Journal of Wine Economics, Volume 17, Number 1, 2022, Pages 27–41
doi:10.1017/jwe.2022.2

Using Neural Network Models for Wine Review Classification

Duwani Katumullage [a], Chenyu Yang [b], Jackson Barth [c] and Jing Cao [d]

Abstract

Wines are usually evaluated by wine experts and enthusiasts who give numeric ratings as well as text reviews. While most wine classification studies have been based on conventional statistical models using numeric variables, there has been very limited work on implementing neural network models using wine reviews. In this paper, we apply neural network models (CNN, BiLSTM, and BERT) to extract useful information from wine reviews and classify wines according to different rating classes. Using a large collection of wine reviews from *Wine Spectator*, the study shows that BERT, a neural network framework recently developed by Google, has the best performance. In the two-class classification (90–100 and 80–89), BERT achieves an accuracy of 89.12%, followed by BiLSTM (88.69%) and CNN (88.02%). In the four-class classification (95–100, 90–94, 85–89, and 80–84), BERT yields an 81.57% accuracy, while the other two produce an 80% accuracy. The neural network models in the paper are independent of domain knowledge and thus can be easily extended to other kinds of text analysis. Expanding the limited work on wine text review classification studies, these models are up-to-date and provide valuable additions to wine data analysis. (JEL Classifications: C45, C88, D83)

Keywords: BERT, BiLSTM, CNN, natural language processing, neural networks, wine reviews.

The authors gratefully acknowledge helpful comments and advice from the editor Karl Storchmann and an anonymous reviewer.

[a] Department of Statistical Science, Southern Methodist University, Dallas, Texas, 75275; e-mail: dkatumullage@smu.edu.
[b] Department of Statistical Science, Southern Methodist University, Dallas, Texas, 75275; e-mail: chenyuy@smu.edu.
[c] Department of Statistical Science, Southern Methodist University, Dallas, Texas, 75275; e-mail: jbarth@smu.edu.
[d] Department of Statistical Science, Southern Methodist University, Dallas, Texas, 75275; e-mail: jcao@smu.edu (corresponding author).

I. Introduction

Wine experts and enthusiasts are privy to an exclusive language to communicate about the sensory characteristics of wines such as aroma, flavor, appearance, and mouthfeel. Wine reviews, tasting notes, and wine ratings are supposed to help general consumers gain more information on different wines and enable them to select wines to their liking more easily. However, wine reviews may include opaque and abstract descriptions that are difficult to comprehend for general consumers. For this reason, wine reviews have been labeled as futile and uninformative regarding the sensory properties of wines (Shesgreen, 2003; Silverstein, 2006; Quandt, 2007; Levinson and Majid, 2014). Nevertheless, reviewers usually follow a set script to describe wines, starting with appearance, then the smell, flavor, and finishing with mouthfeel (Paradis and Eeg-Olofsson, 2013). Croijmans and Majid (2016) found that compared to wine novices, wine experts showed more consistency in describing smell and flavors in wine.

In general, text reviews can be time-consuming to read, especially if there are a large number of similar documents. However, a text review may carry richer information on the subject than a single numeric rating. Thus, extracting useful information from text documents presents a rewarding challenge, which has developed into a research area called text mining. It refers to techniques used to process and analyze text data aiming at finding latent connections and patterns in text documents.

In the context of wine reviews, the goal is to retrieve useful information in a systematic and traceable manner, thus helping customers to make well-informed purchase decisions. In this paper, we will use a large set of online wine review data to investigate whether wine reviews contain useful information on wine quality. Specifically, we will use the wine reviews to compare three commonly used neural network models in text analysis by performing two classification tasks. Our work can provide insight on several key aspects of text analysis applied to wine reviews: (1) whether wine reviews carry useful information, (2) whether neural network models can be used to effectively retrieve the information from wine reviews, and (3) among the commonly used neural network models, which model provides the best performance.

In the literature, some of the common choices among conventional statistical methods to analyze wine quality measured by wine ratings are parametric regression models and non-parametric models. These models treat wine ratings as either a continuous, an ordinal, or a categorical dependent variable. The independent variables typically include objective measurements such as physiochemical data, price, vintage, and data on weather, etc. For example, Thompson and Mutkoski (2011) constructed a multiple regression model to predict wine ratings using raters, vintages, and chateaux as the explanatory variables. Their dataset contained 3,133 ratings that ranged from 50–100, assigned by 3 raters, with 11 vintages dated from 1970–2005, which are categorized into 61 chateaux. The authors reported that the model

could explain 64.6% of the total variation in the wine ratings. Xu and Wang (2017) used a larger dataset (i.e., 150,000 observations) to compare the performance of a number of models in the prediction of wine ratings as a continuous variable based on price, wine variety, and winery locations. The comparison included a ridge linear regression model, a lasso regression model, an elastic net regression model, and a neural network model, among which the ridge regression model has the best performance explaining 46.8% of the total variation in the wine ratings in the testing data.

Lemionet, Liu, and Zhou (2015) treated wine ratings as an ordinal variable. They constructed an ordinal K-nearest-neighbor (KNN) classification model, a weighted linear regression, and a newly proposed additive logistic regression algorithm to predict the wine ratings that ranged from 0 to 10. Their choices of the independent variables were 11 continuous physiochemical measures such as volatile acidity, pH, sulphate, etc. The authors explained that the additive logistic regression algorithm turned the multinomial classification of wine ratings into a series of simultaneous binary classification problems. Among the three models, the additive logistic regression model produced the lowest testing error (44.65%) in predicting the wine ratings. The weighted linear regression model produced a 47.55% testing error, and the ordinal kNN classification model produced a 47.33% error. Huang (2018) used the same dataset analyzed by Lemionet, Liu, and Zhou (2015), treating the wine ratings as a categorical variable where the wines with a rating of 7 or above were put in one category, and the rest of the wines were put in the other category. The random forest model produced the highest accuracy of 88.34%, followed by the boosted decision tree model (87.32%), the lasso logistic regression model (78.12%), and the logistic regression model (76.89%).

The challenge of using text data for analysis is that the information conveyed in the text usually cannot be summarized in a straightforward fashion, and the data has a non-standard format. Text documents vary in length, and they are not directly represented by numerical values. The information is embedded in the context of the text, presenting a daunting task that cannot be addressed by conventional analytical methods. From the 1950s to the 1980s, a research area called natural language processing (NLP) has been developed to learn how the human language can be translated in a way that computers can understand. During this time, NLP systems were designed with handwritten rules (i.e., grammar rules). These early NLP systems were extremely difficult, error-prone, and time-consuming to apply to solve real-world problems (Schank and Abelson, 1977). By the 1990s, with the development in machine learning and computational power, NLP started to focus on feature extraction, which summarizes relevant and meaningful information from a text document and transforms it into a numerical representation. Such extracted features from the text can serve as input factors to be included in further statistical analysis. This paved the way for performing analyses like visualization, summarization, content classification and clustering, and semantic and sentiment analysis on chunks of unstructured text data (Miner et al., 2012). Such applications in the domain of

wine review (Chen et al., 2014; Chen et al., 2018; McCannon, 2020) have been used to efficiently extract features in wine reviews that can be incorporated in further analysis.

With the help of a group of wine experts, Chen et al. (2014) collected keywords (i.e., attributes that explain wine properties, such as "aroma," "full-bodied," "black berry," and "tannins") from the "Top 100 Wines of 2011" in *Wine Spectator* and named this domain knowledge as "Computational Wine Wheel." Chen et al. (2018) applied this domain knowledge to the *Wine Spectator* reviews from 2006–2015, where each review was represented as a vector containing the number of occurrences of each keyword. This method is known as the bag-of-words (BoW) model (Salton and McGill, 1983), which omits information on the order and the structure of the words in text documents. The authors used these frequency-based representations of the reviews as the input for a number of white-box classification algorithms, including the Naïve Bayes classifier and the Support Vector Machine (SVM) classifier, to classify the reviews into two rating classes (90–100 or 80–89 on a 100-point scale). The study reported an overall accuracy of 87.21% for the SVM classifier and 84.71% for the Naïve Bayes classifier. Our study shares a similarity with Chen et al. (2018) in terms of the classification problem being addressed and the data being used. However, there are three distinctions that separate our work from theirs. First, in contrast to their representation of reviews through BoW, we will use the prediction-based representation (for which words with similar meanings will have similar representations) for the reviews. This representation takes the context of the text into account. Second, in contrast to their choice of using statistical methods, we will use neural network models for the classification tasks. Third and foremost, the processing and analysis of text reviews in our study are completely independent of domain knowledge, while Chen et al. (2018) required the construction of the "Computational Wine Wheel."

Note that the use of conventional statistical methods such as linear regression models can be inefficient with text data. Hogenboom, Frasincar, and Kaymak (2010) explained that when analyzing text data, such methods focus on discovering statistical evidence for relations between words (i.e., co-occurrence of words). However, relations based on statistical evidence may not be semantically valid. Neural network models, however, can conduct automatic learning to encode semantics between words using low dimensional dense representations (Bitvai and Cohn, 2015), an advantage that played a big role in the development of neural-network-based machine learning algorithms as text analysis tools. In this paper, we will implement three neural network models for wine classification, which are the convolutional neural network, the bidirectional long short-term memory model, and the bidirectional encoder representations from the transformers model.

Convolutional neural network (CNN) (LeCun et al., 1989) is a class of deep neural networks that was originally developed for image analysis. Convolution can be thought of as a sliding window function (i.e., a filter) that concentrates on just one patch of an image. CNN relies on a large number of these filters to first identify

different low-level patterns and then build them up to identify more complex patterns. One signature property of CNN is location-invariance. For example, the algorithm can correctly identify whether an image contains an object, for example, a grape, regardless of where the grape appears in the image. This property of CNN is inherited when applied to NLP tasks where text can be treated as a one-dimensional subject. The location of words is typically ignored by the algorithm due to its location-invariance property. This may introduce a limitation for CNN in text analysis because word location in a sentence carries useful information in the interpretation of the text. Nevertheless, recent studies on CNN have shown encouraging results in performing NLP tasks. For example, Kim (2014) showed that training a simple one-layer CNN with unsupervised pre-trained word vectors for classifying questions into six types (i.e., a question about a person, a question about a location, a question about numeric information, etc.) yielded an accuracy above 90%. Johnson and Zhang (2015) compared CNN with SVM with the n-gram (a continuous sequence of *n* words from a given piece of text) inputs based on an IMDB dataset. Their study showed that CNN produced an error rate 2% lower than that of SVM. The authors argued that, given the sample size, the CNN model could use the information associated with the n-grams more effectively and efficiently.

The long short-term memory (LSTM) neural network is another commonly used model in text analysis (Hochreiter and Schmidhuber, 1997). Its most prominent feature is that it can learn long-distance dependency between sequences of words. LSTM can be trained to keep important information and omit trivial information as it proceeds through text. Robson and Amdahl-Culleton (2018) compared the use of LSTM with the Naïve Bayes classifier in the classification of wine reviews according to country, province, variety, ratings, and price. The LSTM classifier produced a 27% accuracy in the classification of 21 discrete levels in the rating category, while the Naïve Bayes classifier gave an 18.5% accuracy. The authors claimed that the improvement from LSTM stems from its ability to capture the information on the order of words and the dependency among words in text, which the Naïve Bayes classifier lacks.

In comparison to LSTM, which incorporates the input sequence of words only in the forward direction, bidirectional LSTM (BiLSTM) (Schuster and Paliwal, 1997) can retrieve more information from the input text context as its architecture allows it to look at the input sequence in both forward and backward directions. Aiken and Meister (2018) used GloVe: Global vectors for word representation, which is an unsupervised learning algorithm for obtaining vector representations for words (Pennington, Socher, and Manning, 2014) in wine reviews to training a BiLSTM model to classify wines in 31-class varietals (e.g., Chardonnay, Bordeaux Blend, etc.). The model achieved a 65% classification accuracy. Bigbee et al. (2019) created word-based unigrams for each review and used those as inputs for a BiLSTM model to classify reviews into five classes of ratings (i.e., 80–84, 84–88, 88–92, 92–96, and 96–100). Based on a dataset of about 130,000 reviews, the model achieved a 68.8% accuracy.

BiLSTM is a popular choice for text classification, thanks to its capability to process the text from both forward and backward directions. However, its drawback is that the two-directional process is not performed at the same time. Google has recently introduced a language representation model, bidirectional encoder representations from transformers (BERT) (Devlin et al., 2018), to address this issue. Compared to BiLSTM, the BERT model retrieves contextual information from all positions in the entire sequence of text simultaneously. It can be fine-tuned to get state-of-the-art results for a range of NLP tasks. It is based on transformers, a model introduced by Vaswani et al. (2017), which uses an attention mechanism to create a vector representation of text. The attention mechanism allows the model to focus on relevant information in the text corpus based on the currently processed point. As far as we know, BERT has not been applied to wine classification studies.

In this paper, we use over 140,000 reviews of wines and their corresponding ratings scraped from the renowned wine critique, *Wine Spectator*, pertaining to the period 2005 to 2016. Our goal is to apply three neural network models, CNN, BiLSTM, and BERT, to extract features from wine reviews and to classify those into different rating classes (two-class classification and four-class classification). Then we compare the performance of the three neural network models in terms of their classification accuracies.

In the next section, we will describe the dataset used for the analysis, followed by the methodology section that provides more details of the neural network models. The analysis and results section presents a summary and comparison of the results of the classifiers. The final discussion section provides direction for future research.

II. Data

We have collected over 140,000 reviews from *Wine Spectator* with a rating of 80 or above given by a total of nine reviewers through the period of 2006–2015. *Wine Spectator* has the highest circulation of any wine magazine in the United States and likely the whole world. Each year, its editors choose more than 15,000 wines to review with detailed tasting notes, numeric ratings, and recommendations. Thus, the data source has a major impact on the wine culture. The tastings are conducted under controlled conditions where the reviewers are only aware of the general type of the wine and its vintage. The blinded tasting setup also ensures the wine reviews are mostly impartial and unbiased. Additionally, the reviews are provided by professional and prestigious wine experts, making them consistent and reliable.

Wine Spectator provides wine scores based on a 100-point scale to measure the overall quality of wines. The labels are classified as follow: 95–100 as Classic (i.e., a great wine that is strongly recommended), 90–94 as Outstanding (i.e., a wine of superior character and style), 85–89 as Very Good (i.e., a wine with special qualities), 80–84 as Good (i.e., a solid, well-made wine), 75–79 as Mediocre (i.e., a drinkable

wine that may have minor flaws), and 50–74 as Not Recommended.[1] Because the majority of wines have a rating in the range of 80–100, we only use the data on those wines.

We start with preprocessing the text data, that is, text normalization. We first change all the words to lower case and remove numbers and punctuation that are present in the text. Further text normalizing steps such as stop-word removal, stemming, and lemmatization are then followed. Stop-words are high-frequency words, such as "is," "the," and "are." Lemmatization refers to the task of determining the root of a word. For example, the words *drink, drank, drunk* have the same root *drink*. Stemming refers to a simpler version of lemmatization, which strips suffixes from the end of the word. For example, the words *drinking* and *drinkers* have the same stem, *drink*. In this study, we choose not to use further text normalization steps since the neural network models that we use are complex enough to recognize different word formations and to put less emphasis on those high-frequency words that have trivial contributions to the classification tasks.

For example, one *Wine Spectator* review for a wine that has a rating of 80 reads "Smells great, but it is a bit tough and acidic in the mouth, finishing with ripe cherry and herb flavors. Drink now. 5,000 cases made." After the text normalization, it becomes "smell great bit tough acidic mouth finish ripe cherry herb flavor drink case made."

The most frequent keywords of the preprocessed reviews are included in the word cloud in Figure 1. Notice the reviews mainly outline sensory information of the wines that cannot be readily collected from wine labels.

As we mentioned earlier, the general expectation of a wine review is that it enhances the information provided in the wine rating. Adhering to the same reasoning by Chen et al. (2018), we designed the study to explore if highly rated wines are distinguishable from not-so-highly rated wines based on wine reviews. Therefore, in the two-class classification, we place the wine reviews with a rating of 90 or above in one class (i.e., highly rated wine) and the reviews with a rating between 80 and 89 in the other class (i.e., not-so-highly rated wines). In the four-class classification, we follow the four classes that *Wine Spectator* assigns to their wines with a score between 80 and 100: 95–100 as Classic, 90–94 as Outstanding, 85–89 as Very Good, and 80–84 as Good. Table 1 provides the proportion of wine reviews in each of the rating classes in the dataset. The total number of the reviews for each rating class is quite unbalanced, where the rating group 85–89 is the largest group out of the four classes with a total of 52.15% reviews and the 95–100 class has the lowest number of reviews consisting of only 1.82% of all the reviews. Consequently, the proportion of reviews in the 80–89 group in the two-class case

[1] More information on *Wine Spectator* is available at https://www.winespectator.com/articles/about-our-tastings.

Figure 1

Word Cloud of the Review Corpus

Table 1

Proportion of Reviews in Each Class for the Two-Class and Four-Class Classifications

Rating Categorization	*Class*	*Proportion*
Two-class	80–89	0.6580
	90–100	0.3420
Four-class	80–84	0.1365
	85–89	0.5215
	90–94	0.3238
	95–100	0.0182

is considerably higher than that in the 90–100 group, the former group being almost twice as large as the latter group.

III. Methodology

NLP is the technology used in aiding computers to understand natural language. It includes extracting information from text data and converting it to numerical representations that computers can further process. There are two main approaches to creating the numerical representation: the frequency-based representation and the prediction-based representation. The frequency-based representation generates vectors or matrix representations of text using the frequency of words or the frequency of co-occurring words. Because the word order and lexical information are not considered, it often has sparse representation and fails to capture contextual information in text documents. In contrast, the prediction-based representation learns the text representation while performing a learning task, for example,

classifying a review to one of the rating classes. This dual-purpose operation results in a relatively low dimensional dense vector representation of text. In the prediction-based representation, words with similar semantic meanings are closer to each other in the vector space. For this reason, the numerical representation of text is also called text embedding.

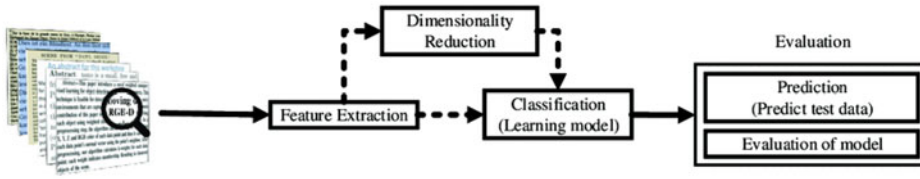
We use three neural network models, CNN, BiLSTM, and BERT, to conduct the two-class and four-class classifications based on the wine reviews. CNN and BiLSTM learn text embeddings while performing the intended classification task. The two classification models consist of an embedding layer along with the neural network layer. The embedding layer takes in the sequence of one-hot encoded word vectors of a review as the input and returns a low dimensional dense vector as the output based on the classification task. A one-hot encoding vector represents each word in the text as a binary vector that has the length of the number of words in the text document, where the value of the position of the current word is 1, and the rest are 0s. The embeddings are the parameters—weights of neurons—of the hidden layer between the embedding layer and the neural network layer. Through proper tuning and training, we can adjust the embeddings to minimize a loss function of choice, for example, the cross-entropy loss. Through such supervised learning, this procedure will produce well-customized embedding vectors trained for the specific classification task. A brief demonstration of this process is demonstrated in Figure 2.

BERT's implementation, demonstrated in Figure 3, is a two-step process consisting of a pre-training step and a fine-tuning step. The pre-training step is conducted simultaneously with two unsupervised tasks. One task is to mask a certain percentage of the input words at random and then predict those masked words. The other task is the next-sentence prediction, where relationships between sentences are learned. As the pre-training corpus, BERT developers used BooksCorpus with 800 million words and English Wikipedia with 2.5 billion words (Step 1 in Figure 3 shows how the BERT model takes in the pre-training corpus). Fortunately, this step has already been accomplished by Google, and the pre-trained model is publicly available. The fine-tuning step is customized for specific downstream tasks such as the current classification task (Devlin et al., 2018). To include this fine-tuning step, we have added a one-hidden-layer classifier on top of the BERT model to perform the classification task and trained it using our dataset. Step 2 in Figure 3 demonstrates how the pre-trained BERT model from Step 1 is fine-tuned for the wine review data based on the hidden classifier layer.

We implement all three classifiers with a stratified 10-fold cross-validation, where the dataset is randomly split into 10 folds. The stratified 10-fold cross-validation ensures that each fold has the same proportion of reviews from each rating class. The classification models are trained based on the reviews and their respective rating classes in the 9 training folds, and then they are used to predict the rating class of the reviews in the testing fold. For each model, we repeat the process 10 times for all the folds and take the average of the accuracies calculated in the testing folds.

Figure 2

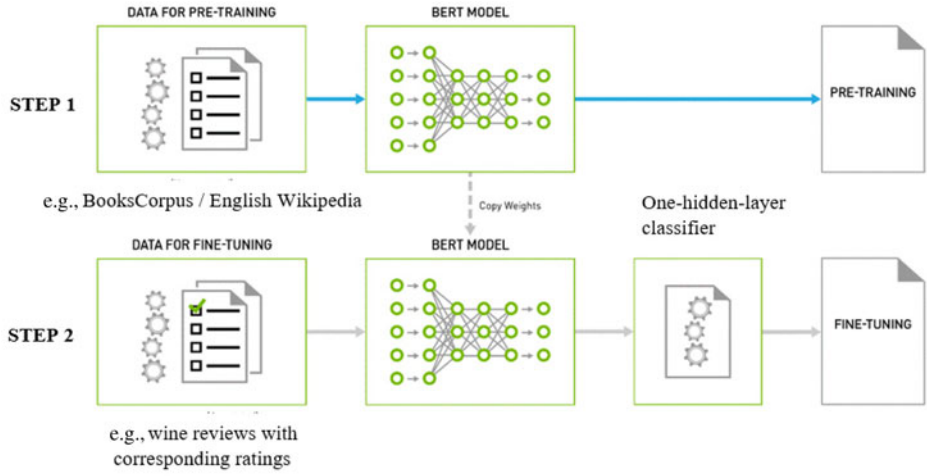
CNN/BiLSTM Pipeline



Source: Kowsari et al. (2019).

Figure 3

BERT Pipeline



Source: Haren (2019).

As shown in Table 1, the number of reviews in each class is quite unbalanced. In such cases, the classification metric, *accuracy*, which is the percentage of all the wines that a classifier has labeled correctly, may not be an appropriate statistic to compare the results. Table 2 shows how *accuracy* is calculated. To make a more explicit example, suppose we have 1,000 wines, where 100 are in Class A and 900 in Class B. A naive classifier blindly classifies all the wines as in Class B, which results in 900 true negatives and 100 false negatives (note that both true positive and false positive are 0), producing an accuracy of 900/1000 = 90%, a result that is very impressive in wine review classification but at the same time useless.

Therefore, in addition to *accuracy*, two additional metrics are used: *precision* and *recall*. *Precision* measures the percentage of the true positives among the identified/

Table 2
Classification Contingency Table

	True Yes	*True No*	
Labeled Yes	True positive (*tp*)	False positive (*fp*)	precision = $\frac{tp}{tp+fp}$
Labeled No	False negative (*fn*)	True negative (*tn*)	
	recall = $\frac{tp}{tp+fn}$		accuracy = $\frac{tp+tn}{tp+fp+tn+fn}$

flagged positives (i.e., true positives + false positives), while recall measures the percentage of the true positives among the observed positives (i.e., true positives + false negatives). *Precision* and *recall* are not affected by the unbalanced classes. In the previous naïve classifier example, despite the 90% accuracy, both *precision* and *recall* are 0 (because there is no true positive). Compared to *accuracy,* which accounts for both true positives and true negatives, *precision* and *recall* only focus on true positives, the subjects of primary interest. The F-1 Score (Lipton, Elkan, and Narayanaswamy, 2014) is a single metric that combines *precision* and *recall*, by taking the harmonic mean of these two. Ideally, we would like a high precision and a high recall for our models. Thus, an F-1 score close to 1 suggests strong model performance.

IV. Analysis and Results

We have implemented the CNN model and the BiLSTM model with Tensorflow on Python. With the 10-fold cross-validation, it takes about four hours to train the CNN model and about two hours to train the BiLSTM model. The Bert model is implemented with PyTorch on Google Colab Notebook, a cloud computing platform. It takes about nine hours to fine-tune the BERT model on the cloud GPU, also with the 10-fold cross-validation.

Table 3 shows the classification results of the CNN, BiLSTM, and BERT models in classifying the reviews into the two-rating and four-rating classes, respectively. For both cases, the BERT model has outperformed the other two models, achieving an 89.12% accuracy in the two-class classification and 81.57% in the four-class classification, respectively. In the two-class classification, BERT shows a 0.43% improvement in accuracy compared to BiLSTM and a 0.67% improvement from CNN. In the four-class classification, BERT has shown a 1.46% improvement from BiLSTM and a 0.76% improvement from CNN. In addition to accuracy, we have also calculated the corresponding F-1 score to compare the performances of the three classifiers. For both of the two classification tasks, the conclusion based on the F-1 score agrees with that based on accuracy, that is, BERT shows the highest F-1 score indicating it has the best performance among the three models.

In the two-class classification, the precision of each model is similar to the corresponding recall, ranging between 86% and 88%. However, in the four-class case, the recall is much lower than the corresponding precision for all the models (i.e., more

Table 3
Accuracy, Precision, Recall, and F-1 Score of the Models

Rating Categorization	*Model*	*Accuracy*	*Precision*	*Recall*	*F-1*
Two-Class	CNN	0.8802	0.8708	0.8626	0.8667
	BiLSTM	0.8869	0.8759	0.8730	0.8744
	BERT	0.8912	0.8797	0.8779	0.8788
Four-Class	CNN	0.7935	0.7767	0.6528	0.7094
	BiLSTM	0.8011	0.7563	0.6675	0.7091
	BERT	0.8157	0.7801	0.7115	0.7442

than 10% lower for CNN, 9% lower for BiLSTM, and 7% for BERT), indicating that, compared to the correct identification among the flagged positives, the models do not perform as well in identifying the correct class out of the observed positives. While this holds for all the three models, it is least serious with the BERT model, which has an F-1 score 3% higher than that of CNN and BiLSTM. In fact, the improvement of BERT over the other two classifiers is more significant in the four-class classification than that in the two-class classification, indicating BERT may have a larger advantage in handling more complex classification tasks.

As we mentioned earlier, Chen et al. (2018) have conducted a two-class classification study using a similar wine review dataset. The highest accuracy achieved in their study is 87.21% with a SVM model. All three neural network classifiers in this study have shown a better performance. Chen et al. have used the BoW representations of wine reviews as the input and applied white-box statistical classifiers (Naïve Bayes and SVM), while we have used the supervised, predictive-based representations of the reviews and employed neural network classifiers. Another distinction between their method and ours is that they have relied on the domain knowledge, that is, the "Computational Wine Wheel" constructed with the help of wine experts to extract keywords and create the BoW representation of the reviews. In contrast, the neural network algorithms applied in this study can automatically extract features from text documents without the need for human intervention, making them more readily adopted for other types of text analysis.

V. Discussion

Wine reviews and wine ratings contain latent sensory information about wines that consumers cannot obtain from objective characteristics measured on wines. Therefore, it is important that wine experts provide consistent reviews and ratings so that consumers can use such sources to learn useful information on different wines. However, the information in wine reviews is embedded in the text, which may be written in opaque and abstract language. The main goal of this study is to explore the performance of neural network algorithms in predicting the rating class of wines based on the latent sensory information contained in wine text

reviews. The successful implementation of such text analysis tools may help general consumers to grasp the essence of wine reviews more efficiently.

We have applied three neural network models to classify a large collection of wine reviews into two rating classes (80–89 and 90–100) and four rating classes (80–84, 85–89, 90–94, 95–100), respectively. The work has provided answers to the debated question of whether wine reviews provide useful information on wine ratings. Wine reviews do carry useful information: the accuracy is quite high on the classification of wine ratings based on wine reviews. Neural network models can be used to effectively retrieve information from wine reviews. In fact, all three neural network models have demonstrated competent performance in the classification tasks. The best performing model is the BERT model in both classification tasks. Its improvement over the other models (CNN and BiLSTM) is consistent, especially in the four-class classification. It is possible that this improvement is due to the fact that BERT is better at capturing contextual information of words in its vector representation than CNN and BiLSTM.

These neural network models could likely yield even better results with additional data or further model tuning. The models are also versatile because they can be optimized with respect to a metric of choice, for example, accuracy, precision, recall, etc. As for future direction, we will investigate how the models can handle consumer reviews, providing an option to study the difference between expert wine reviews and consumer wine reviews. The application of these models to other consumer product reviews with sensory information, such as whiskey, coffee, and cheese, presents a future interest as well.

References

Aiken, T., and Meister, C. (2018). Applying natural language processing to the world of wine. Unpublished manuscript. Available at http://cs230.stanford.edu/projects_spring_2018/reports/8290440.pdf.

Bigbee, J., Chung, S. Y., Im, D. K., Kim, N., and Thirani, J. (2019). Wine rating prediction by reviews. CIS 530 Final Report. Available at http://www.davidim.info/docs/wine_rating.pdf.

Bitvai, Z., and Cohn, T. (2015). Non-linear text regression with a deep convolutional neural network. In C. Zong and M. Strube (eds.), *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Short Papers)*, 180–185. Beijing, China: Association for Computational Linguistics.

Chen, B., Rhodes, C., Crawford, A., and Hambuchen, L. (2014). Wineinformatics: Applying data mining on wine sensory reviews processed by the computational wine wheel. *2014 IEEE International Conference on Data Mining Workshop,* 142–149. doi:10.1109/ICDMW.2014.149.

Chen, B., Velchev, V., Palmer, J., and Atkison, T. (2018). Wineinformatics: A quantitative analysis of wine reviewers. *Fermentation*, 4(4), 82. doi:10.3390/fermentation4040082.

Croijmans, I., and Majid, A. (2016). Not all flavor expertise is equal: The language of wine and coffee experts. *Plos One*, 11(6), e0155845. doi:10.1371/journal.pone.0155845.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805v2 [cs.CL]. Available at https://arxiv.org/pdf/1810.04805.pdf.

Haren, F. V. (2019). Natural language processing (NLP): Meaningful advancements with BERT. Available at https://www.linkedin.com/pulse/natural-language-processing-nlp-meaningful-bert-frederic-van-haren/.

Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780. doi:10.1162/neco.1997.9.8.1735.

Hogenboom, F. P., Frasincar, F., and Kaymak, U. (2010). An overview of approaches to extract information from natural language corpora. In *Proceedings of the 10th Dutch-Belgian Information Retrieval Workshop (DIR 2010)*, 69–70. Nijmegen, Netherlands: Radboud Universiteit Nijmegen.

Huang, J. (2018). Wine quality prediction. Available from http://rstudio-pubs-static.s3.amazonaws.com/438329_edfaab4011ce44a59fb9ae2d216d8dea.html#glm.

Johnson, R., and Zhang, T. (2015). Effective use of word order for text categorization with convolutional neural networks. arXiv preprint arXiv:1412.1058 [cs.CL]. Available at https://arxiv.org/pdf/1412.1058.pdf.

Kim, Y. (2014). Convolutional neural networks for sentence classification. arXiv:1408.5882 [cs.CL]. Available at https://arxiv.org/pdf/1408.5882.pdf.

Kowsari, K., Meimandi, K. J., Heidarysafa, M., Mendu, S., Barnes, L., and Brown, D. (2019). Text classification algorithms: A survey. *Information*, 10(4). doi:10.3390/info10040150.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L. (1989). Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4), 541–55. doi:10.1162/neco.1989.1.4.541.

Lemionet, A., Liu, Y., and Zhou, Z. (2015). Predicting quality of wine based on chemical attributes. Unpublished manuscript. Available from http://cs229.stanford.edu/proj2015/245_report.pdf.

Levinson, S. C., and Majid, A. (2014). Differential ineffability and the senses. *Mind & Language*, 29(4), 407–427. doi:10.1111/mila.12057.

Lipton, Z. C., Elkan, C., and Narayanaswamy, B. (2014). Thresholding classifiers to maximize F1 score. arXiv:1402.1892 [stat.ML]. Available at https://arxiv.org/pdf/1402.1892.pdf.

McCannon, B. C. (2020). Wine descriptions provide information: A text analysis. *Journal of Wine Economics*, 15(1), 71–94.

Miner, G. D., Elder, J., Hill, T., Nisbet, R., Delen, D., and Fast, A. (2012). *Practical Text Mining and Statistical Analysis for Non-Structured Text Data Applications*. Amsterdam: Academic Press.

Paradis, C., and Eeg-Olofsson, M. (2013). Describing sensory experience: The genre of wine reviews. *Metaphor and Symbol*, 28(1), 22–40. doi:10.1080/10926488.2013.742838.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In A. Moschitti, B. Pang, and W. Daelemans (eds.), *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Doha, Qatar: Association for Computational Linguistics. doi:10.3115/v1/D14-1162.

Quandt, R. E. (2007). On wine bullshit: Some new software? *Journal of Wine Economics*, 2(2), 129–135. doi:10.1017/S1931436100000389.

Robson, F., and Amdahl-Culleton, L. (2018). Classy classification: Classifying and generating expert wine review. Unpublished manuscript. Available from https://web.stanford.edu/class/archive/cs/cs224n/cs224n.1184/reports/6909240.pdf.

Salton, G., and McGill, M. J. (1983). *Introduction to Modern Information Retrieval*. New York, NY: McGraw-Hill Book Co.

Schank, R. C., and Abelson, R. P. (1977). *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Structures*. Mahwah, NJ: Lawrence Erlbaum.

Schuster, M., and Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673–2681. doi:10.1109/78.650093.

Shesgreen, S. (2003). Wet dogs and gushing oranges: Winespeak for a new millennium. *Chronicle of Higher Education*, 49(26), B15–B16.

Silverstein, M. (2006). Old wine, new ethnographic lexicography. *Annual Review of Anthropology*, 35(1), 481–496.

Thompson, G. M., and Mutkoski, S. (2011). Reconsidering the 1855 Bordeaux classification of the Medoc and Graves using wine ratings from 1970–2005. *Journal of Wine Economics*, 6(1), 15–36.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. arXiv:1706.03762 [cs.CL]. Available at https://arxiv.org/pdf/1706.03762.pdf.

Xu, K., and Wang, X. (2017). Wine rating prediction. Unpublished manuscript. Available at http://cs229.stanford.edu/proj2017/final-reports/5217737.pdf.