CS 221 Project Proposal: Wine Review Keyword Prediction

Noam Habot, Mackenzie Pearson, and Shalini Ranmuthu

The preliminary goal of our project is to be able to take information about various wines such as the region, price, and variety, and be able to predict keywords about the wine without looking at the wine's description or review. Our secondary goal is to apply the method that we develop to other datasets and predict meaningful keywords for different inputs.

1 Project Scope and Infrastructure

We will train on an online dataset from Kaggle that contains wine reviews scraped from the WineEnthusiast website. The image below shows an example from this dataset that we would use as a training input. There is a field for the detailed review itself, as well as other fields for related information.

country	description	designation	points	price	province	region_1	region_2	variety	winery
US	Mac Watson honors the memory of a wine once made by his mother in this tremendously delicious, balanced and complex botrytised white. Dark gold in color, it layers toasted hazelnut, pear compote and orange peel flavors, reveling in the succulence of its 122 g/L of residual sugar.	Special Selected Late Harvest	96	90	California	Knights Valley	Sonoma	Sauvignon Blanc	Macauley

While the Kaggle dataset has already been cleaned, we will need to do some pre-processing on this dataset before we can actually use it in our model. First off, we will need to extract the most important keywords from all of the wine reviews and build a "dictionary" of important key terms that describe wines. One method to accomplish this is to get the top 200 most frequently used words in the reviews, and then remove an intersection of the highest frequency keywords from a corpus unrelated to wines, such as a book. Let's call the remaining keywords a part of the "dictionary" with its size being m.

Next, we will build a separate model for each of these keywords in our "dictionary." For each of these keyword models, we will train them on our train data set, with the response variable being a 1 if the current keyword is in the review and 0 otherwise. After this step, we will have trained m models, one for each keyword. Then, for prediction purposes, we will run all of the m models on a new wine to see how likely a keyword of a given model will appear in its review. After this, we can aggregate all of the predicted likely words, resulting in a list of keywords that describe the wine.

Our algorithm will be tested on a reserved subset of the Kaggle dataset (not used for training). During testing, the inputs will be in the same format as the training inputs with the wine review field removed, and the outputs will be a list of keywords for each input. We will have 2 evaluation metrics of performance. The first will be what percentage of the "dictionary" keywords present in the review did our model include in the output prediction (in the range [0,1]). The second will be a negative percentage of how many predicted words in the output were not present in the review (in the range [-1,0]). The separate metrics will help us determine if our model is too verbose or not accurate enough, and combining the metrics could also produce an overarching indicator of performance (in the range [-1,1]).

As an extension to this project, we wish to extend this entire algorithm to "plug and play" into another dataset, such as coffee attributes and descriptions or beer attributes and descriptions. These datasets may need additional cleaning in order to get them to the point where we can use the algorithm developed on the wine one to predict keywords for the descriptions of those as well.

2 Baseline and Oracle

We consider a baseline and oracle for both the construction of the key words step in our project and for predicting these key words from our inputted feature variables.
For the construction of the dictionary our baseline will be created by extracting the top 50% key words based on frequency and our oracle will be created by extracting the most frequent key words and removing known none descriptive words such as “the”, “a”, “to”, ... To bridge this gap we plan on utilizing the external corpus (book) strategy described above.
For the construction of our prection model, our baseline will be created by assigning all keywords extracted to all observations and our oracle will be created by assigning keywords given the description of the wine. To bridge this gap we will utilize a validation data test with our train set, tuning parameters as needed.

3 Potential Challenges

Some potential challenges include:

- Remaining as unbiased as possible when picking the dictionary of keywords, while still choosing relevant keywords. Comparing the scores for combinations of words from an n-gram model in the wine reviews vs. an n-gram model in the external corpus (book) could be a good start (i.e. relevant keywords might have a high n-gram model score in the wine review but not the corpus, and irrelevant words might have comparable scores between the two). We could use K nearest neighbors to separate these two categories and prune the preliminary dictionary.
- Predicting whether or not we would expect a word to be in the review. To address this challenge we can construct a model by implementing logistic regression using stochastic gradient descent.

4 Related Work

We found some related works that provide interesting insight into different parts of our project. Sarkar discusses his hybrid technique of keyphrase extraction while Hulth et al. describes his automated keyword extraction using domain knowledge.
In Sarkar's paper[(2)], a hybrid approach to keyphrase extraction is presented from medical documents. Sarkar combines the first approach of assigning weights to candidate keyphrases based on an effective combination of features such as position, term frequency (TF), inverse document frequency (IDF) with the second approach of assigning weights to candidate keyphrases using some knowledge about their similarities to the structure and characteristics of keyphrases available in the stored list of keyphrases. He concludes by showing that the experimental results prove that this hybrid approach performs better than other singular keyphrase extraction approaches.
In Hulth's paper[(1)], they first extract keywords using frequency analysis. Then, they construct a hierarchical domain-specific thesaurus as a second knowledge source. They extract the best keywords by ranking the measures of matching the previously assigned keywords.

5 References

1. Hulth, Anette et al.: ”Automatic Keyword Extraction Using Domain Knowledge”
2. Sarkar, Kamal: ”A Hybrid Approach to Extract Keyphrases from Medical Documents”