribute *missingness*. Figure 4a shows differences between the rating behaviors of Elites and Non-Elites. X-axis represents each rating or missing indicator. Elites tend to give more moderate ratings (3 and 4 stars) whereas Non-Elites give more extreme ratings (2 and 5 stars) across attributes. More importantly, Non-Elites tend to *miss* more attributes in their reviews (especially *ambiance*). Such differences suggest Elites might have different motivation to give ratings than Non-Elites.

Figures 4b and 4c show how rating behavior differs across low-end ($ and $$ on Yelp, indicating $\leq$$30 per person) and high-end ($$$ and $$$$ on Yelp; $>$$30 per person) restaurants and across chain and independent restaurants respectively. On average, high-end restaurant reviews have more attributes (less *missing*) except for *location* which is mentioned more in low-end restaurant reviews. The ratings are generally more positive for high-end restaurants. Chain reviews

[17]Elite reviewers receive an *Elite* badge that is displayed on their profile. They also get invited to special events. Most other observable characteristics are highly correlated with Elite, for e.g., elites are generally more experienced and have more friends.

[18]We identify restaurants as chains if they have multiple stores by the same name owned by a single firm.

Electronic copy available at: https://ssrn.com/abstract=3395012

tend to talk more about *service* and *location*, while reviews of independents talk about other attributes. The reviews for chains generally get an average of (3-star) or below on attributes whereas independents receive more 4 and 5-star attribute ratings.

Beyond the clear differences in attribute rating behavior (silence and valence) based on observables of restaurants and reviewers, the mapping from experience utilty to attribute ratings and attribute ratings to weights to obtain an overall rating can vary due to a variety of unobservables. To accommodate this, we allow for unobserved heterogeneity on these in our structural model.

RESULTS

We describe the results in five parts: (i) attribute sentiment classification performance of various text mining methods; (ii) estimates of the structural model of rating behavior; (iii) drivers of attribute silence; (iv) validation; and (v) the impact of correcting for attribute silence.

Attribute Sentiment Classification

We report the performance on attribute sentiment classification in three parts: (1) Overall classification accuracy; (2) Classification accuracy on "hard" sentene types; (3) polarity and attribute classification.

Overall Classification Accuracy. We begin by reporting the performance of the various models in terms of attribute and sentiment classification accuracy on the test dataset described earlier in the data section. The lexicon based method that relies on carefully crafted rules and human-tagged lexicons performs better than most supervised machine learning algorithms and is as good as the convolutional-LSTM in the attribute classification task. This is because this task is relatively unambiguous and the lexicons are constructed specific to the domain of restaurant reviews. However, this method does very poorly in the more complex 5-grained sentiment analysis task. Among supervised algorithms, Support Vector Machines (SVM) do better than most of the other classifiers in both attribute and sentiment classification tasks. This is in line with past literature that has shown that SVMs are the best Machine Learning based text classifiers. The network with only

Electronic copy available at: https://ssrn.com/abstract=3395012

convolutional layer just matches the performance of the SVM. However, the convolutional-LSTM does better than all methods in both attribute and sentiment classification tasks. The accuracy of the convolutional-LSTM in the task of 5-level sentiment classification is 50%—lower than state of art accuracy 56% reported in (Brahma 2018), but on a different dataset for which we do not know the differential mix of "hard" versus "easy" sentences in the corpus. Further, they also do not provide metrics like confusion matrices, which helps assess other dimensions of classification accuracy.[19]

[Insert Table 8a here]

The convolutional-LSTM model with self-trained embeddings does slightly better than the one using pre-trained Glove embeddings both in terms of attribute and sentiment accuracy. This could be attributed to the slightly more relevant vocabulary generated when word vectors are trained from scratch on a specific corpus as shown in Table A3 in online appendix.

[Insert Table A3 here]

Classification Accuracy on Hard sentence types. To develop some intuition behind what drives the performance accuracy of these models, we test these models on simple and various types of hard sentences. We sampled 100 sentences of each type from the test dataset. Table 8c reports the comparative performance of the deep learning models, the best supervised machine learning model (SVM) and the lexicon method. As expected, the hybrid convolutional-LSTM performs better than most other models in all of these tough classification scenarios and especially in classifying scattered sentiment in long sentences. Interestingly, the convolutional-LSTM model does significantly better on simple sentences as well.

[19] As an aside, we note that nlpprogress website which tracks state of the art (SOTA) for NLP tasks reports 72% accuracy using Yelp data as of 2019 for the 5 level sentiment task at the *review document* level. This is of course different from our 5 level *sentence level* sentiment task. But as a point of comparison, our model's performance for this document level task is 70%— comparable to the previous SOTA paper from 2017 (e.g., Johnson and Zhang 2017). Interestingly, we use a much smaller training data to achieve the same accuracy. While we make no claims in terms of being state of the art in terms of accuracy, we note that our classification results are in the ball park of "good" models. Our focus is on the performance on "hard" sentences.

Electronic copy available at: https://ssrn.com/abstract=3395012

[Insert Table 8c here]

Polarity and Attribute Classification. As we mentioned in the section *Performance Measures*, though accuracy is a first-order metric for hard problems like granular sentiment detection, we need other measures to refine model choice; especially among models with similar accuracy scores. Table 8d shows that the convolutional-LSTM model using Glove pre-trained embedding is slightly better than the one using self-trained embedding (though the overall accuracy is higher for the latter) because it preserves polarity better i.e. it mostly mis-classifies within the granular sentiment classes (positive, negative, neutral) and thus has lower polarity reversal.

Table 8e assesses attribute classification accuracy. We find that both the convolutional-LSTM based attribute classifiers using GloVe and self-trained embeddings do a fairly good job in classifying attributes across classes. Further, their performance is not driven simply by getting high-frequency classes like food right.

[Insert Tables 8d, 8e here]

Structural Model Estimates

Overall, we find a three segment model fits best.[20] Segment 1 the smallest segment, constitutes about 9% of the market. Segment 2, the largest segment accounts for 59% of the market, while Segment 3 constitutes 32% of the market.

Ordinal Logit Model. The estimates of the ordinal logit model that maps latent utility to attribute ratings is presented in two parts. Table 9a presents the mapping between restaurant observables and true latent attribute level experience. As expected, restaurants with higher ratings have overall higher latent utility, chains have lower latent utility, and prices reduce latent utility. The thresholds $C_s(s \in 2,3,4,5)$— the cutoff between score $s-1$ and s of the ordinal logit for each of the three latent segments are shown in Figure 6. As expected, these thresholds are monotone and increasing in rating scale, but nonlinear. Getting higher score requires higher-quality experience across attributes as expected, but the marginal satisfaction required for each score is different

[20] We assessed fit based on BIC for two, three and four segment models.

Electronic copy available at: https://ssrn.com/abstract=3395012

across attributes, scores and reviewer segments. It should be noted that even though the thresholds often appear parallel, its implications for probability of a given rating for a segment is highly nonlinear and therefore heterogeneous. This is because there is much higher density in the middle than at the extremes.

[Insert Table 9a and Figure 6 here]

Overall Rating Regression. The weights on the attribute ratings that impact overall rating for the three latent segments are presented in Table 9b. Note for ease of interpretation, the weights reported have been normalized such as the sum of the weights add to 1. Also, note that the model was estimated without normalization and all coefficients were estimated as positive.

[Insert Table 9b here]

Segment 1, the smallest at 9%, places the most importance on food in terms of their overall ratings. Segment 2, the largest at 59% cares not only about food, but also service. In contrast, for Segment 3, with 32% of reviewers, ratings are driven mostly about price and location.

Segment Interpretation. Finally, we report the descriptive statistics of each segment in Table 10 to aid interpretation. The smallest segment 1 (9% of reviewers) consists of 65% elites, writes most often and contributes double their share in reviews (18%). They write the longest reviews, and include the most number of attributes. They tend to write earlier than others on average. They tend to be harsher than the average rating of the restaurants and have relatively low variance of ratings. Given the high percentage of elites, greater frequency, and more comprehensive and longer reviews, we name them as "*status-seeking regulars*."

In contrast, Segment 3 accounting for 32% of reviewers has no elites, writes least frequently, contributing only 24% of reviews. The reviewers write the shortest reviews and include the fewest number of attributes. They tend to write at later stages after others have provided their reviews. They generally tend to be more generous in their overall ratings. Interestingly, they also have the highest variance in their reviews, though they visit restaurants with high ratings and lower variance.

Electronic copy available at: https://ssrn.com/abstract=3395012

We call them the ‘‘*emotive irregulars*,” given their lower frequency, and limited contributions in text reviews. They tend to offer either very positive or relatively negative reviews.

Finally, the largest segment 2 with 59% of the reviewers has only 26% elites. The reviewers are in the middle between Segment 1 and 3 in rating behaviors. They write fewer, shorter reviews and include fewer attributes than segment 1, but more than Segment 3. Their ratings are very similar to the average of the restaurant ratings. We call these reviewers as the “*altruistic mass*,” the bulk of the Yelp reviewing community, who write reviews diligently, but with little expectation of rewards or merely wanting their voice to be heard.

[Insert Table 10 here]

Drivers of Attribute Silence (Missingness)

With the estimates of the structural model, we now interpret attribute silence of each reviewer segment. We conjectured three plausible reasons driving attribute silence: (i) informativeness; (ii) attribute importance; and (iii) need to praise/vent. We assess each of these conjectures in turn.

Informativeness. Table 11 reports a logistic regression result with attribute presence as the DV. As conjectured, we expect experience attributes (food, service, ambiance) to be written more often than search attributes like price (which is a major component of value) and location. Further, if experience/search attribute is the driver of missingness, food and service should be missing more often at chains than at other restaurants. In addition to the attributes, the explanatory variables are positive and negative deviations in past attribute rating against predicted ratings; and the variance of the past attribute ratings.

The higher positive attribute coefficients (for food, service, ambiance) relative to the normalized location coefficient of zero, and value support our conjecture that reviewers write more often on experience attributes and tend to be more silent on search attributes which can be discovered easily on the site. Further, as expected, variance has a positive coefficient, supporting our hypothesis that attributes are more likely to be mentioned when opinions around that restaurant is not settled. Interestingly, for deviations, negative deviations induce the attribute to be mentioned, but

Electronic copy available at: https://ssrn.com/abstract=3395012

vice versa for positive deviations. This is the case across all segments. Thus there is overall support for the informativeness conjecture. A subtle point from the results is that people are more likely to share information about unmet expectations (negative deviations) than positive deviations.

[Insert Table 11 here]

Attribute Importance. First, we compare the probability of missing attribute by segment in the bottom panel of Table 10 with the attribute importance weights of the three segments reported in Table 9b. Food and service (and to a lesser extant ambiance) have the lowest rating of missing. Food, service and ambiance also have among the highest impact on overall ratings for Segments 1 and 2. But for segment 3, even though food and service do not drive overall ratings, they still are the most written about attributes. Similarly, even though value and location impact overall rating for Segment 3 these are still the most missing attributes in text reviews. Thus there is not a clear pattern that attribute missingness is driven by the importance of that attribute. To test this formally, we conducted a logistic regression with attribute presence as the DV and attribute importance as an explanatory variable along with additional control variables. Interestingly, we do not find a significant positive effect on attribute importance. In fact, the regression results show a consistent negative effect. Thus our results question the conventional wisdom, and the implicit assumption underlying many topic models, that the frequency of occurrence of topics is implicitly assumed to be related to its importance. However, we temper our conclusion around importance, because food and service which are most present may also be the most important in driving the decision to visit a restaurant, but our estimated attribute importance is conditional on visit.

[Insert Table 12 here]

Praise/Vent Need. As discussed earlier, we report the missing odds as the ratio π_{gk}^{s} defined as $\frac{Pr(A_{gk}=s|M_{gk}=1)}{Pr(A_{gk}=s|M_{gk}=0)}$ for each attribute by segment as a function of predicted sentiment level in Figure 6. The patterns of attribute silence differ by attributes and by segment. For food, service and ambiance, all three segments tend to be more silent when they are dissatisfied, and write more

Electronic copy available at: https://ssrn.com/abstract=3395012

when they are satisfied. However, segment 3 which places the most importance on location and value is more likely to write about these attributes when they are dissatisfied. This seems consistent with our label for them —as emotive irregulars. They don't write often, but they write when they are very satisfied with food, service and ambiance, but dissatisfied with value and location. This may also explain the higher variance in their overall ratings.

[Insert Figure 6 here]

In summary, the information value of reviews play a significant role in the motivation to write about attributes across all segments. We also found the motivation to both praise good performance and vent about bad performance, but this varied across segments and attributes. For staple features like food, service and ambiance, all three segments are more likely to write when satisfied and less likely to write when dissatisfied. Overall, this might explain in general why reviews tend to be skewed to be more positive on rating sites—if this also translates to selection into who writes reviews. However segment 3 is likely to vent more when dissatisfied about two attributes that drive its ratings—value and location. The lack of a strong link between importance and mentions in reviews of attributes suggests that online reviews may not be as complete a source of topic and need identification as previously believed. However, we note that this could be because our attribute importance estimate are conditional on visit to restaurants, and may not account for its importance in decision to visit the restaurant. At the very least, our results suggest that we might want to be circumspect in the use of frequency of mentions as a proxy for benefit or need importance and explore this issue in future research.

Validation of Imputation

We validate our model-based imputation approach in Table 13 by assessing the ability to predict attribute ratings on a holdout sample, relative to no segmentation, where we assume reviewers have homogeneous rating styles, and ad-hoc imputation approaches, where reviewers who missed attribute ratings experienced average (score 3) or very low (score 1) or very high (score 5) level

Electronic copy available at: https://ssrn.com/abstract=3395012

of satisfaction. To be specific, we compare the predicted attribute ratings vs. observed rating if an attribute rating is present on hold-out sample (10% of the observations). The overall RMSE across all attributes is lower for our model relative to the benchmark models. Even when the RMSE is compared by attribute, we find that our model does better on all attributes, except location, where a uniform imputation of 3 can get slightly better prediction. Given the large share of missing data for location, the model identification was the weakest for this attribute. However, for all the other attributes the imputation from the model indeed does better.

[Insert Table 13 here]

Correction for Attribute Silence in Attribute Ratings

We illustrate how correcting for attribute silence through imputation at the individual review level can impact overall attribute rating for a restaurant. We see that correction for missing attributes has significant impact on attributes that are missing more frequently: value and location in general, and food for chain restaurants. The correction could be either upward or downward depending on attribute, restaurant type and reviewer type. For example, at an independent restaurant in Phoenix where most reviewers are found to remain silent about service at higher satisfaction levels, observed service ratings are lower than actual service ratings after imputing for missing attribute ratings. Then, correction results in higher service ratings than observed ratings (Figure 7a). Food and ambiance scores barely change, and value and location scores slightly go up after imputation for this restaurant. In Figure 7b, we illustrate a chain restaurant in Las Vegas where many of the reviewers miss food and location attributes, when satisfied, and miss value rating when dissatisfied. Here food and location scores to go up and value score to go down. Overall this shows that our imputation approach based on restaurant observables, rater observable and unobservable heterogeneity is extremely flexible in its imputations and the ability to correct for missing attribute ratings.

[Insert Table 14 and Figure 7 here]

Electronic copy available at: https://ssrn.com/abstract=3395012

CONCLUSION

The paper addresses the general problem of using unstructured text data to generate quantifiable market feedback typically obtained through surveys; the specific application is to use restaurant reviews to generate attribute level ratings of restaurants. The paper addresses two novel and challenging problems around online text reviews: (i) convert text into *fine-grained numerical sentiment scores* on pre-specified attributes (e.g., food, service) by accounting for language structure; and (ii) accounting for attribute silence in attribute sentiment scoring. For the first problem, the it uses a deep learning convolution-LSTM model that exploits the spatial and sequential structure of language to improve sentiment classification, especially on known types of "hard" sentences in NLP. For addressing attribute silence, the paper develops and estimates a structural model of reviewer rating behavior that takes into account the data generating process to develop a model-based imputation procedure to address attribute silence. Overall, the paper illustrates the value of combining "engineering" thinking underlying machine learning approaches with "social science" thinking from econometrics to answer novel marketing questions.

Substantively, the paper identified three segments of reviewers—the smallest but most active reviewers ("Status Seeking Regulars,") the largest segment ("Altruistic Mass,") who review without reward expectations, and "Emotive Irregulars," who review infrequently, but write about attributes they are extremely satisfied or dissatisfied. Our insights around attribute silence in reviews shows that informativeness and need to praise/vent drive more of the writing than the importance of the attribute. Not only does this contribute to the literature on why people engage in online word of mouth (Berger 2014), it also has implications for using reviews as a source of data for needs/benefits identification. In particular, contrary to conventional wisdom, the frequency of mentions of a benefit or a topic may not necessarily be a proxy of its importance.

We conclude with a discussion of some suggestions for future research. First from the machine learning perspective, the research around improving performance on "hard" sentences needs to be pursued to further improve accuracy; while the performance improved for all of the hard sentence

Electronic copy available at: https://ssrn.com/abstract=3395012

types there is more room for improvement. It would be useful to consider how recent methods such as BERT or GPT can improve on the fine-grained sentiment scoring problem for ''hard" sentences. From the substantive/econometric perspective, it would be useful to more systematically understand the drivers of attribute silence. While our current results offer suggestive evidence for our conjectures, a more systematic causal investigation of the attribute level motivations can further enrich the literature on the drivers of WOM. It would also be worth combining our content analysis at the attribute level with work on fake reviews/review shading to get a richer understanding of how to correct for these issues in tracking WOM.

Electronic copy available at: https://ssrn.com/abstract=3395012

REFERENCES

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

Athey S, Bayati M, Doudchenko N, Imbens G, Khosravi K (2018) Matrix completion methods for causal panel data models. Technical report, National Bureau of Economic Research.

Bengio Y, CA M (2015) Rmsprop and equilibrated adaptive learning rates for nonconvex optimization. *corr abs/1502.04390* .

Berger J (2014) Word of mouth and interpersonal communication: A review and directions for future research. *Journal of consumer psychology* 24(4):586–607.

Berger J, Sorensen AT, Rasmussen SJ (2010) Positive effects of negative publicity: When negative reviews increase sales. *Marketing Science* 29(5):815–827.

Brahma S (2018) Improved sentence modeling using suffix bidirectional lstm. *arXiv preprint arXiv:1805.07340* .

Büschken J, Allenby GM (2016) Sentence-based text analysis for customer reviews. *Marketing Science* 35(6):953–975.

Chevalier JA, Mayzlin D (2006) The effect of word of mouth on sales: Online book reviews. *Journal of marketing research* 43(3):345–354.

Dhar V, Chang EA (2009) Does chatter matter? the impact of user-generated content on music sales. *Journal of Interactive Marketing* 23(4):300–307.

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

Duan W, Gu B, Whinston AB (2008) Do online reviews matter?—an empirical investigation of panel data. *Decision support systems* 45(4):1007–1016.

Feldman R (2013) Techniques and applications for sentiment analysis. *Communications of the ACM* 56(4):82–89.

Electronic copy available at: https://ssrn.com/abstract=3395012

Ganu G, Elhadad N, Marian A (2009) Beyond the stars: improving rating predictions using review text content. *WebDB*, volume 9, 1–6 (Citeseer).

Ghose A, Ipeirotis PG (2007) Designing novel review ranking systems: predicting the usefulness and impact of reviews. *Proceedings of the ninth international conference on Electronic commerce*, 303–310 (ACM).

Godes D, Mayzlin D (2004) Using online conversations to study word-of-mouth communication. *Marketing science* 23(4):545–560.

Goodfellow IJ, Bengio Y, Courville AC (2016) *Deep Learning*. Adaptive computation and machine learning (MIT Press).

Gurney N, Loewenstein G (2019) Filling in the blanks: What restaurant patrons assume about missing sanitation inspection grades. *Journal of Public Policy & Marketing* 0743915619875419.

Harris ZS (1954) Distributional structure. *Word* 10(2-3):146–162.

Hochreiter S, Schmidhuber J (1997) Long short-term memory. *Neural Comput.* 9(8):1735–1780, ISSN 0899-7667.

Hollenbeck B (2018) Online reputation mechanisms and the decreasing value of chain affiliation. *Journal of Marketing Research* 55(5):636–654.

Huang JL, Liu M, Bowling NA (2015) Insufficient effort responding: Examining an insidious confound in survey data. *Journal of Applied Psychology* 100(3):828.

Joachims T (2002) *Learning to classify text using support vector machines*, volume 668 (Springer Science & Business Media).

Johnson R, Zhang T (2017) Deep pyramid convolutional neural networks for text categorization. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Vol 1: Long Papers)*.

Kim Y (2014) Convolutional neural networks for sentence classification. *arXiv:1408.5882* .

Krosnick JA (1991) Response strategies for coping with the cognitive demands of attitude measures in surveys. *Applied cognitive psychology* 5(3):213–236.

Electronic copy available at: https://ssrn.com/abstract=3395012

Lee TY, Bradlow ET (2011) Automated marketing research using online customer reviews. *Journal of Marketing Research* 48(5):881–894.

Li X, Hitt LM (2008) Self-selection and information role of online product reviews. *Information Systems Research* 19(4):456–474.

Li Y, Lu SF, Lu LX (2019) Do yelp reviews influence consumer choice in the presence of government ratings? evidence from us nursing homes. *Evidence from US Nursing Homes (October 1, 2019)* .

Little RJ, Rubin DB (2019) *Statistical analysis with missing data*, volume 793 (John Wiley & Sons).

Liu B, et al. (2010) Sentiment analysis and subjectivity. *Handbook of natural language processing* 2(2010):627–666.

Liu X, Lee D, Srinivasan K (2019) Large-scale cross-category analysis of consumer review content on sales conversion leveraging deep learning. *Journal of Marketing Research* 56(6):918–943.

Luca M (2016) Reviews, reputation, and revenue: The case of yelp. com .

Luca M, Vats S (2013) Digitizing doctor demand: The impact of online reviews on doctor choice. *Cambridge, MA: Harvard Business School* .

Luca M, Zervas G (2016) Fake it till you make it: Reputation, competition, and yelp review fraud. *Management Science* 62(12):3412–3427.

Mayzlin D, Dover Y, Chevalier J (2014) Promotional reviews: An empirical investigation of online review manipulation. *American Economic Review* 104(8):2421–55.

Mikolov T, Sutskever I, Chen K, Corrado GS, Dean J (2013) Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 3111–3119.

Mittal V, Katrichis JM, Kumar P (2001) Attribute performance and customer satisfaction over time: evidence from two field studies. *Journal of Services Marketing* .

Electronic copy available at: https://ssrn.com/abstract=3395012

Mittal V, Kumar P, Tsiros M (1999) Attribute-level performance, satisfaction, and behavioral intentions over time: a consumption-system approach. *Journal of Marketing* 63(2):88–101.

Mohan K, Pearl J (2018) Graphical models for processing missing data. *arXiv preprint arXiv:1801.03583* .

Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines. *Proceedings of the 27th International Conference on International Conference on Machine Learning*, 807–814, ICML'10, ISBN 978-1-60558-907-7.

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

Nielsen FÅ (2011) A new anew: Evaluation of a word list for sentiment analysis in microblogs. *arXiv preprint arXiv:1103.2903* .

Onishi H, Manchanda P (2012) Marketing activity, blogging and sales. *International Journal of Research in Marketing* 29(3):221–234.

Pak A, Paroubek P (2010) Twitter as a corpus for sentiment analysis and opinion mining. *LREc*, volume 10.

Peloza J, Ye C, Montford WJ (2015) When companies do good, are their products good for you? how corporate social responsibility creates a health halo. *Journal of Public Policy & Marketing* 34(1):19–31.

Pennington J, Socher R, Manning C (2014) Glove: Global vectors for word representation. *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Puranam D, Narayan V, Kadiyali V (2017) The effect of calorie posting regulation on consumer opinion: a flexible latent dirichlet allocation model with informative priors. *Marketing Science* 36(5):726–746.

Rubin DB (1976) Inference and missing data. *Biometrika* 63(3):581–592.

Electronic copy available at: https://ssrn.com/abstract=3395012

Schouten K, Frasincar F (2015) Survey on aspect-level sentiment analysis. *IEEE Transactions on Knowledge and Data Engineering* 28(3):813–830.

Slovic P, MacPhillamy D (1974) Dimensional commensurability and cue utilization in comparative judgment. *Organizational Behavior and Human Performance* 11(2):172–194.

Socher R, Perelygin A, Wu J, Chuang J, Manning CD, Ng A, Potts C (2013) Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of the 2013 conference on empirical methods in natural language processing*, 1631–1642.

Taboada M, Brooke J, Tofiloski M, Voll K, Stede M (2011) Lexicon-based methods for sentiment analysis. *Computational linguistics* 37(2):267–307.

Timoshenko A, Hauser JR (2018) Identifying customer needs from user-generated content. *Marketing Science (Forthcoming)* .

Tirunillai S, Tellis GJ (2014) Mining marketing meaning from online chatter: Strategic brand analysis of big data using latent dirichlet allocation. *Journal of Marketing Research* 51(4):463–479.

Wang H, Lu Y, Zhai C (2010) Latent aspect rating analysis on review text data: a rating regression approach. *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 783–792 (ACM).

Wang J, Yu LC, Lai KR, Zhang X (2016) Dimensional sentiment analysis using a regional cnn-lstm model. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Vol 2: Short Papers)*, 225–230.

Xu X (2019) Examining the relevance of online customer textual reviews on hotels' product and service attributes. *Journal of Hospitality & Tourism Research* 43(1):141–163.

Zhou C, Sun C, Liu Z, Lau F (2015) A c-lstm neural network for text classification. *preprint arXiv:1511.08630* .

Zhu F, Zhang X (2010) Impact of online consumer reviews on sales: The moderating role of product and consumer characteristics. *Journal of marketing* 74(2):133–148.

Electronic copy available at: https://ssrn.com/abstract=3395012

Tables and Figures

Table 1: Examples of "hard" sentences for attribute sentiment scoring

Type	**Example**
Negations and Sentiment Degree	Pizza is *good* > Pizza is *not that good* > Pizza is *not at all good*
Long sentences and Scattered Sentiments	OK, in fact *good*, to start with but *kept getting worse* and wait staff were *unapologetic* but manager *saved the night*.
Contrastive Conjunctions	Despite the *creativity in the menu*, *execution was a disappointment*
Implied Sentiments	The place is a *treasure* if *only you are lucky to be there on the right day*

Table 2: Most Relevant Marketing Literature on Text Analytics

Paper	Analysis Unit	Sentiment Analysis (Y/N)	Sentiment Granularity	Method	Performance Metric	Attribute Silence (Y/N)
Godes and Mayzlin (2004) & Chevalier and Mayzlin (2006)	Document	NA	NA	No Text Mining	NA	N
Lee and Bradlow (2011)	Document	N	NA	Bag of Words	Overall	N
Archak et al. (2011)	Document	Y/N	Binary	Semi-supervised	Overall	N
Netzer et al. (2012)	Document	N	NA	Lexical Networks	Overall	N
Tirunillai and Tellis (2014)	Document	Y/N	Binary	LDA	Overall	N
Timoshenko and Hauser (2018)	Sentence	N	NA	CNN	Overall	N
Büschken and Allenby (2016)	Sentence	N	NA	Sentence LDA	Overall	N
Liu et al. (2019)	Document	Y	Binary	CNN, RNN, LSTM	Overall	N
This paper	Sentence	Y	5-level	Convolutional-LSTM	Overall & Hard Sentences	Y

Electronic copy available at: https://ssrn.com/abstract=3395012

Table 3: Algorithm for Attribute Sentiment Analysis

Algorithm : Derive Attribute scores from Review Text
Input : Review text *s: no of sentences, w_s: words in sentence s*
Step 0 : Choose relevant sentiment and attribute scale **Step 1:** Split review doc r_d into sentence vectors of s sentences using standard tokenizers **Step 2:** For all s sentences , repeat steps 3 through 7 **Step 3:** Pre-process the sentence to convert characters to lower-case, remove stop-words and punctuations **Step 4:** Pass one sentence at a time into an Attribute Sentiment Classifier *AS* **Step 5a :** *AS* classifies sentence into an aspect class based on its algorithm (lexicon, machine learning or deep learning) **Step 5b:** *AS* classifies sentence into a sentiment class based on its algorithm **Step 6a:** Attribute Score $\longrightarrow$ mean(attribute sentiment across all sentences) **Step 6b:** If an attribute is not mentioned in any sentence s, assign it a missing sentiment score

Table 4: Description of Datasets

Data	**Size**	**Criteria**	**Purpose**
Yelp Restaurant Corpus	1.2 Mn reviews	All restaurant reviews	Exploratory Analysis
Supervised Learning	2400 sentences	Balance of attribute and sentiment classes	Training/Testing Supervised Models
Stratified Sample	45,652 reviews	Business$\geq$20 reviews Mix of Business and Reviewer Types	Estimating Structural Model
Restaurant Panel	250,000 reviews	Restaurants in Stratified Sample	Deriving past review characteristics

Table 5: Class Balance: Attribute and Sentiment Classes (N: 2400)

Attribute			Sentiment		
Class	Training Data	Test Data	Class	Training Data	Test Data
Food	34%	37%	Negative	18%	25%
Service	21%	23%	Positive	35%	27%
Ambiance	14%	12%	Very Negative	11%	9%
Value	11%	10%	Very Positive	21%	27%
Location	4%	7%	Neutral	15%	11%

Table 6: Distribution of Sentence Types (N: 706)

	Positive	Neutral	Negative	
Overall	52%	12%	36%	
Simple	64%	53%	34%	52%
Implied	6%	5%	32%	15%
Contrastive	7%	20%	11%	10%
Long	26%	24%	28%	27%

Electronic copy available at: https://ssrn.com/abstract=3395012

Table 7a: Summary Statistics of Full Dataset vs. Sample

	Full			Sample		
Number of Reviews	1.2M			45,652		
Number of Reviewers	1.02M			19,583		
	Mean	Median	SD	Mean	Median	SD
Star Rating	3.7	3.8	1.09	3.6	3.76	0.92
Number of Reviews per Reviewer	24	5	82	25.15	17	23.2
Reviewer's Experience on Yelp	58	56	27.5	54.6	51.8	36.2
Review Length (number of characters)	1,109	599	732	709	498	670

Table 7b: Sample Summary Statistics by Restaurant Type

	All	By Price Range		By Chain	
		Low-end	High-end	Chain	Non-Chain
Number of Businesses	2,707	1,611	1,096	1,063	1,644
Number of Reviews	45,652	21,066	24586	10,528	35,124
Star Rating: Mean (SD)	3.5 (1.4)	3.4 (1.4)	3.6 (1.4)	2.8 (1.5)	3.7 (1.3)

Table 8a: Comparison of Text Mining Methods

Type	Method	Attribute accuracy	Sentiment accuracy	Building Effort	Scalability	Interpretability
Lexicon	Lexicon	68%	31%	High	Low	High
Machine Learning	SVM	60%	40%	Moderate	High	Low
	Naives Bayes	43%	39%			
	Logistic Regression	59%	41%			
Deep Learning	CNN	62%	41%	Moderate	High	Low
	LSTM	62%	40%			
	conv-LSTM (pre-trained)	**68**%	**47**%			
	conv-LSTM (self-trained)	**71**%	**50**%			

Electronic copy available at: https://ssrn.com/abstract=3395012

Table 8c: Performance on Hard Sentence Types

	Simple	Hard(Overall)	Scattered	Implied	Contrastive
Lexicon	46%	17%	17%	18%	16%
SVM	47%	19%	18%	20%	20%
CNN	44%	21%	22%	17%	24%
LSTM	46%	30%	37%	28%	25%
Convolutional-LSTM	52%	34%	41%	31%	28%

Table 8d: Polarity Reversal Confusion Matrix (Sentiment Analysis)

	CNN			Convolutional-LSTM (self trained)			Convolutional-LSTM(Glove 300)		
True Class	Negative	Neutral	Positive	Negative	Neutral	Positive	Negative	Neutral	Positive
Very Negative	**31%**	7%	51%	**47%**	6%	47%	**71%**	2%	24%
Negative	**33%**	11%	63%	**45%**	6%	49%	**64%**	4%	35%
Neutral	14%	37%	49%	16%	33%	51%	44%	18%	39%
Positive	14%	9%	**77%**	15%	6%	**80%**	31%	5%	**64%**
Very Positive	9%	10%	**81%**	21%	2%	**88%**	18%	5%	**77%**

Table 8e: Simple Confusion Matrix (Attribute Analysis)

	Convolutional-LSTM (self-trained)					Convolutional-LSTM (Glove 100)				
Predicted \ True	food	service	ambiance	value	location	food	service	ambiance	value	location
Food	**79%**	4%	2%	3%	2%	**75%**	6%	6%	3%	1%
Service	10%	**60%**	9%	5%	3%	7%	**76%**	8%	2%	0
Ambiance	8%	0	**58%**	3%	10%	2%	3%	**77%**	2%	2%
Value	10%	2	6%	**75%**	2%	8%	8%	6%	**74%**	4%
Location	8%	6%	11%	3%	**56%**	6%	14%	36%	3%	31%

Table 9a: Structural Model Estimates
Link between Restaurant characteristics and attribute latent utility

	food	service	ambiance	value	location
Biz price $$	0.171*	0.107**	0.002 ***	0.036***	0.035***
	(0.093)	(0.051)	(0.002)	(0.013)	(0.016)
Biz price $$$	-0.007	0.306***	0.255***	-0.115***	0.356
	(0.084)	(0.0779)	(0.041)	(0.030)	(0.065)
Biz price $$$$	0.100**	0.269**	0.338***	-0.219***	0.650***
	(0.046)	(0.109)	(0.053)	(0.059)	(0.085)
Biz chain	-0.388***	-0.013	-0.079 **	-0.225 ***	-0.351***
	(0.089)	(0.032)	(0.050)	(0.036)	(0.041)
Biz average stars	0.263***	0.395***	0.249***	0.317***	0.254***
	(0.051)	(0.038)	(0.038)	(0.018)	(0.017)
Previous reviews: average attribute rating	0.338***	0.166***	0.046*	0.053***	0.027***
	(0.063)	(0.036)	(0.031)	(0.016)	(0.014)
N	38630	34636	17305	16227	10463

Electronic copy available at: https://ssrn.com/abstract=3395012

Table 9b: Structural Model Estimates
Attribute weights (normalized to sum to 1) on Ratings

	Segment 1	Segment 2	Segment 3
Food	0.229	0.322	0.01
Service	0.217	0.173	0.00
Ambiance	0.154	0.180	0.01
Value	0.194	0.163	0.39
Location	0.206	0.162	0.59
Segment size (by Review)	18%	58%	24%
Segment size (by Reviewer)	9%	59%	32%

Table 10: Segment Characteristics

Characteristic	**Status-seeking Regulars** Mean (SD)	**Altruistic Mass** Mean (SD)	**Emotive Irregulars** Mean (SD)
% Elites	65%	26%	0%
Review Length (Chars)	889 (744)	677 (640)	349 (329)
No of Attributes	2.87 (1.1)	2.53 (1)	1.87 (0.83)
No of earlier reviews	22 (34.1)	24.2 (38)	36.7 (47.4)
Experience (Months)	33.6 (25.7)	24.6 (25.1)	16.9 (20)
Reviewer Rating	3.9 (0.4)	3.31 (1)	4.08 (1.2)
Business Rating	3.63 (0.7)	3.47 (1.1)	3.84 (0.7)
Proportion of Missing Attributes by Segment			
Food	0.10	0.18	0.22
Service	0.22	0.24	0.32
Ambiance	0.53	0.66	0.73
Value	0.58	0.62	0.89
Location	0.71	0.76	0.90

Table 11: Impact of Informativeness on Attribute Presence

	Dependent variable:			
	Attribute Presence			
	(1)	(2)	(3)	(4)
Intercept	0.544^{***}	0.580^{***}	0.555^{***}	0.183^{***}
Food	0.160^{***}	0.153^{***}	0.169^{***}	0.273^{***}
Service	0.134^{***}	0.130^{***}	0.134^{***}	0.245^{***}
Ambiance	-0.009^{**}	-0.035^{***}	0.017^{***}	0.063^{***}
Value	0.023^{***}	0.028^{***}	0.030^{***}	-0.020
Chain	0.098^{***}	0.104^{***}	0.078^{***}	0.079^{***}
Food × Chain	-0.112^{***}	-0.129^{***}	-0.077^{***}	-0.187^{***}
Service × Chain	-0.070^{***}	-0.065^{***}	-0.056^{***}	-0.103^{***}
Ambiance × Chain	-0.107^{***}	-0.101^{***}	-0.112^{***}	-0.070^{*}
Value × Chain	-0.074^{***}	-0.091^{***}	-0.067^{***}	0.027
Variance	0.061^{***}	0.061^{***}	0.059^{***}	0.068^{***}
Positive Difference	-0.368^{***}	-0.378^{***}	-0.366^{***}	-0.241^{***}
Negative Difference	0.087^{***}	0.077^{***}	0.086^{***}	0.167^{***}
N	136,600	67,255	60,422	8,923

Note: *p<0.1; **p<0.05; ***p<0.01
Positive Difference = $\|Actual\text{-}Own\|.I(Actual > Own)$,
Negative Difference = $\|Actual\text{-}Own\|.I(Actual < Own)$

Electronic copy available at: https://ssrn.com/abstract=3395012

Table 12: Impact of Importance on Attribute Presence

	Dependent variable:		
	Attribute Presence		
	(1)	(2)	(3)
Intercept	0.597***	0.319***	0.317***
Food		0.612***	0.577***
Service		0.492***	0.448***
Ambiance		0.136***	0.146***
Value		0.122***	0.157***
Importance	−0.146***	−0.132***	−0.127***
Importance × Food			0.084***
Importance × Service			0.134***
Importance × Ambiance			−0.034
Importance × Value			−0.087***
Observations	148,605	148,605	148,605
Log Likelihood	−106,713.200	−87,654.920	−87,518.190
Akaike Inf. Crit.	213,430.400	175,321.800	175,056.400

Note: *p<0.1; **p<0.05; ***p<0.01

Table 13: Model Fit: Root Mean Squared Error (RMSE) across Imputations

Attribute	Our method	No Heterogeneity	Fixed Imputation Scores		
			Score 1	Score 3	Score 5
Overall	0.689	0.932	1.879	0.778	1.211
Food	0.699	0.881	2.631	0.973	1.323
Service	0.891	0.967	2.304	0.939	1.587
Ambiance	0.578	1.169	1.768	0.711	0.960
Value	0.664	0.924	1.534	0.706	1.132
Location	0.613	0.717	1.161	0.564	1.055

Table 14: Impact of Imputation on Attribute Ratings

	Average Correction		% of corrections ≥ 0.5	
	Chain	Independent	Chain	Independent
Food	0.33	0.12	22%	1%
Service	0.24	0.32	13%	16%
Ambiance	0.83	0.62	83%	62%
Price	1.07	0.83	92%	91%
Location	1.29	1.16	92%	95%
N: 2719				

Electronic copy available at: https://ssrn.com/abstract=3395012

Figure 1: Sentiment Analysis Methods Evaluation

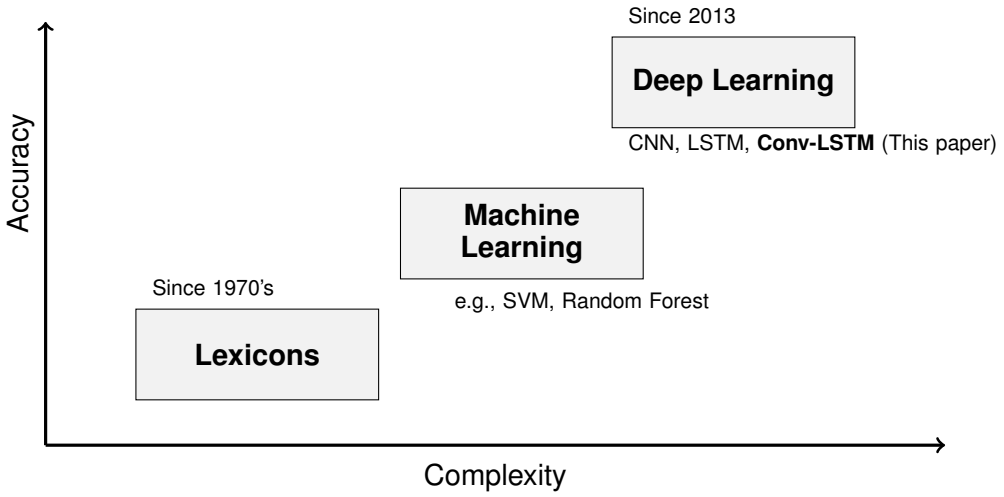
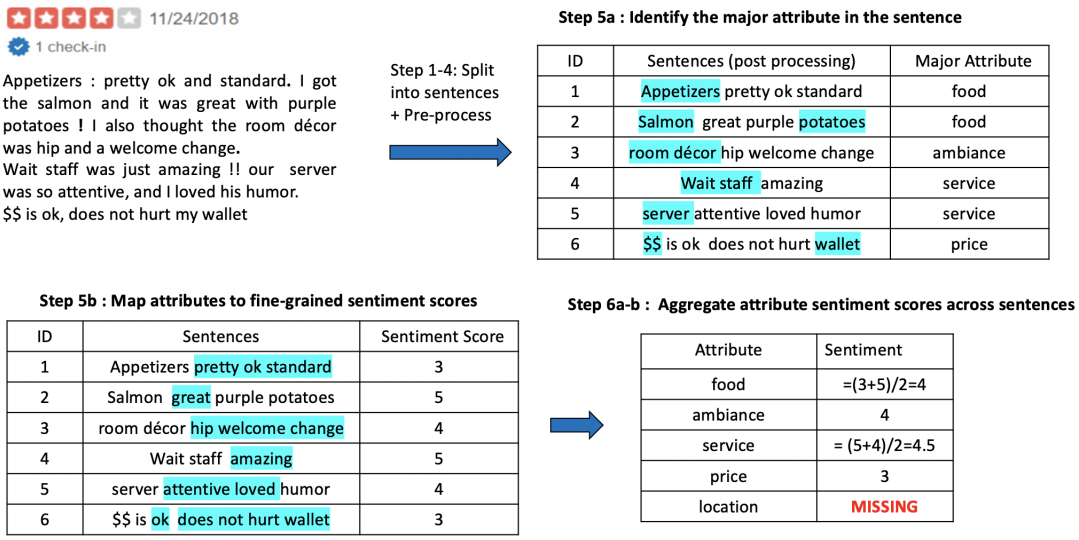
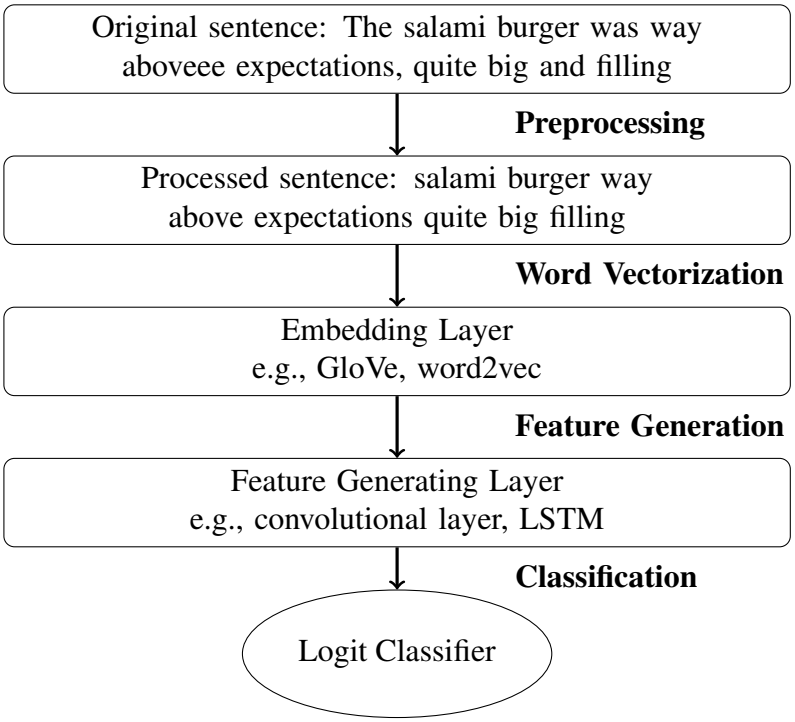


Figure 2: Illustration of Attribute-Level Sentiment Analysis



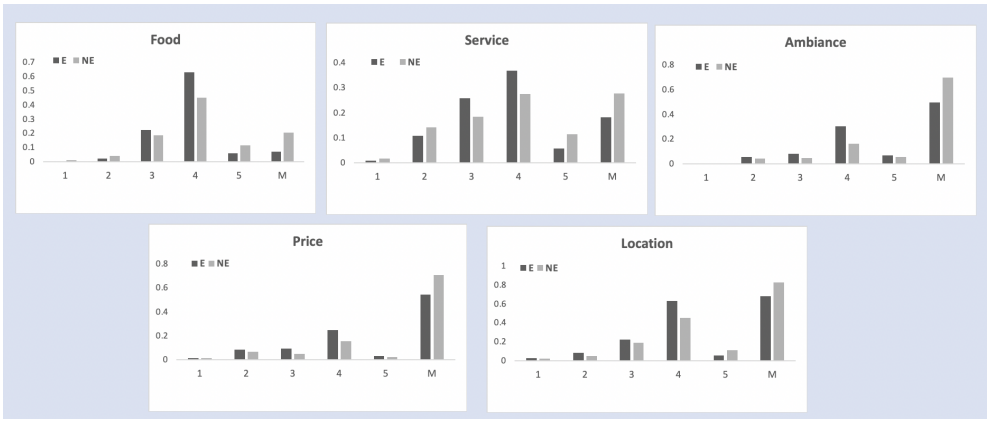
Electronic copy available at: https://ssrn.com/abstract=3395012

Figure 3: General Architecture of a Deep Learning Network for Text Classification

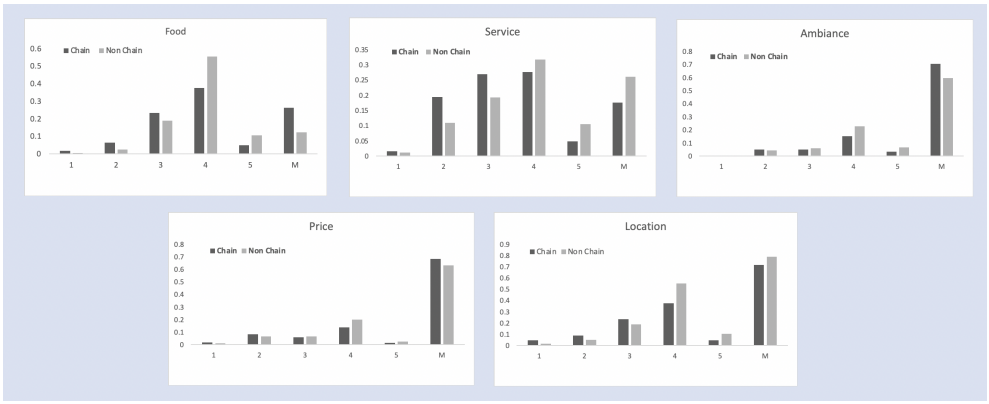


Electronic copy available at: https://ssrn.com/abstract=3395012

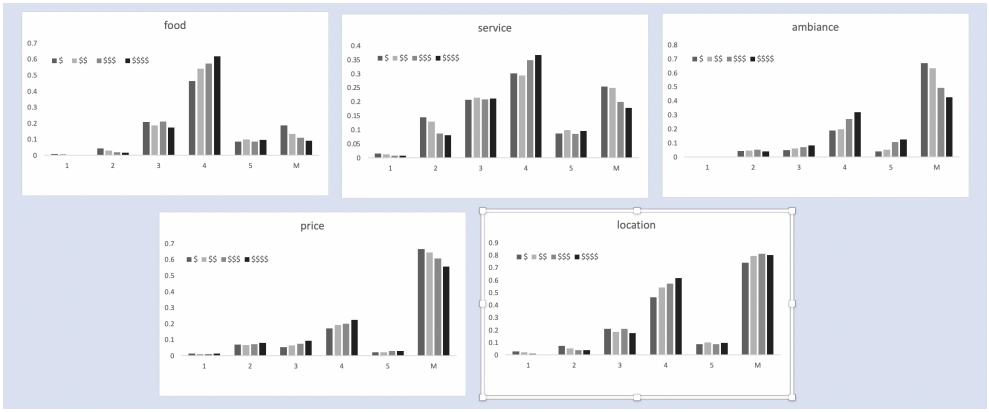
Figure 4: Differences based on Observable Characteristics



(a) Elites and Non -Elites



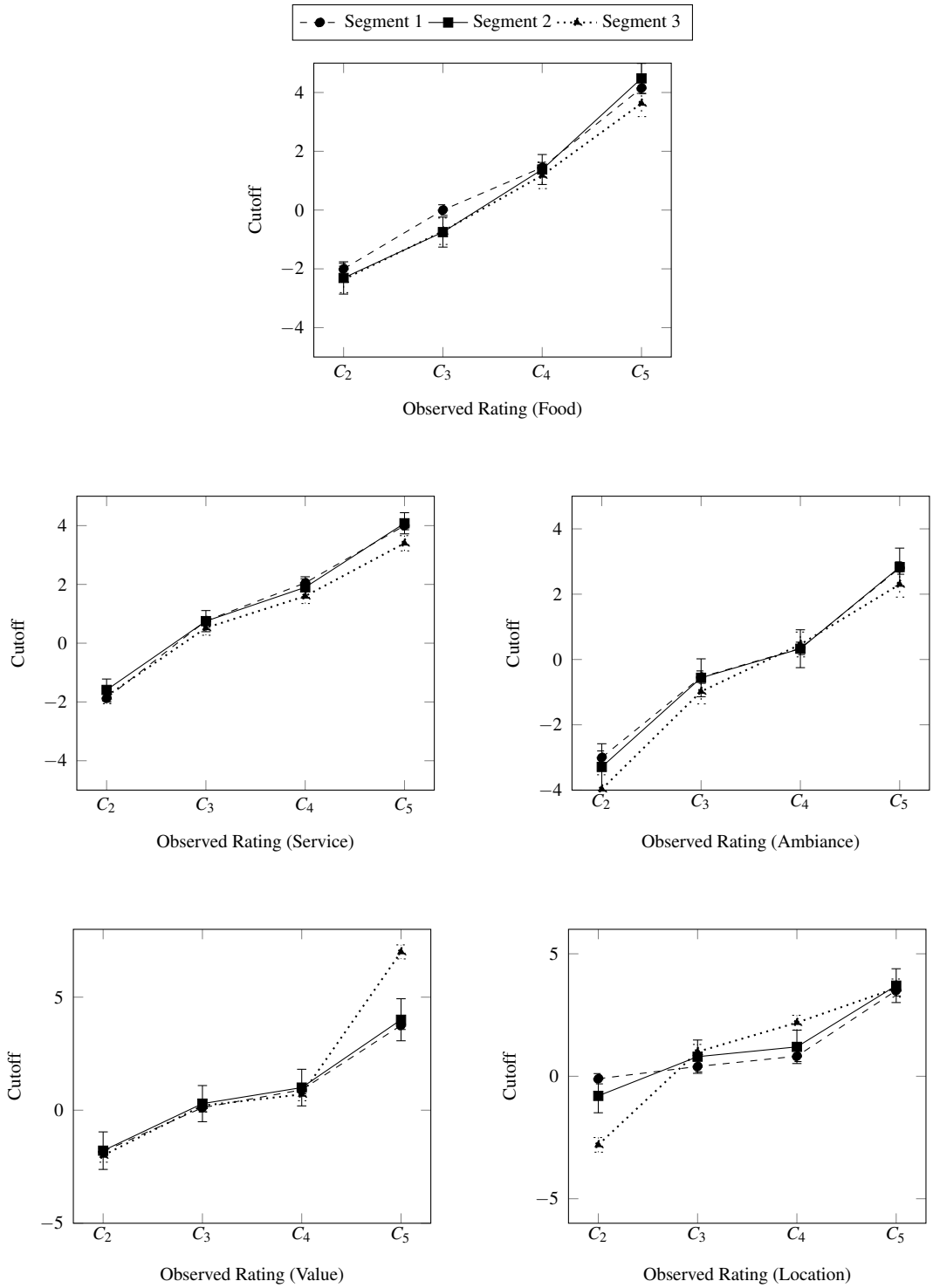
(b) Chains and Independent Restaurants



(c) Price Ranges (1-4)

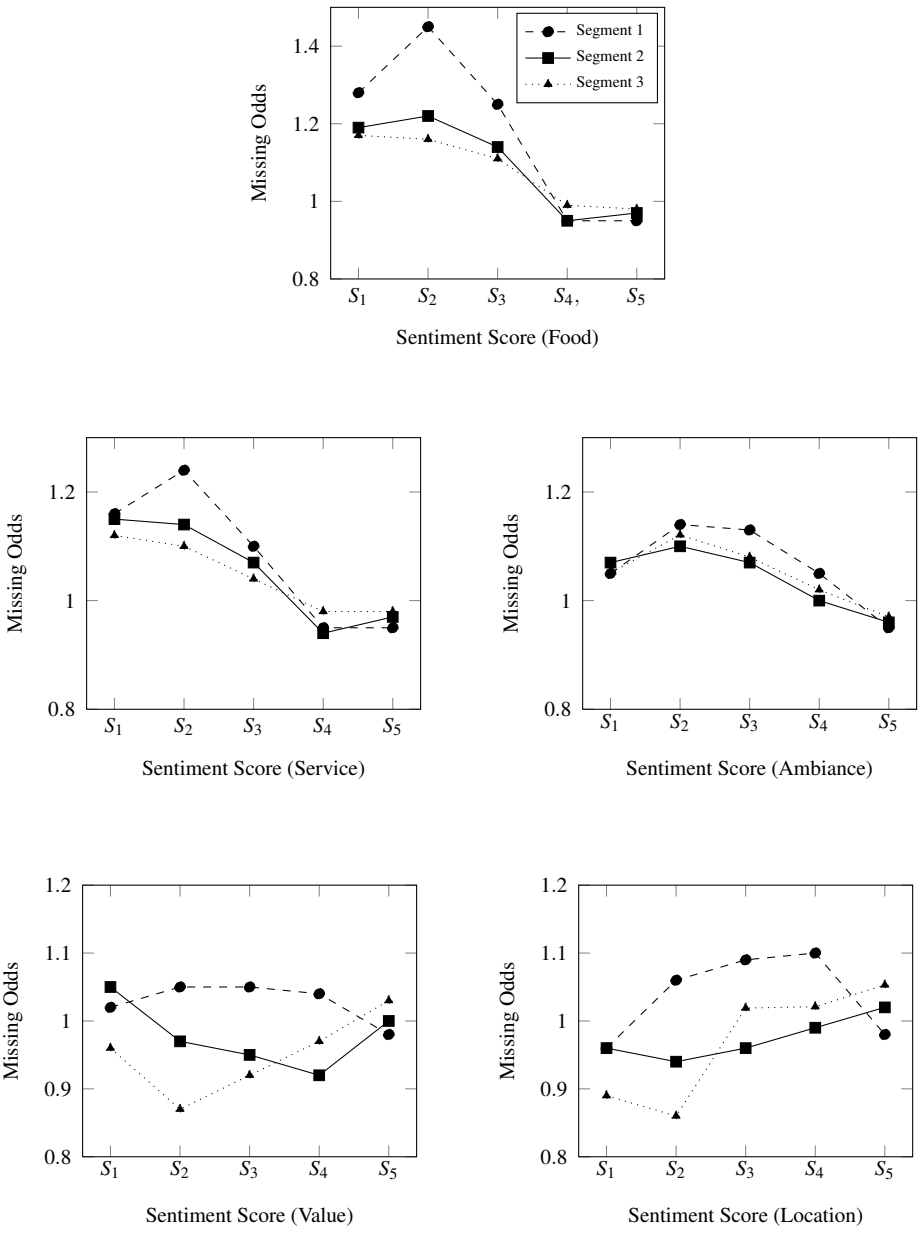
Electronic copy available at: https://ssrn.com/abstract=3395012

Figure 5: Structural Model Estimates: Attribute Level Thresholds of Latent Utility by Segment



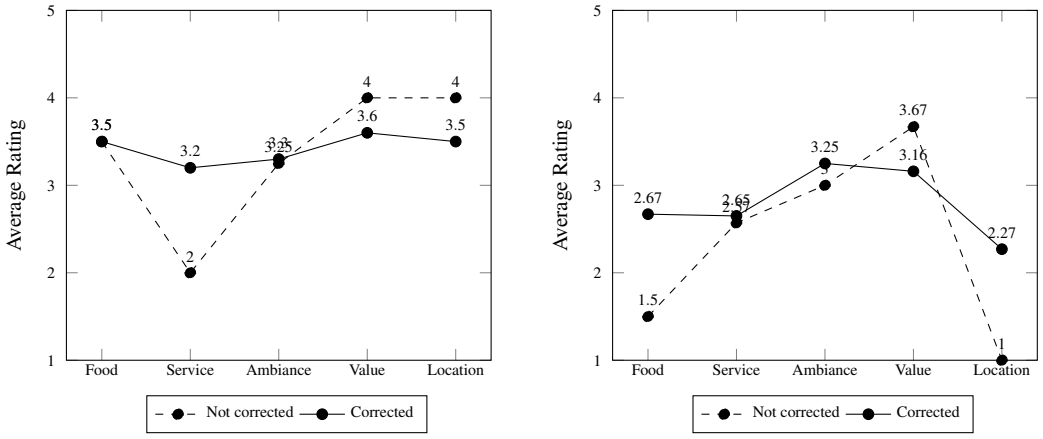
Electronic copy available at: https://ssrn.com/abstract=3395012

Figure 6: Odds(π) of attribute missing in reviews as a function of sentiment level



Electronic copy available at: https://ssrn.com/abstract=3395012

Figure 7: Change in Average Attribute Rating



(a) Independent restaurant in Phoenix

(b) Chain restaurant in Las Vegas

Electronic copy available at: https://ssrn.com/abstract=3395012

ONLINE APPENDIX

Mturk Experiment

In this section, we describe the Mturk Experiment we run to motivate the importance of attribute level ratings. First, to establish that enhanced ratings are useful for customers to make better decisions, we conducted a 2×2 between subjects study on MTurk with 165 participants. Both the treatment and control groups are shown 4 restaurant reviews and asked to chose a restaurant. Every restaurant is extremely good at one of the attributes— food, service, price or ambiance and average on other attributes. The only additional information given to the treatment group is enhanced attribute level ratings. See Figure A1 for details of the study design. We compare the treatment and control groups on two parameters— match and attention. We consider a match when a person's restaurant choice matches with their separately elicited preference i.e. a person who says she values food chooses the restaurant that has excellent food and so on. We get our measure of attention based on whether the survey respondent correctly answers the attention check question: "How many restaurant choices did you have in the previous question?" asked right after the restaurant choice question.

We show in Table A1 that providing attribute sentiment scores in addition to text significantly improves the ability of customers to choose restaurants consistent with their separately elicited preferences over restaurant attributes. There is also a significant positive impact on attention. The fact that the treatment group is more attentive and makes choices more consistent with preferences shows that attribute level ratings reduce the cognitive burden of consumers and helps them in decision making.

[Insert Figure A1 and Table A1 here]

LDA and s-LDA for exploratory topic analysis

We use document level LDA and s-LDA as an exploratory tool to identify which topics are discussed in reviews. Figure A2a and A2b present the topics identified from the document level LDA

Electronic copy available at: https://ssrn.com/abstract=3395012

and s-LDA.

[Insert Figures A2a and A2b]

A Hybrid convolutional-LSTM Deep Learning Architecture

In this section, we include a more detailed discussion of the two most important layers of the hybrid CNN-LSTM: the convolutional layer and the long short term memory layer.

Convolution Layer. The first feature generating layer in our architecture that follows the embedding layer is the convolution layer. Convolution refers to a cross-correlation operation that captures the interactions between a variable sized input and a fixed size weight matrix called filter (Goodfellow et al. 2016). A convolutional layer is a collection of several filters where each filter is a weight matrix that extracts a particular feature of the data. In the context of text classification, a filter could be extracting features like bi-grams that stand for negation *e.g. not good* or unigrams that stand for a particular attribute e.g. chicken. The two key ideas in a convolutional neural network are weight-sharing and sparse connections. Weight-sharing means using the same filter to interact with different parts of the data and sparse connection refers to the fact that there are fewer links between the neurons in adjacent layers. These two features reduce the parameter space of the model to a great extent thereby lowering the training time and number of training examples needed. Thus, CNN-based models take relatively little time to train compared to fully-connected networks or sequential networks. Training a CNN involves fixing the weight matrix of the shared filters by repeatedly updating the weights with the objective of minimizing a loss function that captures how far the predicted classification of the model is from the true class of training data.

An embedded sentence vector of dimension $n \times d$ enters the convolution layer. Filters of height h (where filter height denotes length of n-gram captured) and width d act on the input vector to generate one feature map each. For illustration purposes, let us consider a filter matrix F of size $h \times d$ that moves across the entire range of the input I of size $n \times d$, convolving with a subset of the input of size $h \times d$ to generate a feature map M of dimension $(n-h+1) \times 1$. A typical convolution operation involves computing a map by element-wise multiplication of a window of word vectors

Electronic copy available at: https://ssrn.com/abstract=3395012

with the filter matrix in the following manner:

$$M(i,1) = \sum_{i=1}^{n-h+1} \sum_{m=1}^{h} \sum_{n=1}^{d} I(i+(m-1),n)F(m,n) \tag{6}$$

When there is a combination of filters of varying heights (say 1,2,3 etc.), we get feature maps of variable sizes ($n, n-1, n-2$ and so on).

Max-pooling and flattening operations are performed to concatenate variable size feature maps into a single feature vector that is passed to the next feature generating layer.

The role of the convolutional layer in this model is to extract phrase-level location invariant features that can aid in attribute and sentiment classification. A feature map emerging from a convolution of word vectors can be visualized as several higher-order representations of the original sentence like n-grams that capture negation like "not good" or "not that great experience" or n-grams that describe an attribute like "waiting staff" or "owner's wife." The number of filters to be used, N_f is fixed during hyper parameter tuning. Feature maps from all filters are passed through a non-linear activation function a_f with a small bias or constant term b to generate an output that would serve as input for the next stages of the model.

$$O_i = a_f(M_i + b) \tag{7}$$

The function f here can be any non-linear transformation that acts on the element-wise multiplication of the filter weights and word vectors plus a small bias term b. We use Rectified Linear Units (RELU) that is more robust in ensuring the network continues to learn for longer time periods compared to other activation functions like the tanh function (Nair and Hinton 2010). This activation function has the following format:

$$RELU(x) = max(0,x) \tag{8}$$

This activation function sets all negative terms in the feature maps to zero while preserving the

Electronic copy available at: https://ssrn.com/abstract=3395012

positive outputs.

[Insert Figure A3a here]

Figures A3a and A3b show the structure of the convolution layer and the convolution operation respectively. Figure A3c shows a sample visualization of a feature map. During the course of training, each filter specializes in identifying a particular class. For instance, this filter has specialized in detecting *good food*.

Long Short Term Memory (LSTM) layer. The concatenated feature maps from the convolution layer are next fed into a Long Short Term Memory (LSTM) layer. LSTM is a special variant of the recurrent neural networks (RNN) that specialize in handling long-range dependencies. RNNs have a sequential structure and hence they can model inter-dependencies between the current input and the previous inputs using a history variable that is passed from one time period to the next. However, in practice, RNNs fail to do text classification tasks better than CNNs due to the "vanishing gradient" problem which causes a network to totally stop learning after some iterations (Nair and Hinton 2010). Vanishing gradients in the earlier layers of a recurrent neural network mainly result from a combination of non-linear activation functions like sigmoid and small weights in the later layers. LSTMs solve this problem by using a special memory unit with a fixed weight self-connection and linear activation function that ensures a constant non-vanishing error flow within the cell. Further, to ensure that irrelevant units do not perturb this cell, they employ a combination of gate structures that constantly make choices about what parts of the history need to be forgotten and what needs to be retained to improve the accuracy of the task at hand (Hochreiter and Schmidhuber 1997). This architecture has shown remarkable success in several natural language processing tasks like machine translation and speech to text transcription.

[Insert Figures A3a, A3b and A3c here]

[Insert Figure A4 here]

Figure A4 is a comparison of RNN and LSTM architectures. In an RNN, the output at a particular time t is fed back into the same network in a feedback loop. In this way, a new input

Electronic copy available at: https://ssrn.com/abstract=3395012

x_t interacts with the old history variable h_{t-1} to create the new output o_t and the a new history variable h_t. This is like in a relay race where each cell of the network passes on information of its past state to the next cell (but each cell is identical, and therefore it is equivalent to passing on the information to itself). The Long Short Term Memory (LSTM) cell differs from the RNN cell on two important aspects—the existence of a cell state C_t (the long term memory) and a combination of gates that regulate the flow of information into the cell state. The cell state is like a conveyor belt that stores the information that the network decides to take forward at any point in time t. Gates are sigmoidal units whose value is multiplied with the values of the other nodes. If the gate has a value of zero, it can completely block the information coming from another node whereas if the gate has a value $\in (0,1)$, it can selectively allow some portion of the information to pass. Thus, gates are like "regulators" of what information flows into and remains active within the system. The LSTM has three gates — a forget gate G_F, an update gate G_U and an output gate G_O.

Suppose x_t represents the input to the LSTM at a particular time t and h_{t-1} denotes the hidden state (or history) that is stored from a previous time period. At the first stage, the forget gate decides what part of the previous state needs to be forgotten or removed from the cell state. For instance, in a long sentence, once the LSTM has figured out that the sentence is primarily about the taste of a burger, it might chose to remove useless information regarding weather or day of the week that says nothing about food taste. The transition function for the forget gate can be represented as :

$$f_t = \sigma(W_f \dot{[}h_{t-1}, x_t] + b_f) \tag{9}$$

This equation is a typical neural network equation that involves an element-wise multiplication of a weight function with the hidden state h_{t-1} and current input x_t followed by the addition of a bias term and subsequent non-linearity. The other transition functions of the LSTM include an update function and an output function. The update function decides what part of the current input needs to be updated to the cell state. The output function first determines the output o_t for the current time period and subsequently, the new hidden state h_t that is passed to the next time period

Electronic copy available at: https://ssrn.com/abstract=3395012

by selectively combining the current output and cell state contents that seem most relevant.

$$
\begin{cases}
i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) & (10) \\
\tilde{C}_t = \tanh(W_c[h_{t-1}, x_t] + b_c) & (11) \\
C_t = (f_t C_{t-1} + i_t \tilde{C}_t) & (12) \\
o_t = \sigma(W_o[h_t - 1, x_t] + b_o) & (13) \\
h_t = o_t \tanh(C_t) & (14)
\end{cases}
$$

All the weight matrices W_f, W_i, W_c and W_o are shared across different time steps. Thus, training an LSTM basically involves training these shared weight matrices by optimizing over a loss function.

Additional Performance Metrics

In this section, we describe some additional performance metrics that were excluded in the main text for brevity. These include some objective metrics like precision and recall (derived from the confusion matrices) and some practical considerations like model building effort, scalability and interpretability. See Tables A4b and A4c to see how different models perform on metrics like precision, recall and F1 score.

[Insert Table A4b and A4c here]

Model Building Time Lexicon models take approximately 175-180 hours of construction time. Most of the time is spent on human-tagging of the 8575 attribute and sentiment words into specific classes using Amazon's Mechanical Turk. Similarly, the creation of training and test data sets for the supervised learning algorithms takes approximately 100 hours.[21] However, once created, we could use the same dataset to train and test a variety of machine learning and deep learning classifiers (e.g., SVM, Random Forest, Naive Bayes, CNN, LSTM and CNN-LSTM). After generating

[21] A human tagger takes around 1 minute to classify every word and 2-3 minutes to classify full sentences

Electronic copy available at: https://ssrn.com/abstract=3395012

the training data, supervised learning models (including the deep learning models) need time for hyper parameter tuning and model training. Though this is an iterative process, all deep learning models take less than 10 minutes (in a quad core processor) for completing one training cycle and hence model calibration can be completed in 6-7 hours. Thus, model building is time-consuming for all algorithms but is a one-time activity.

Scalability The more time-sensitive metric is scalability i.e. the time required for a trained model to classify new examples. With respect to the scalability metric, the deep learning classifiers clearly outperform the lexicon based classifiers with the machine learning classifiers in between the other two. The main reason is the "look-up" method employed by lexicon based methods. Every word in a sentence needs to be sequentially searched through the entire lexicon to determine its class. Hence, the lexicon methods need several hours to classify our corpus of 27,332 reviews comprising of 999,885 sentences. On the other hand, deep learning models are able to classify our entire review dataset comprising in approximately 18- 20 minutes.

Interpretability refers to how well a machine classifier can explain the reasoning or logic behind its classifications (Doshi-Velez and Kim 2017). In general, text mining methods differ in their strengths and weakness across various dimensions, there is no one method that is superior in all dimensions. Though the CNN-LSTM model outperforms all the other models in accuracy and scalability, however, it falls short in terms of interpretability with respect to lexicon methods.

Electronic copy available at: https://ssrn.com/abstract=3395012

Figures and Tables

Table A1: Match and Attention Comparison: Treatment (Attribute Scores)

	N	Mean (SD)	
		Match	Attention
Treatment	74	0.7 (0.46)	0.94 (0.46)
Control	90	0.38 (0.49)	0.83(0.49)
		$p<0.01$	$p<0.05$

Table A2: Top attribute and sentiment words

Attribute	**Attribute words**	**Positive Sentiment Words**	**Negative Sentiment Words**
Food	Food, chicken, beef, steak, appetizers, cheese, bacon, pork, taste, waffle, dish, shrimp, side, fries, menu, options, vegetarian, meat, gluten, salads, burger, mac, bread, cornbread, ingredients, egg, pancake, portions, brunch, lunch, dinner, breakfast, snack, potatoes, selection, entrée, dessert, maincourse, cake, brownie, ice cream, drink, water, alchol, nonalcoholic, tea, coffee, mocha , vodka, tequila, mocktail, beer, cocktails, cellar, glasses, wine, water	delicious, good, great, fresh, tasty, rich, hot, juicy, perfect, impressed, impressive, overwhelming, crispy, crunchy, warm, authentic, savory, amazing, real, nice, filling ,fantastic, quality, favorite, decent, enormous, special, fluffy, perfection, addicting, hearty, satisfactory, green, outstanding, yummy	not good, not the best, underwhelming, less, light, limited, stale, cold, not fresh, disappointing, awful, salty, off, soggy, unsatisfactory, bland, tasteless, cold, undercooked,watery
Service	Server, waiter, waitress, girl, boy, owner, ladies, manager, staff, bartender, customer service, service, seated, wait time, presentation, hostess, tip, chefs, front desk, reception, greeted, seated, filled, serve, refill, wait time	responsive, quick, friendly, accommodating, helpful, knowledgeable, fast, regular, great, immediately, amazing, kind, polite, great, smile, smiling, attentive, sweet	slow, bored, long, less, irritated, displeased, busy, inattentive, did not ask, rude, cold, long time, queue, long, angry, impolite, careless, dishonest, lied
Price	Price, dollars, money, numbers ($1, $ 5 etc.), credit, debit, cash, payment, discount, deal, offer, pay, total, charge, happy hour, save, spent, worth, bucks, cost, bill, tip, coupon	totally worth, cheap, good deal, bargain, free, worthy, inexpensive	expensive, pricy, pricey, steep, surchage, high, higher, overpriced, loot, too rich, lot, steep, additional charge
Location	location, located, street, address, spot, parking , college, office, airport, neighborhood, area, ny, vegas, california	near, nearby, convenient, walking, short, easy, safe, ample parking, on the way	far, secluded, away, shady, unsafe, dingy, long, travel time, no parking
Ambiance	atmosphere, ambience, ambiance, décor, decore, chair, sofa, tables, place, view, patio, terrace, washroom, restroom, design, furniture, crowd, casino, music, lounge, noise	Impressive, friendly, elegant, beautiful, cool, modern, upscale, outgoing, romantic, mind blowing classy, country ,inviting, big, spectacular, open, lively, very clean, nicely done, calm, positive vibe	busy, crowded, noisy, boring , loud, crunched, old, small, shabby, dirty, stinking, negative, wannabe, not great, shitty, dark, not airy

Electronic copy available at: https://ssrn.com/abstract=3395012

Table A3: Top Similar Words in Word Embeddings: Cosine Similarity Scores

Attribute	**Self Trained**	**Glove Pre Trained**
Food	sushi (0.59), cuisine (0.57), meal (0.56), pizza (0.53), restaurant (0.49), dimsum (0.48), foods (0.48), fare (0.47), burgers (0.47), grub (0.46), salsa (0.44), menu (0.35), fish (0.33)	foods (0.66), eat (0.59), meat (0.56), meal (0.57), vegetables (0.54), nutrition (0.54), foodstuffs (0.53), cooking (0.52), bread (0.51), drinks (0.51), chicken (0.43), seafood (0.48))
Service	service (0.65), waitstaff (0.56), communication (0.55), staff (0.55), hospitality (0.48), consistently (0.48), experience (0.46), attitude (0.44), server (0.34), waiter (0.3)	service (0.79), news (0.48), phone (0.47), mail (0.47), provider (0.47), employee (0.46), customers (0.46), operate (0.45), serve (0.45)
Ambiance	ambience (0.95), atmosphere (0.91), decor (0.85), décor (0.75), vibe (0.75), environment (0.73), decoration (0.65), interior (0.63), aesthetic (0.62), cosy (0.61), setting (0.61)	ambience (0.85), décor (0.57), homey (0.58), convivial (0.53), rustic (0.53), woodsy (0.46), elegant (0.45), clubby (0.45), vibrant (0.44), opulent (0.43), surroundings (0.43), spacious (0.43), cozy (0.43), atmosphere (0.42)
Value	pricing (0.80), prices (0.75), value (0.60), rates (0.55), cost (0.54), markup (0.51), size (0.50), quality (0.50), pricey (0.46), overpriced (0.41), bucks (0.41), expensive (0.4)	prices (0.81), cost (0.61), value (0.57), pricing (0.55), share (0.54), premium (0.54), rates (0.53), inflation (0.52), sales (0.51), buy (0.5), dollar (0.5), low (0.5), rise (0.5)
Location	place (0.70), store (0.62), venue (0.61), starbucks (0.6), theatre (0.59), locale (0.58), intersection (0.58), marketplace (0.58), hotel (0.57), safeway (0.54), parking (0.3)	proximity (0.57), site (0.57), located (0.55), area (0.54), vicinity (0.54), venue (0.52), places (0.52), adjacent (0.5), nearby (0.49), geographical (0.48), situated (0.47), convenient (0.45), remote (0.44), facility (0.43)

Table A4a: Sensitivity to hyper parameter tuning (CNN-LSTM)

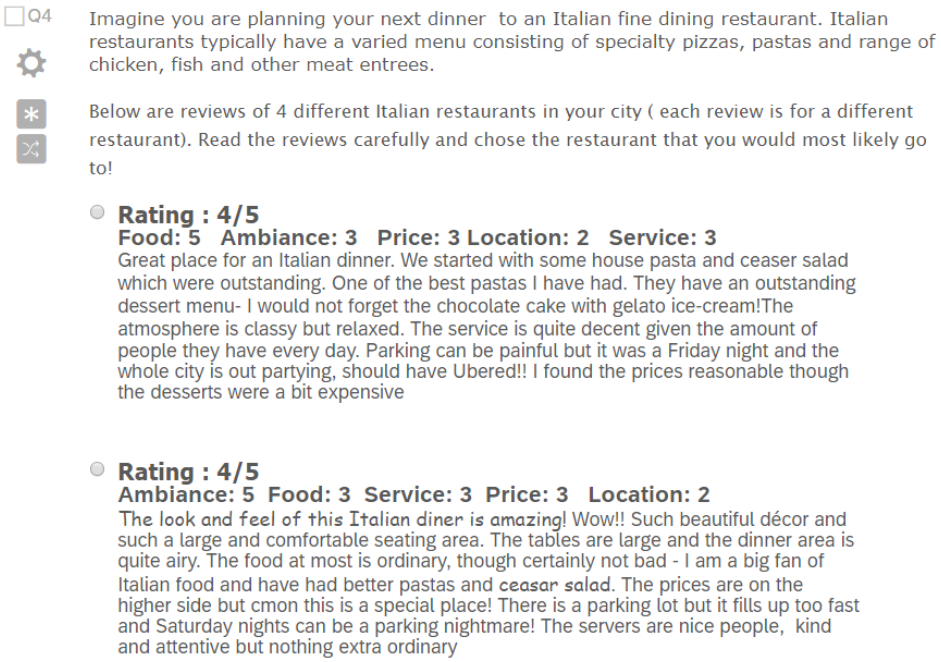
Hyper parameter	Configuration	Attribute Accuracy	Sentiment Accuracy
Embedding dimension	word2vec	58%	40%
	GloVe 100	68%	45%
	GloVe 300	66%	47%
Filter size	unigram	68%	40%
	bigram	67%	42%
	trigram	64%	38%
	[1,2]	66%	41%
	[1,2,3]	66%	42%
	[1,2,3,4]	66%	44%
	[1,2,3,4,5]	64%	47%

Table A4b: Precision and Recall (Sentiment Classification)

	CNN			CNN-LSTM (self-trained)			CNN-LSTM (Glove 300)		
Class	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1
Very Negative	38%	22%	28%	43%	24%	31%	30%	28%	29%
Negative	51%	27%	35%	52%	38%	44%	44%	59%	51%
Neutral	34%	37%	35%	46%	33%	39%	36%	18%	24%
Positive	34%	60%	43%	37%	68%	48%	40%	63%	49%
Very Positive	52%	40%	45%	67%	44%	53%	80%	58%	67%

Electronic copy available at: https://ssrn.com/abstract=3395012

Figure A1: Mturk experiment: Importance of attribute sentiments


Q4

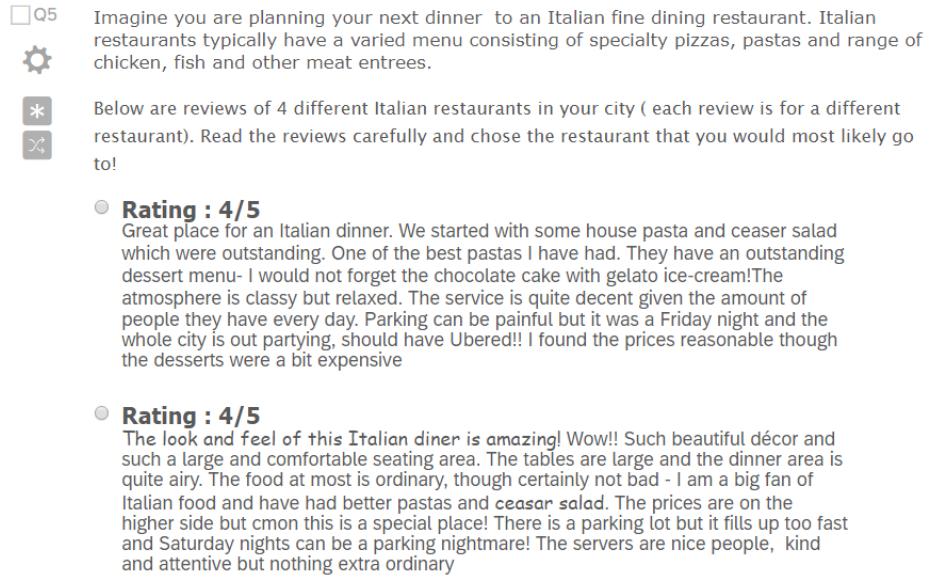
Imagine you are planning your next dinner to an Italian fine dining restaurant. Italian restaurants typically have a varied menu consisting of specialty pizzas, pastas and range of chicken, fish and other meat entrees.

Below are reviews of 4 different Italian restaurants in your city (each review is for a different restaurant). Read the reviews carefully and chose the restaurant that you would most likely go to!

Rating : 4/5
Food: 5 Ambiance: 3 Price: 3 Location: 2 Service: 3
Great place for an Italian dinner. We started with some house pasta and ceaser salad which were outstanding. One of the best pastas I have had. They have an outstanding dessert menu- I would not forget the chocolate cake with gelato ice-cream!The atmosphere is classy but relaxed. The service is quite decent given the amount of people they have every day. Parking can be painful but it was a Friday night and the whole city is out partying, should have Ubered!! I found the prices reasonable though the desserts were a bit expensive

Rating : 4/5
Ambiance: 5 Food: 3 Service: 3 Price: 3 Location: 2
The look and feel of this Italian diner is amazing! Wow!! Such beautiful décor and such a large and comfortable seating area. The tables are large and the dinner area is quite airy. The food at most is ordinary, though certainly not bad - I am a big fan of Italian food and have had better pastas and ceasar salad. The prices are on the higher side but cmon this is a special place! There is a parking lot but it fills up too fast and Saturday nights can be a parking nightmare! The servers are nice people, kind and attentive but nothing extra ordinary

(a) Treatment Group


Q5

Imagine you are planning your next dinner to an Italian fine dining restaurant. Italian restaurants typically have a varied menu consisting of specialty pizzas, pastas and range of chicken, fish and other meat entrees.

Below are reviews of 4 different Italian restaurants in your city (each review is for a different restaurant). Read the reviews carefully and chose the restaurant that you would most likely go to!

Rating : 4/5
Great place for an Italian dinner. We started with some house pasta and ceaser salad which were outstanding. One of the best pastas I have had. They have an outstanding dessert menu- I would not forget the chocolate cake with gelato ice-cream!The atmosphere is classy but relaxed. The service is quite decent given the amount of people they have every day. Parking can be painful but it was a Friday night and the whole city is out partying, should have Ubered!! I found the prices reasonable though the desserts were a bit expensive

Rating : 4/5
The look and feel of this Italian diner is amazing! Wow!! Such beautiful décor and such a large and comfortable seating area. The tables are large and the dinner area is quite airy. The food at most is ordinary, though certainly not bad - I am a big fan of Italian food and have had better pastas and ceasar salad. The prices are on the higher side but cmon this is a special place! There is a parking lot but it fills up too fast and Saturday nights can be a parking nightmare! The servers are nice people, kind and attentive but nothing extra ordinary

(b) Control Group

Electronic copy available at: https://ssrn.com/abstract=3395012

Table A4c: Precision and Recall (Attribute Analysis)

	CNN-LSTM (self trained)			CNN-LSTM (Glove 100)		
Class	Precision	Recall	F1	Precision	Recall	F1
ambiance	60%	58%	59%	55%	77%	64%
food	83%	79%	81%	86%	75%	80%
location	57%	56%	56%	73%	31%	43%
service	80%	60%	69%	71%	76%	73%
price	72%	75%	74%	76%	75%	76%

Figure A2a: LDA topics for Yelp review corpus (with seed words)



Electronic copy available at: https://ssrn.com/abstract=3395012

Figure A2b: SLDA topics for Yelp review corpus

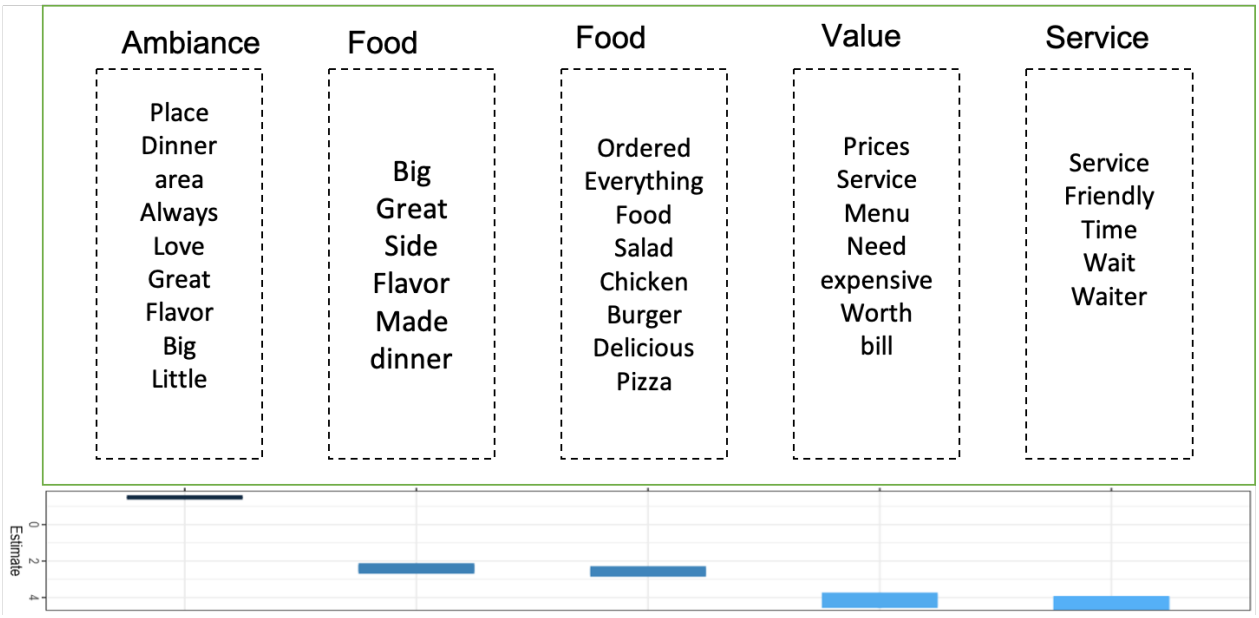
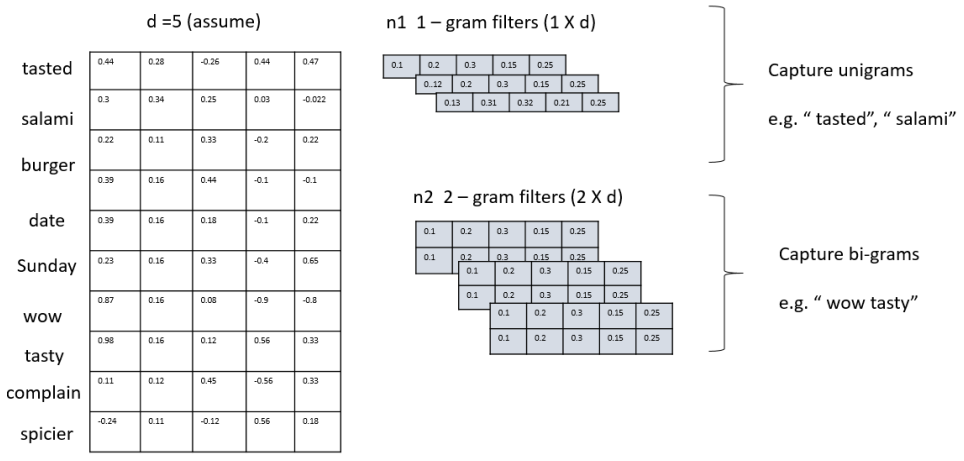


Figure A3a: Convolutional Neural Network: filters and hyper parameters



- The filter sizes (1,2,3), number of filters (n1,n2) and embedding dimension d are important tunable hyper parameters

Electronic copy available at: https://ssrn.com/abstract=3395012

Figure A3b: Convolution Operation

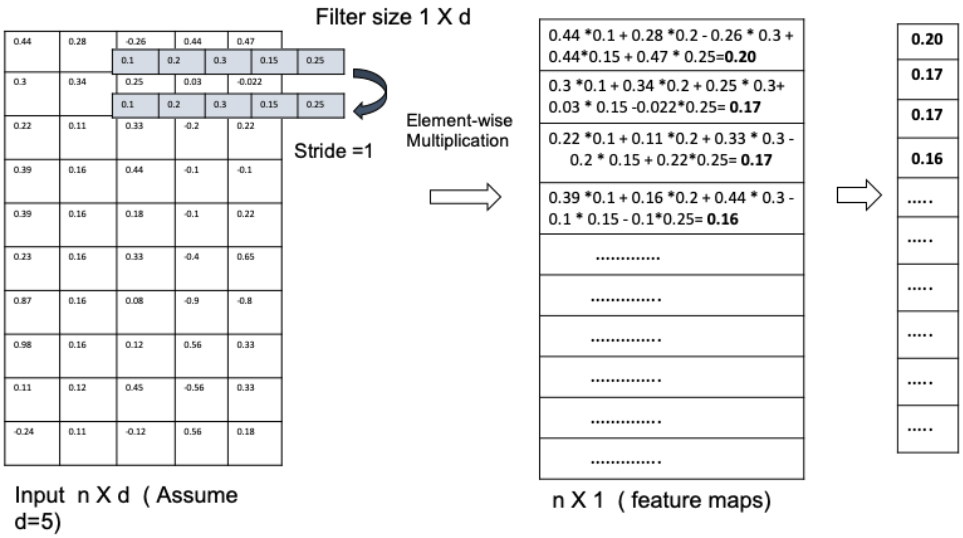
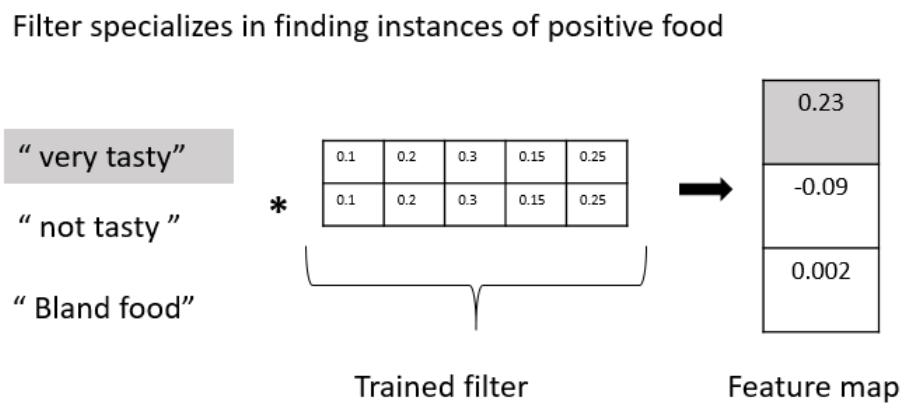
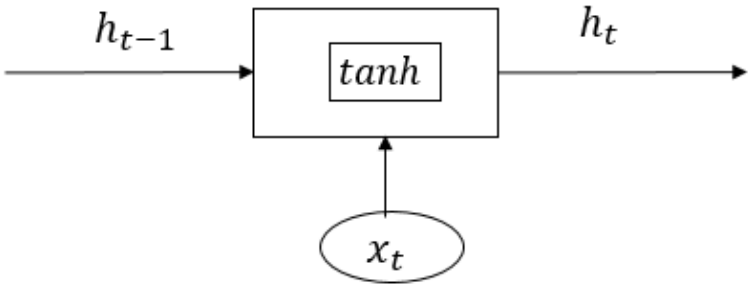


Figure A3c: Visualization of a Feature Map

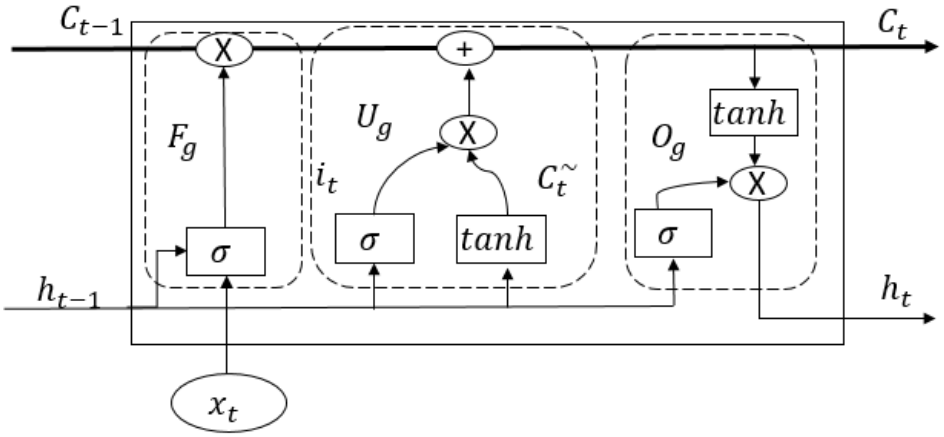


Electronic copy available at: https://ssrn.com/abstract=3395012

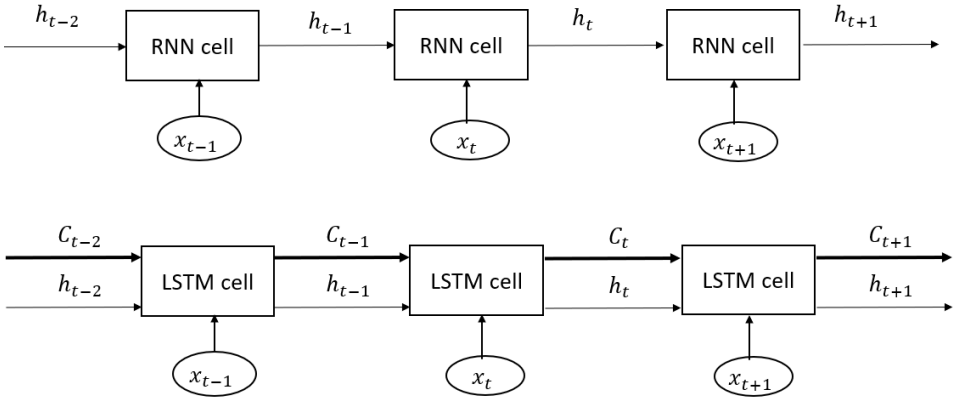
Figure A4: Comparison of RNN and LSTM cells



(a) RNN cell



(b) LSTM cell



(c) Unrolled RNN and LSTM networks

Electronic copy available at: https://ssrn.com/abstract=3395012