**Online Appendix for “An Empirical Comparison of Machine Learning Methods for Text-based Sentiment Analysis of Online Consumer Reviews”**

**Table of Contents**

|  |  |
| --- | --- |
| **Web Appendices** | **Page** |
| A. Model Descriptions | 2 |
| 1 Dictionary | 2 |
| Linguistic Inquiry and Word Count (LIWC) | 2 |
| SentiWordNet | 2 |
| 2 Machine Learning Classification | 3 |
| 2.1 Non-Neural Network-based Machine Learning | 3 |
| K-Nearest Neighbours (KNN) | 3 |
| Naïve Bayes | 3 |
| Support Vector Machine (SVM) | 4 |
| Logistic Regression and Ordered Logistic Regression | 4 |
| Decision Tree and Random Forest | 5 |
| XGBoost | 5 |
| AdaBoost | 5 |
| 2.2 Neural Networks | 6 |
| Feedforward Neural Network | 6 |
| Long Short Term Memory (LSTM) | 6 |
| Bidirectional LSTM (BiLSTM) | 7 |
| Convolutional Neural Network (CNN) for Text | 7 |
| CNN-LSTM | 8 |
| FastText | 8 |
| BERT, RoBERTa, DistilBERT, ALBERT, and XLNet | 8 |
| 3 Topic Discovery | 9 |
| B. Model Explanation (LIME) | 10 |
| C. Additional Tables | 13 |
| Table A1. Marketing Publications with Text Reviews | 13 |
| Table A2. Performance of Text Classification Models in the Computer Science Literature: Yelp and Amazon Fine-Grained Datasets | 18 |
| Additional References | 23 |

**A. Model Descriptions**

**1 Dictionary**

In this section, we discuss two dictionary-based text analysis approaches that are commonly used in marketing research, LIWC (Goes et al., 2014; Ludwig et al., 2013; Sridhar & Srinivasan, 2012; Villarroel Ordenes et al., 2017) and WordNet (Archak et al., 2011; Ghose et al., 2019).

**Linguistic Inquiry and Word Count (LIWC)**

LIWC is commercial software developed by a social psychologist (Tausczik & Pennebaker, 2010). As described on its official website,[[1]](#footnote-1) For a given text, it counts the percentage of pre-defined words from a dictionary which “reflect different emotions, thinking styles, social concerns, and even parts of speech.” The LIWC2015 master dictionary includes “almost 6,400 words, word stems, and selected emoticons.” Related to review sentiment, LIWC contains a dictionary of positive and negative emotions. Positive emotions are all contained in a single category while negative emotions are contained in sub-dictionaries of anxiety, anger, and sadness. LIWC reports for each review, the percentage of positive and negative words, which can then be linked to overall numerical ratings for prediction purposes.

**SentiWordNet**

As described on its official website, [[2]](#footnote-2) WordNet “is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept.” Related to WordNet, SentiWordNet, as described by Baccianella, Esuli, and Sebastiani (2010), is “a resource for supporting opinion mining applications, obtained by tagging all the WordNet 3.0 synsets according to their estimated degrees of positivity, negativity, and neutrality.” Like LIWC, these degrees can be linked to numerical ratings for prediction purposes.

**2 Machine Learning Classification**

As noted in Table 1, we group machine learning methods into two broad categories, classical machine learning methods which do not leverage neural networks, and neural network-based methods.

**2.1 Non-Neural Network-based Machine Learning**

**K-Nearest Neighbors (KNN)**

K-Nearest Neighbors (KNN) is a non-parametric classification method, which is based on a similarity measure such as distance functions (Altman, 1992). In our applications, the input is review text, and the output is class membership (overall numerical rating/sentiment). Given the number of nearest neighbors, , the KNN classifier runs through the entire dataset to compute the distance between the input review and reviews in each training observation. The algorithm assigns the points that are closest to the input review into a set . The classifier then computes for each class, the fraction of points in that have the focal class label and uses it as the estimated conditional probability. Finally, the class with the highest probability is assigned as the prediction for the input review.

**Naïve Bayes**

Naïve Bayes (Lewis, 1998) is a classification method based on Bayes' theorem. It is denoted “naïve” because it assumes independence between the features. Denote the words in a review as and assume independence between , conditional on the category (numerical rating or sentiment) . Then the conditional distribution over the category is , and the category with the highest conditional probability is selected for the review.

**Support Vector Machine (SVM)**

SVM (Cortes & Vapnik, 1995) is a widely used classification method. Given review textand category (numerical rating or sentiment) , SVM finds a hyperplane which separates the two classes with highest margin. The hyperplane equation is defined as , where is a weight vector and is the bias term. If there are two classes (1 and -1), the data are separated according to the inequalities (1) when and (2) when . Input vectors that touch the margin boundaries but not the hyperplane, are called support vectors. The optimization problem is to find and that maximize the margin between the two classes.

**Logistic Regression and Ordered Logistic Regression**

Logistic regression has been widely applied on numerical data. In the context of text reviews, logistic regression can be used to model the probability that the review falls into a certain category (of the numerical rating or sentiment). The binary classification is defined as where is the tokenized text and y is a given numerical rating or sentiment. Each word is represented by numerical feature vectors such as its term frequency–inverse document frequency (TFIDF), which measures the importance of a word to a document in a collection of documents and reduces the effect of less useful tokens that appear frequently in the text reviews. The binary logistic regression predicts whether the review is associated with a certain numerical rating (or sentiment). It treats each rating as independent, without consideration of the sequential order of the ratings. In our context, there is a sequential order within numerical ratings, for instance, a 5-star rating is superior to a 4-star rating, which in turn is superior to a 3-star rating, etc. In this case, the binary logistic regression can be extended to the ordered logistic regression, which considers the ordinal feature.

**Decision Tree and Random Forest**

**Decision Tree**. In contrast to SVM and logistic regression, where input variables enter linearly into the model, the decision tree can allow for complex interactions among variables. The algorithm splits the data via covariates, e.g., words which give the highest information gain and therefore best classify the data. The goal is to find regions (defined by characteristics of review text) that minimize the overall sum of squared errors , where is review 's rating (or sentiment), and is the average rating (or sentiment) of reviews that are in the same region as review .

**Random Forest**. Random forest (RF) is a collection of decision trees. It uses random subsets of the features/words at each split to minimize variance and overfitting for the test set. By averaging many decision trees, this method provides better performance for complicated relationships without hyper-parameter tuning.

**XGBoost**

XGBoost (eXtreme Gradient Boosting) is a decision-tree-based ensemble learning algorithm which is built within a gradient boosting framework.

**AdaBoost**

AdaBoost (Adaptive Boosting) is an ensemble learning algorithm, which uses an iterative approach to increase the efficiency of classifiers. It combines multiple weak classifiers into a strong classifier by learning from the mistakes of weak classifiers through iterations.

**2.2 Neural Networks**

**Feedforward Neural Network**

Feedforward Neural Network is the most basic artificial neural network. It is called “feedforward” because the connections among its nodes do not form any cycle and the data moves in one direction from the input nodes on the left to the output nodes on the right, through the hidden nodes if any. The simplest type of neural network is called a single layer perceptron in which there is no hidden layer. The network is called “shallow” when it has one hidden layer and “deep” when it has at least two hidden layers. The network is represented by each hidden layer which consists of a number of nodes that are represented by subsets of hyper or predefined parameters, including the number of layers, number of nodes in each layer, and learning rate. In our context, the input layer includes tokenized review text, and the output layer is a numerical review rating or sentiment. Each hidden layer, along with the output layer, consists of a set of nodes that are computed by transfer and activation functions (e.g., sigmoid function, ReLU function). The weights, connections among the nodes, and biases are numerically computed using optimization techniques, such as Stochastic Gradient Descent and Adam.

**Long Short Term Memory (LSTM)**

Unlike Feedforward Neural Networks which ignore the ordering of words within the text, Recurrent Neural Networks (RNN) capture the sequential dependencies in the text. It reads one token at a time and feeds the output on each token as an input for the next token. The parameterization is shared over all the positions, and thus the output of the last token recursively depends on the whole sequence. Long Short Term Memory (LSTM), proposed by Hochreiter and Schmidhuber (1997), is a type of RNN that can remember information for longer sequence lengths.

Each LSTM cell has three types of gates that control the flow of information: (i) input gate, (ii) update gate, and (iii) forget gate. These gates enable the LSTM to control the information which passes through the cell. First, the input gate decides what information to add to the cell. Second, the update gate calibrates the level of information to keep from the past. Third, the forget gate decides what information to eliminate from the cell (Olah, 2015).

**Bidirectional LSTM (BiLSTM)**

BiLSTM, proposed by Graves and Schmidhuber (2005), is an advanced version of LSTM in which the network reads the text in both forward and backward directions. This enables BiLSTM to combine information flowing from both directions to typically achieve higher accuracy than LSTM. The architecture of BiLSTM includes two separate hidden states to capture past and future information. The final output is formed by embedding values of the two hidden states in a hyperbolic tangent function.

**Convolutional Neural Network (CNN) for Text**

Convolutional Neural Network (CNN) is a type of Feedforward Neural Net, it consists of convolutional layers which are combined with pooling layers. CNN models are carried out in two steps: “convolution” and “pooling.” Specifically, “‘convolution’ applies a filter over each sliding window of the sentence” to capture important phrases in the text in our context. “‘Pooling’ aggregates the outputs from the filters by creating a location-insensitive summary statistic” (Liu et al., 2019). The model uses local context elements as separate linear prediction tasks on context words corresponding to relative positions to get region representations. In the convolution layer, a small matrix (called kernel or mask) is used to perform data reduction of the input data. After the convolution step, the pooling layer performs a nonlinear transformation, such as maximum pooling or average pooling, on the kernel data.

**CNN-LSTM**

CNN-LSTM (Kiros, Salakhutdinov, & Zemel, 2014) is a neural network that integrates CNN's convolutional and pooling layers into LSTM. Inputs are regions from the text, which are processed by the convolutional layer to extract features. Then, the convoluted text is pooled to reduce data size and the results are used as inputs to the LSTM layer. So, in this type of processing, the model extracts information by analyzing regions of the text rather than words. The LSTM layer sequentially processes the specified regions from texts by minimizing a predefined cost function.

**FastText**

Developed by Facebook, FastText is “is an open-source, free, lightweight library that allows users to learn text representations and text classifiers”.[[3]](#footnote-3) The input is a set of ngram features that are embedded then averaged to form the hidden variable. Next, a sigmoid function is applied to the hidden variables to compute the output, which is the predicted class. One of the key features of FastText is its ability to create word vectors for any kind of word, which allows the model to create vectors for miss-specified words.

**BERT, RoBERTa, DistilBERT, ALBERT, and XLNet**

One of the major problems with the methods so far is that they rely primarily, often solely, on the training data, which is often insufficient to capture the nuances of language.

**BERT** (Devlin et al., 2018), which stands for Bidirectional Encoder Representations from Transformers, is a language representation model that first pretrains deep bidirectional representations from a large unlabeled corpus, thus learning what words mean and how they are used. Then, the pretrained parameters are fine-tuned using the task specific labeled data. Thus, BERT is treated primarily as a black box and the parameters are fine-tuned in an end-to-end manner. Four models related to BERT are RoBERTa (Robustly optimized BERT approach), DistilBERT (a distilled (approximate) version of BERT), ALBERT (A Lite BERT), and XLNet.

**RoBERTa** is a retraining of BERT with improved training methodology. Compared with BERT, RoBERTa includes more subwords than BERT, applies bigger training data, uses dynamic (instead of static) masking pattern, and has different training objectives. Please see Liu et al. (2019a) for details.

**DistilBERT** is a distilled (approximate) version of BERT, which approximates the large neural network of BERT with a smaller and lighter network. Please see Sanh et al. (2019) for details.

**ALBERT** is a lite version of BERT, which reduces memory consumption and increases training speed through parameter-reduction techniques. Please see Lan et al. (2019) for details.

**XLNet** is a large bidirectional transformer, which can achieve better prediction than BERT with improved training methodology. Specifically, it uses a permutation-based training method that predict tokens in random order, which helps the model to learn bidirectional relationship. It uses Transformer XL as the base architecture, which is state-of-the-art autoregressive model.

**3 Topic Discovery**

Latent Dirichlet Allocation (LDA) is a widely applied method in topic discovery. In our context of text reviews, as described by Büschken and Allenby (2016), the LDA assumes there exists a fixed number of latent topics across multiple reviews. Each review has its a mixture of topics, which is represented as a discrete probability distribution over words. Therefore, the presence of a word in a text review is a function of the presence of a latent topic, and the latent topic can be characterized by words with high probabilities to be present in the review. As noted earlier, as an unsupervised learning approach, LDA cannot be directly applied for classification or prediction. However, it affords diagnostic abilities which allow us to analyze the content of reviews with different ratings or sentiment.

**B. Model Explanation (LIME)**

Ribeiro et al. (2016) propose LIME, which has the capability to explain the prediction of any classifier. This is achieved by training an interpretable model locally around the prediction that a researcher wants to explain. LIME is short for Local Interpretable Model-Agnostic Explanations. Local refers to local fidelity that aims to imitate the behavior of the classifier nearby the instance. This procedure is model-agnostic because it works on any classifier and the explanation could ideally be interpretable by the final user.

We only consider words that LIME generates in supporting the predicted rating. This approach is consistent across all ratings, i.e., ratings of 1-5 for the hotel data or 1-3 for the airline data. Figure 1 Panels A and B in the manuscript are explanations of hotel and airline reviews that LIME generates for the rating 5 (and not 5) for hotels, and rating 3 (and not 3) for airlines. Since LIME provides explanations based on a prespecified number of words that the user chooses and we want to automate this value, we vary it based on the following equation:

where is the number of words in instance and is the flooring function that is used to obtain an integer value for the number of words, which will be used in the generated explanation.

We combine explanations of different ratings for aggregating positive and negative sentiments. Specifically, we combine explanations of ratings of 1 and 2 for negative sentiments and explanations of ratings of 4 and 5 for positive sentiments for the hotel dataset, and for the airline dataset we maintained the ratings of 1 (negative), and 3 (positive). We did not consider explanations of the rating of 3 for the hotel dataset or 2 for the airline dataset, because they tend to be neutral sentiments.

Pointwise mutual information (PMI) is a widely used metric in the natural language processing literature to identify importance of explanations (Church & Hanks, 1990). We use PMI with a smoothing method to cope with data sparseness. Our PMI metric is defined as follows:

and

where and refer to the number of instances where the word appears in the positive and negative explanations, and are the number of instances with positive and negative sentiments, and is the total number of instances, i.e., . Figure 1 Panel A shows an example of multiclass LIME explanation of a hotel review in which the output of rating 5 is shown. In this review, “curteous” is the significant word that support the prediction, while “nothing”, “crooked”, “room”, “noisy”, and “dated” are against the prediction. Figure 1 Panel B shows an example of LIME explanation of an airline review in which the output of rating 3 is shown. In this review, “good”, “easy”, “happyflier”, and “boat” support the prediction, while “missing” is against the prediction.

To assess LIME in a way that is generally comparable to the ordered Logit results, we use the test datasets that we did not employ for estimating BERT for hotel and airline datasets. We analyze all text reviews in the test samples across the hotel and airline datasets. We compute the numbers of positive and negative sentiments as follows:

By using the four formulas above, we conclude that the hotel dataset includes 1,389 positive sentiments and 278 negative sentiments, and the airline dataset includes 654.5 positive sentiments and 1,785.5 negative sentiments. We specifically need these values for computing the PMI metrics for positive and negative words.

**C. Additional Tables**

**Table A1. Marketing Publications with Text Reviews**

| **General approach** | **Paper** | **Context** | **Approach** | **DV** | **IVs related to Classification or Scaling of review text** | **Main finding** | **Comparison of approaches?** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Dictionary-based | Kirmani et al., 2017 | Service providers | Manual coding | Perceived usefulness of reviews | Attributes in review (competence, warmth, morality, and other attributes) | Reviews that are perceived more useful mention more competence and morality attributes. | No |
| Dictionary-based | Umashan-kar, Ward, & Dahl, 2017 | Service providers | Manual coding | Probable loyalty in Yelp review | review extent, review failure severity | The positive effect of complaining is higher in strong-tie than in weak-tie industries. | No |
| Dictionary-based | Rubera, 2015 | Automobiles | Manual coding | Sales | Design innovativeness | Design innovativeness decreases initial sales’ status but increases growth rates of sales. There is a negative interaction effect of design innovativeness and technological innovativeness on sales’ initial status, but a positive interaction effect on growth rates of sales. Brand strength and brand advertising expenditures have moderating effects. | No |
| Dictionary-based | Ludwig et al., 2013 | Book | LIWC | Conversion rate | Affective content and linguistic style matches | The level of positive affective content has a strong positive effect on conversion rate. | No |
| Dictionary-based | Ransbotham, Lurie, & Liu, 2019 | Movie | LIWC | Review consumption value | Affective content, concreteness, text characteristics (past, perceptive, personal, informal, cognitive, one-sided, social) | Consumers value mobile reviews less over time. | No |
| Dictionary-based | Goes, Lin, & Au Yeung, 2014 | Epinion.com | LIWC | Objectivity of review, valence of review | None | Users produce more reviews and more objective reviews as they become more popular; but their numeric ratings get more negative and more varied. | No |
| Dictionary-based | Chen, 2017 | Restaurant | LIWC | Sentiment | None | As a reviewer’s number of friends increases, the reviewer provides more negative Yelp reviews. | No |
| Dictionary-based | Van Laer et al., 2019 | “Things to do” on TripAdvisor | LIWC | Positive feedback, attitude, purchase intent | Narrative content, narrative disclosure, review eloquence, extremity, readability | In consumer reviews, narrative content and discourse categories are more persuasive. Such content positively affects consumer response. | No |
| Dictionary-based | Yin, Bond, & Zhang, 2017 | App | RDAL dictionary | Helpful votes | Reading difficulty, emotional valence | At low levels of arousal, increases in expressed arousal increases review helpfulness, but as arousal increases, this relationship gets weaker and eventually becomes negative. | No |
| Dictionary-based | Villarroel Ordenes et al., 2017 | Book, hotel | Part-of-speech tagger; Stanford Sentence and Grammatical Dependency Parser; LIWC; PCNet; manual coding | Positive/negative high/low activated proportions for a reviewer | Explicit sentiment expressions, implicit sentiment expressions, disclosure patterns | For text-based reviews, high activation level and/or boosted sentiment expressions have stronger effects on the review’s overall sentiment strength than low activation level and/or attenuated sentiment expressions. | No |
| Dictionary-based  & Classification | Sridhar & Srinivasan, 2012 | Hotel | Manual coding (product failures), LIWC (valence) | Online product rating | None | Online reviewers are influenced by ratings of other reviewers, and the sign of the effect depends on the reviewer’s idiosyncratic product experience. | No |
| Dictionary-based & Classification | Ghose, Ipeirotis, & Li, 2019 | Hotel | WordNet etc. for feature;  use AMT to assign explicit polarity semantics to words on service features;  classifier for readability and subjectivity | Consumer utility | Average sentence length, number of syllables, spelling errors, subjectivity, topic complexity, sentiment score for helpfulness or staff, food quality, bathroom quality, parking facilities, bedroom quality, check-in/out efficiency | They propose and estimate a structural model to estimate consumers’ preferences and costs on search engines. | No |
| Dictionary-based & Classification | Vermeer et al., 2019 | Facebook and Twitter posts about different brands and industries | Supervised machine learning | Relevance and eWOM | Text | Logistic regression, stochastic gradient descent and passive aggressive classifiers are most accurate. It is efficient to detect relevant eWOM for satisfied consumers and less effective for unsatisfied consumers. For specific industries, domain specificity could produce more accurate results compared to a generic classifier. Classifiers that are trained on Facebook posts produce comparable results on Twitter. | Yes (with dictionary based only) |
| Dictionary-based & Classification | Hartmann et al., 2019 | Major social media platforms | Comparison of lexicon and machine learning approaches | Uncovering human intuition on sentiment and content classification | Text | Random forest and naïve Bayes models outperform the other methods. Lexicon-based approaches perform poorly compared with machine learning approaches. | Yes (five lexicon-based and five machine-learning based) |
| Classification | Ghose, Ipeirotis, & Li, 2012 | Hotel | Naïve Bayes, Manual coding with AMT | Consumer utility | Overall rating, disclosure of reviewer identity, subjectivity, readability | They proposal and estimate a structural model on hotel choice, and develop a new hotel ranking system which is based on the average utility inferred from the model. | No |
| Classification | Packard & Berger, 2017 | Book | Supervised learning | Endorsement style | None | Consumers who were less knowledgeable are more likely to use explicit endorsements. | No |
| Classification | Anderson & Simester, 2014 | Apparel retailer | Automated processing | Deception | Word count, word length, repeated exclamation point | They present evidence that some reviewers write reviews without purchasing the products. These reviews have systematically lower ratings and different text comments, and these reviewers are some of the firm’s best customers. | No |
| Classification | Tirunillai & Tellis, 2012 | Shopping websites such as Amazon | Naïve Bayes, SVM (valence) | Stock performance (return, idiosyncratic risk, trading volume) | Valence of chatter | Volume of chatter leads abnormal returns by a few days. Among the metrics of UGC, volume of chatter has the strongest relationship with abnormal returns and trading volume. There is asymmetric effect of negative metrics of UGC and positive metrics of UGC on abnormal returns. | No |
| Classification | Tirunillai & Tellis, 2017 | Consumer goods | Naïve Bayes, SVM (valence) | Online chatter (popularity, negativity, visibility, virality) | None | Offline TV advertising has a short positive effect on online chatter. This effect is stronger on information-spread dimensions than on content-based dimensions. Importantly, advertising has a small short-term effect in decreasing negativity in online chatter. | No |
| Classification | Wu et al., 2015 | Restaurant | AlchemyAPI | Restaurant choice | Sentiment | The value from Dianping for each user is about 7 CNY, and the value for the reviewed restaurants is about 8.6 CNY per user. The value mostly comes from reviews on restaurant quality, and textual review information is more valuable than numerical ratings. | No |
| Classification | Liu, Lee, & Srinivasan (2019) | About 600 product categories in Home & Garden and Technology departments | Supervised learning | Sales | Price and quality content dimensions | Aesthetics and price content increase sales conversion. | Yes (SVM, Naïve Bayes, LSTM, recursive neural networks, CNN |
| Classification | Zhang & Godes, 2018 | Book | Manual coding + Supervised learning in RTextTools (Review informative or not) | Posted rating (decision quality) | Informativeness of review | Having more ties can lead to better decisions, with sufficient experience. In addition, the strength of the tie can influence the dynamic effects. | No |
| Classification | Lappas, Sabnis, & Valkanas, 2016 | Hotel | Opinion-mining algorithm of Ding et al.  (2008) | Online visibility | Fake review | Even limited injections of fake reviews can impact online visibility. | No |
| Classification | Bai et al., 2020 | Restaurant | SVM, BERT | Yelp rating | Sentiment of perceived food quality, sentiment of perceived service quality | The effect of daily deals on restaurant ratings is mediated by perceived food quality and perceived service quality. | No |
| Topic discovery | Tirunillai & Tellis, 2014 | Shopping websites such as Amazon | LDA | None | None | Methodology focused paper.  They propose a unified framework to extract latent dimensions in marketing applications from user-generated content. | No |
| Topic discovery | Lee & Bradlow, 2011 | Digital cameras | LDA | None | None | Methodology focused paper.  They propose an approach to visualize market structure based on online customer reviews. | No |
| Topic discovery | Büschken & Allenby, 2016 | Restaurant | LDA | Rating | Text | Methodology focused paper.  They propose a new LDA model that utilizes the sentence structure in the reviews, and demonstrate that its superiority relative to existing LDA models in inference and prediction of consumer ratings. | Yes (with previous LDA models only) |
| Topic discovery | Puranam et al., 2017 | Restaurant | LDA | Review topics | None | The proportion of consumer online reviews on health topics increased after the Calorie posting regulation in New York City in 2008. | No |
| Topic discovery | Wang & Chaudhry, 2018 | Hotel | LDA | Subsequent online review ratings | Managers’ response to online reviews | Manager responses to negative reviews, if are observable at the time of reviewing, can positively influence subsequent opinion. | No |
| Dictionary-based & Topic discovery | Archak et al., 2011 | Digital cameras and camcorders | Part-of-speech tagger; WordNet; hierarchical agglomerative clustering | Sales rank on Amazon | Product feature | The textual content in product reviews can explain a significant part of the variation in product demand, which is over and above the predictive power of numeric information such as product price, product age, trends, seasonal effects, as well as the review valence and review volume . | Yes (with manual approach only) |
| Topic discovery | Timoshenko & Hauser, 2019 | Oral-care product | CNN |  | Text | Methodology focused paper.  They propose a machine-learning approach that is based on convolutional neural network (CNN) to select content for efficient review, which can facilitate qualitative analysis. | Yes (with manual approach only) |

**Table A2. Performance of Text Classification Models in the Computer Science Literature: Yelp and Amazon Fine-Grained Datasets**

**Panel A. Yelp**

| **Author and Year** | **Model** | **Predictive Ability** | **Comparison with other Models** | **Is Diagnostic Ability Assessed?** |
| --- | --- | --- | --- | --- |
| Sun et al., 2019 | BERT large + ITPT | Error=28.62 | * ULMFiT, Error=29.98 * BERT base, Error=30.06 * BERT base + ITPT, Error=29.42 * BERT large, Error=29.25 | No |
| Xie et al., 2019 | BERT large (supervised) | Error=29.32 | * Pre-BERT SOTA, Error=29.98 | No |
| Sun et al., 2019 | BERT base + ITPT | Error=29.42 | Comparison is above in (Sun et al., 2019) (BERT large + ITPT) | No |
| Johnson & Zhang, 2017 | DPCNN + tv unsupervised embedding | Error=30.58 | * ShallowCNN + tv unsupervised embedding, Error=32.39 * [CSBL16]’s Character-level CNN, Error=35.28 * FastText bigrams, Error=36.1 * [ZZL15]’s char-level CNN, Error=37.95 * [ZZL15]’s word-level CNN + w2v unsupervised embedding, Error=39.58 * [ZZL15]’s linear model, Error=40.14 | No |
| Xie et al., 2019 | BERT finetune with UDA (semi-supervised) | Error=32.08 | * Random initialized transformer, Error=50.80 * Random initialized transformer with UDA, Error=41.35 * BERT base, Error=41.00 * BERT base with UDA, Error=33.80 * BERT large, Error=38.90 * BERT large with UDA, Error=33.54 * BERT finetune, Error=32.39 | No |
| Johnson & Zhang, 2017 | CNN | Error=32.39 | Comparison is above in Johnson & Zhang, 2017 (DPCNN + tv unsupervised embedding) | No |
| Chen, Ling, & Zhu, 2018 | BiLSTM generalized pooling | Accuracy=66.55 | * BiLSTM max pooling (Lin et al., 2017), Accuracy=61.99 * CNN max pooling (Lin et al., 2017), Accuracy=62.05 * BiLSTM self-attention (Lin et al., 2017), Accuracy=64.21 * BiLSTM max pooling, Accuracy=65.00 * BiLSTM mean pooling, Accuracy=65.30 * BiLSTM last pooling, Accuracy=64.95 | No |
| Joulin et al., 2016 | FastText, h=10, bigram | Accuracy=63.9 | * BoW (Zhang, Zhao, & LeCun, 2015). Accuracy=58.0 * Ngrams (Zhang et al., 2015), Accuracy=56.3 * Ngrams TFIDF (Zhang et al., 2015), Accuracy=54.8 * Char-CNN (Zhang & LeCun, 2015), Accuracy=62.0 * Char-CRNN (Xiao & Cho, 2016), Accuracy=61.8 * VDCNN (Conneau et al., 2016), Accuracy=64.7 * FastText, h=10, Accuracy=60.4 | No |
| Zhang et al., 2015 | Small full CNN with lookup table loading | Error=37.95 | * BoW, Error=42.01 * BoW TFIDF, Error=40.14 * Ngrams, Error=43.74 * Ngrams TFIDF, Error=45.20 * Bag-of-means, Error=47.46 * LSTM, Error=41.83 * Large word2vec CNN, Error=40.16 * Small word2vec CNN, Error=42.13 * Large word2vec CNN with thesaurus, Error=39.58 * Small word2vec CNN with thesaurus, Error=41.09 * Large lookup table CNN, Error=40.52 * Small lookup table CNN, Error=41.41 * Large lookup table CNN with thesaurus, Error=40.52 * Small lookup table CNN with thesaurus, Error=41.17 * Large full CNN, Error=38.40 * Small full CNN, Error=38.82 * Large full CNN with thesaurus, Error=38.04 * Large CNN, Error=39.62 * Small CNN, Error=40.84 * Large CNN with thesaurus, Error=39.30 * Small CNN with thesaurus, Error=40.16 | No |
| Duque et al., 2019 | SVDCNN (29-layers) | Accuracy=64.26 | * SVDCNN (9-layers), Accuracy=61.97 * SVDCNN (17-layers), Accuracy=63.00 * SVDCNN (29-layers), Accuracy=63.20 * VDCNN (9-layers), Accuracy=63.27 * VDCNN (17-layers), Accuracy=63.93 * Char-CNN, Accuracy=62.05 | No |

**Panel B. Amazon**

| **Author and Year** | **Model** | **Predictive Ability** | **Comparison with other Models** | **Is Diagnostic Ability Assessed?** |
| --- | --- | --- | --- | --- |
| Xie et al., 2019 | BERT large (supervised) | Error=34.17 | * Pre-BERT SOTA, Error=34.81 | No |
| Johnson & Zhang, 2017 | DPCNN + tv unsupervised embedding | Error=34.81 | * ShallowCNN + tv unsupervised embedding, Error=36.24 * Hierarchical attention network, Error=36.4 * [CSBL16]’s Character-level CNN, Error=37.00 * FastText bigrams, Error=39.8 * [ZZL15]’s char-level CNN, Error=40.43 * [ZZL15]’s word-level CNN + w2v unsupervised embedding, Error=42.39 * [ZZL15]’s linear model, Error=44.74 | No |
| Xie et al., 2019 | BERT large finetune UDA (semi-supervised) | Error=37.12 | * Random initialized transformer, Error=55.70 * Random initialized transformer with UDA, Error=44.19 * BERT base, Error=44.09 * BERT base with UDA, Error=38.40 * BERT large, Error=42.30 * BERT large with UDA, Error=37.80 * BERT finetune, Error=37.32 | No |
| Joulin et al., 2016 | FastText, h=10, bigram | Accuracy=60.2 | * BoW (Zhang et al., 2015), Accuracy=54.6 * Ngrams (Zhang et al., 2015), Accuracy=54.3 * Ngrams TFIDF (Zhang et al., 2015), Accuracy=52.4 * Char-CNN (Zhang & LeCun, 2015), Accuracy=59.5 * Char-CRNN (Xiao & Cho, 2016), Accuracy=59.2 * VDCNN (Conneau et al., 2016), Accuracy=63.0 * FastText, h=10, Accuracy=55.8 | No |
| Zhang et al., 2015 | Small CNN with thesaurus | Error=40.43 | * BoW, Error=45.36 * BoW TFIDF, Error=44.74 * Ngrams, Error=45.73 * Ngrams TFIDF, Error=47.56 * Bag-of-means, Error=55.87 * LSTM, Error=40.57 * Large word2vec CNN, Error=44.40 * Small word2vec CNN, Error=42.59 * Large word2vec CNN with thesaurus, Error=43.75 * Small word2vec CNN with thesaurus, Error=42.50 * Large lookup table CNN, Error=45.95 * Small lookup table CNN, Error=43.66 * Large lookup table CNN with thesaurus, Error=42.39 * Small lookup table CNN with thesaurus, Error=43.19 * Large full CNN, Error=40.89 * Small full CNN, Error=40.88 * Large full CNN with thesaurus, Error=40.54 * Small full CNN with thesaurus, Error=40.53 * Large CNN, Error=41.31 * Small CNN, Error=40.53 * Large CNN with thesaurus, Error=40.45 | No |

**Additional References**

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, *46*(3), 175–185.

Baccianella, S., Esuli, A., & Sebastiani, F. (2010). Sentiwordnet 3.0: an enhanced lexical resource for sentiment analysis and opinion mining. *Lrec*, *10*, 2200–2204.

Chen, Q., Ling, Z.-H., & Zhu, X. (2018). Enhancing sentence embedding with generalized pooling. *ArXiv Preprint ArXiv:1806.09828*.

Conneau, A., Schwenk, H., Barrault, L., & Lecun, Y. (2016). Very deep convolutional networks for text classification. *ArXiv Preprint ArXiv:1606.01781*.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, *20*(3), 273–297.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *ArXiv Preprint ArXiv:1810.04805*.

Duque, A. B., Santos, L. L. J., Macêdo, D., & Zanchettin, C. (2019). Squeezed very deep convolutional neural networks for text classification. *International Conference on Artificial Neural Networks*, 193–207.

Graves, A., & Schmidhuber, J. (2005). Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks*, *18*(5–6), 602–610.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780.

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 (Volume 1: Long Papers)*, 562–570.

Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016). Bag of tricks for efficient text classification. *ArXiv Preprint ArXiv:1607.01759*.

Kiros, R., Salakhutdinov, R., & Zemel, R. S. (2014). Unifying visual-semantic embeddings with multimodal neural language models. ArXiv Preprint ArXiv:1411.2539.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., & Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. arXiv preprint arXiv:1909.11942.

Lappas, T., Sabnis, G., & Valkanas, G. (2016). The impact of fake reviews on online visibility: A vulnerability assessment of the hotel industry. *Information Systems Research*, *27*(4), 940–961.

Lewis, D. D. (1998). Naive (Bayes) at forty: The independence assumption in information retrieval. *European Conference on Machine Learning*, 4–15.

Lin, Z., Feng, M., Santos, C. N. dos, Yu, M., Xiang, B., Zhou, B., & Bengio, Y. (2017). A structured self-attentive sentence embedding. *ArXiv Preprint ArXiv:1703.03130*.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019a). Roberta: A robustly optimized bert pretraining approach. *ArXiv Preprint ArXiv:1907.11692*.

Olah, C. (2015). *Colah’s Blog*. https://colah.github.io/posts/2015-08-Understanding-LSTMs/

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108.

Sridhar, S., & Srinivasan, R. (2012). Social influence effects in online product ratings. *Journal of Marketing*, *76*(5), 70–88.

Sun, C., Qiu, X., Xu, Y., & Huang, X. (2019). How to fine-tune BERT for text classification? *China National Conference on Chinese Computational Linguistics*, 194–206.

Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of Language and Social Psychology*, *29*(1), 24–54.

Umashankar, Nita, Morgan K Ward, and Darren W Dahl (2017), “The benefit of becoming friends: Complaining after service failures leads customers with strong ties to increase loyalty,” *Journal of Marketing*, 81 (6), 79–98.

Villarroel Ordenes, F., Ludwig, S., De Ruyter, K., Grewal, D., & Wetzels, M. (2017). Unveiling what is written in the stars: Analyzing explicit, implicit, and discourse patterns of sentiment in social media. *Journal of Consumer Research*, *43*(6), 875–894.

Wu, C., Che, H., Chan, T. Y., & Lu, X. (2015). The economic value of online reviews. *Marketing Science*, *34*(5), 739–754.

Xiao, Y., & Cho, K. (2016). Efficient character-level document classification by combining convolution and recurrent layers. *ArXiv Preprint ArXiv:1602.00367*.

Xie, Q., Dai, Z., Hovy, E., Luong, M.-T., & Le, Q. V. (2019). Unsupervised data augmentation. *ArXiv Preprint ArXiv:1904.12848*.

Yin, D., Bond, S. D., & Zhang, H. (2017). Keep your cool or let it out: Nonlinear effects of expressed arousal on perceptions of consumer reviews. *Journal of Marketing Research*, *54*(3), 447–463.

Zhang, X., & LeCun, Y. (2015). Text understanding from scratch. *ArXiv Preprint ArXiv:1502.01710*.

Zhang, X., Zhao, J., & LeCun, Y. (2015). Character-level convolutional networks for text classification. *Advances in Neural Information Processing Systems*, 649–657.

1. <https://liwc.wpengine.com/how-it-works/> [↑](#footnote-ref-1)
2. <https://wordnet.princeton.edu/> [↑](#footnote-ref-2)
3. <https://fasttext.cc/> [↑](#footnote-ref-3)