Recommender System: Finding The Perfect Wine

David McCleave

October 1, 2022

1 Introduction

The world of wine entices many but confuses most. How does one decide which variety of wine to buy for the flavours they seek, or which wine will pair best with the meal they're preparing for friends and family? In this paper, a recommender system is proposed that will recommend a selection of wine varieties that best match a description given in natural language. The recommender system is a content-based recommendation system that's suggestions are generated by a trained naive Bayes classifier. The recommender system is capable of imitating the role of a Sommelier (a trained wine professional) and was trained on descriptions of various wine varieties in the form of online wine reviews. The classifier achieved an accuracy of 47.92%, however, given that Sommeliers will typically suggest a selection of wines that may match a given description, the accuracy was also calculated using the top three predictions per test description. This resulted in an accuracy of 73.25%, an acceptable result given that the classifier was trained on fairly abstract descriptions based on the human perception of flavour and scent.

This type of recommender system could be personalized for a company, where the reviews of wines or food items they sell can be used to train the classifier and provide recommendations within their catalogue.

1.1 The Dataset

The dataset used to train and evaluate the recommender system was taken from Kaggle, an online dataset hosting website that contains datasets freely available to the public. The dataset, named *winemag-data-130k-v2*, was downloaded in comma separated value (*csv*) format and loaded into a Pandas [2] Dataframe. Headings from the original dataset are given below. The value given in brackets for each heading is the percentage missing values for that column in the dataframe. High values are highlighted in orange. In total there were 129971 entries.

1. **country** (0.05%) - The wine's country of origin.
2. **description** (0.00%) - A description of the wine's taste.
3. **designation** (28.83%) - Type of this brand of wine (for example, "Winter Reserve")
4. **points** (0.00%) - Rating given in the review out of 100.
5. **price** (6.92%) - Price in USD ($).
6. **province** (0.05%) - Home province where the grapes were grown.
7. **region_1** (16.35%) - Region in the province.
8. **region_2** (61.14%) - Additional region field for particular region within a region.
9. **taster_name** (20.19%) - Name of the reviewer.
10. **taster_twitter_handle** (24.02%) - Twitter handle of the reviewer.

11. **title** (0.00%) - Title of the review.
12. **variety** (0.00%) - Variety of the wine (for example, "Chardonnay").
13. **winery** (0.00%) - The winery that the wine is from.

Fields that contained a high percentage of missing values and fields that did not add any useful information for classification and user recommendation were dropped. The fields that were kept are shown below. Entries with no variety and no price were dropped as these fields were used by the classifier and when giving regional suggestions. Dropping these rows resulted in no missing values for these columns. The remaining missing values did not need to be corrected for as they would not impact recommendations nor training and evaluation of the classifier. After this process, 120974 entries remained.

1. **country** (0.05%)
2. **description** (0.00%)
3. **points** (0.00%)
4. **price** (0.00%)
5. **province** (0.05%)
6. **variety** (0.00%)
7. **winery** (0.00%)

Variety Simplification

In total, 697 unique varieties were mentioned in the **variety** column. The vast majority of these varieties had very few reviews and training a classifier when there are 697 possible classes proved to be a tedious time-consuming task. In order to speed up the process of training the model and increase it's accuracy, only the top 38 varieties were kept as class labels, a frequency threshold of 500. Figure 1 shows the frequency of each variety of wine before and after selection. After this, 103075 entries were left, 79.35% of the original dataset.

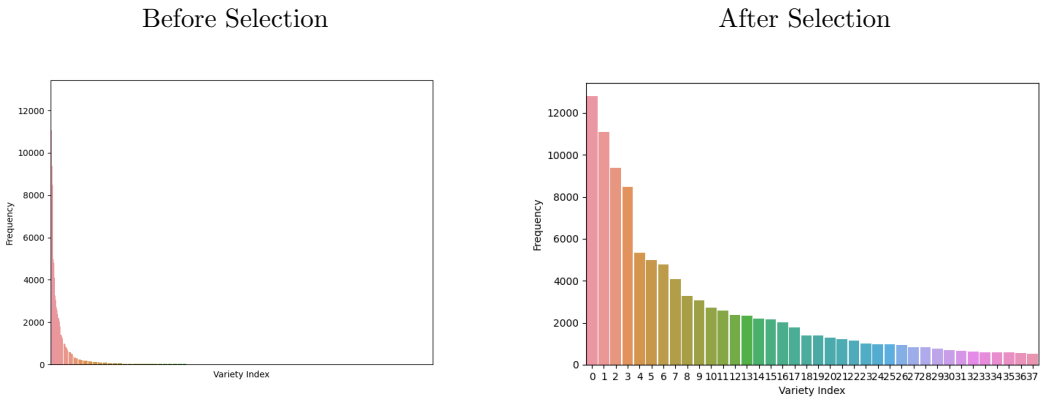

Figure 1: The graphs for frequency versus wine variety before and after selection of the top 38 varieties.

2 Recommender System Process

The sole purpose of this recommender system is to emulate the advice given by a Sommelier. In wine circles, people speak of the flavours and scents of a wine as they would discuss any other hobby, using natural language. The first step in creating this recommender system therefore, was to find a dataset containing natural language descriptions of wine varieties with the associated variety. The dataset taken from Kaggle, *winemag-data-130k-v2.csv*, contains data scraped from the WineMag wine reviewers forum which was scraped over the course of a week during June 2017. A global community of wine reviewers have contributed to this forum resulting in a dataset that contains reviews for wines from all around the world. The reviewers that are allowed to post reviews on this forum are screened by WineMag, meaning only serious wine enthusiasts with some level of credentials have written reviews and the type of language used to describe the wines are consistent with professional standards. This meant the reviews, given in English, in this dataset would be perfect as the chosen feature to train the classification model.

In order to create similarity between items, in this case wine varieties, based on their features (natural language descriptions) a naive Bayes classifier was used. The descriptions went through pre-processing as described in the Implementation section 3 before being used to train the classifier. The naive Bayes classifier works on conditional probability using Bayes theorem given in Equation 1. The naive Bayes classifier assumes all features, in this case lemmatized words in the descriptions of wines, are independent of one another. With reference to Equation 1, B represents the evidence and A represents a class. Thus with respect to classifying wine varieties based on their descriptions, A represents a variety of wine and B represents the lemmatized words in the description for which the probability is being calculated.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{1}$$

where A and B are events.

Through the iterative process of training the naive Bayes classifier, the probability of a class becomes defined by the occurrence of words in the input descriptions. Descriptions with similar words result in similar predictions, hence, the similarity between wine varieties is calculated by the similarity in probabilities for the varieties based on a description. If we refer to probability as the strength of the "belief" the classifier has that a certain wine variety best matches the given description, the recommender system is capable of suggesting the top three wines that it believes the user will enjoy based on the given description.

3 Implementation

All code for the recommender system was implemented in the Python programming language [3]. Pandas dataframes were used to store data read from csv files. The natural language toolkit (NLTK) [1] library was used for some data pre-processing and for classification. Columns were dropped as is described in the Introduction section (section 1) and all descriptions were converted to a list of equal length to the dataframe.

Following this, the descriptions were tokenized. Tokenization utilised the `split` function within Python to split the descriptions into lists of individual words. The lists were cleaned by removing punctuation such as commas and full stops and lowercasing all words before moving on to the lemmatization process.

Lemmatization is the process of grouping together the inflected forms of words so that they can be analysed as a single item. This drastically reduces the number of words the classifier is trained on whilst not losing much meaning in the sentences. An example of a description before and after tokenization and lemmatization is given below.

Before

"This is ripe and fruity, a wine that is smooth while still structured.
Firm tannins are filled out with juicy red berry fruits and freshened with acidity.
It's already drinkable, although it will certainly be better from 2016."

After

'this', 'be', 'ripe', 'and', 'fruity', 'a', 'wine', 'that', 'be', 'smooth',
'while', 'still', 'structure', 'firm', 'tannin', 'be', 'fill', 'out', 'with',
'juicy', 'red', 'berry', 'fruit', 'and', 'freshen', 'with', 'acidity', "it's",
'already', 'drinkable', 'although', 'it', 'will', 'certainly', 'be', 'well', 'from', '2016'

Following tokenization and lemmatization, the data had to be prepared for the classifier model. In order to do this, all words in a description are placed in a dictionary where the words are the keys and the values are all equal to `True`. A tuple is created by combining the dictionary just created with the class label, in this case, the wine variety associated with the description.

To train the model a train, test split of 80 : 20 was used. A train, test split of 60 : 40 and 70 : 30 were also tested, however 80 : 20 resulted in the best balance of training to evaluation accuracy. Since Sommeliers generally will recommend a selection of wines based on the description given by their client, the accuracy measurement for the classification model was calculated using two metrics. The first metric simply calculates the number of times the classifier's highest probability predicted class was equal to the true value. The second metric calculated the number of times the classifier correctly predicted the true value as any of it's top three predictions. The second metric, referred to from this time forward as *Top3 accuracy*, is a more accurate portrayal of the model's predicting capabilities when compared to the typical procedure followed by a human Sommelier.

The same tokenization, lemmatization and preparation by conversion to a dictionary process is applied to any input text given to the model before it can be evaluated. Once the input text has been processed for classification it can be fed into the model which returns a list of all the wine varieties with their associated probabilities. The top three varieties are taken and a highly rated wine for each variety are returned as well, to give an indication of what products the user may want to buy if they wish to experience the flavours and scents they have described.

4 Results

The results section encompasses two subsections. The first subsection, EDA, gives a basic exploratory data analysis on the wine dataframe post the processing described in the Introduction section (section 1) and the Implementation section (section 3). The second subsection, Recommender System, shows results generated by the classifier with some example usage of the recommender system and it's output.

4.1 EDA

Before implementing the recommender system, some exploratory data analysis was conducted on the *WineMag* dataset. Figure 2 shows basic statistical values for the **points** and **price** columns. Interestingly, the results shown in Figure 2 reflect similar results to the values calculated for the *WineMag* dataset before removal of less popular wine varieties. This implies that the most expensive and highest rated wines as well as the cheapest and lowest rated wines are all of popular varieties.

Statistic	**points**	**price** (USD)
Count	103075	103075
Mean	88.49	$36.51
STD	3.08	$42.38
Min	80	$4
25%	86	$17
50%	88	$27
75%	91	$45
Max	100	$3300

Figure 2: A table showing a basic statistical analysis on the continuous columns of the *WineMag* dataset.

The correlation coefficient between **price** and **points** is equal to 0.42. This indicates that there is a low correlation between the price of a wine and how well it is rated in the dataset. This was not expected, as you'll often here people saying that expensive wines always taste better than cheaper wines. The correlation coefficient for **price** and **points** for wines from Stellenbosch was equal to 0.54, indicating a moderate correlation above the normal trend globally. Stellenbosch wines tend to taste better as the price increases.

The most popular varities of wine and their frequencies in the dataset are given in Figure 3.

Variety	Freq.	Variety	Freq.
Pinot Noir	12787	Pinot Gris	1391
Chardonnay	11080	Cabernet Franc	1305
Cabernet Sauvignon	9386	Champagne Blend	1211
Red Blend	8476	Grüner Veltliner	1145
Bordeaux-style Red Blend	5340	Pinot Grigio	1002
Riesling	4972	Portuguese White	986
Sauvignon Blanc	4783	Viognier	985
Syrah	4086	Gewürztraminer	956
Rosé	3262	Gamay	836
Merlot	3062	Shiraz	822
Zinfandel	2708	Petite Sirah	768
Malbec	2593	Bordeaux-style White Blend	695
Sangiovese	2377	Grenache	648
Nebbiolo	2331	Barbera	617
Portuguese Red	2196	Glera	604
White Blend	2172	Sangiovese Grosso	589
Sparkling Blend	2027	Tempranillo Blend	583
Tempranillo	1789	Carmenère	567
Rhône-style Red Blend	1405	Chenin Blanc	533

Figure 3: The most popualar wines and their frequencies in the processed *WineMag* dataset.

4.2 Recommender System

In this subsection, the top 10 most informative features are shown in Figure 4 and the results of some examples are given. The ratio shown for the most informative features represents the number of occurences of the word in reviews for the variety on the left side of the ratio to the same for the variety on the right side. For example, the word grenache appears on average 2845.2 times in reviews for Rhône-style Red Blend for every time it appears in reviews for Chardonnay. In essence, these are the most polarizing words used in the wine reviews.

Feature Name	Variety Names	Ratio
grenache	Rhône-style Red Blend : Chardonnay	2845.2 : 1.0
peach	Glera : Cabernet Sauvignon	2413.8 : 1.0
brunello	Sangiovese : Red Blend	2237.6 : 1.0
viognier	Viognier : Pinot Gris	2234.0 : 1.0
pear	Pinot Grigio : Red Blend	2150.4 : 1.0
gris	Pinot Noir : Chardonnay	1616.1 : 1.0
petite	Petite Sirah : Pinot Grigio	1439.1 : 1.0
chenin	Chenin Blanc : Chardonnay	1373.1 : 1.0
shiraz	Shiraz : Cabernet Franc	1312.6 : 1.0
chardonnay	Chardonnay : Red Blend	1305.4 : 1.0

Figure 4: The top 10 most informative features used to guide the classifier's decision making.

Input Description	Top 3 Varieties	Probability
"strong grassy aroma, farmyard character with a lingering citrus after taste"	Sauvignon Blanc	76.26%
	White Blend	6.47%
	Sparkling Blend	2.97%
"sweet aromatic flowery perfumed wine with high acidity."	Gewürztraminer	53.80%
	Riesling	26.93%
	White Blend	4.07%
"very citrusy, floral with notes of mango and orange on the nose."	Riesling	51.24%
	Chardonnay	12.70%
	Viognier	11.66%
pencil shavings	Syrah	41.45%
	Petite Sirah	11.03%
	Red Blend	7.39%
powerful, overwhelming, strong, unappealing	Petite Sirah	15.19%
	Malbec	12.45%
	Chardonnay	11.99%

Figure 5: Some examples of suggestions given based on the input description.

Some examples of input descriptions and their corresponding top three predicted classes are given in Figure 5. A full example of the additional recommendations returned to a user for an input description is given in Figure 6. In this table, three purchasable wines are presented to the user that would be returned based on the first description given in Figure 5, *"strong grassy aroma, farmyard character with a lingering citrus after taste"*. Interestingly, the descriptions of the wines that are suggested do not contain any of the adjectives in the input description, however, comparable flavours are described. Longer descriptions result in better prediction accuracies. This was expected as the classifier has more features to use to predict which variety of wine is being described. Many varieties share usage of certain words as well, so more words in the input description results in less overlap between varieties.

Due to the nature if this recommender system, all users are treated as first-time users. It would be possible to integrate user profiles, where the suggested varieties of wine would be influenced by previous searches marked as correct by the user. This would begin to shift the behaviour of the recommender system to behave as if it were trained on a heavily skewed class distribution that does not accurately portray the global trends in the wine tasting industry. The additional influence on recommendations given for a specific description would start to reduce the true accuracy of the classifier as the input description would no longer be the only influence. A user whose interests in wine suddenly changes would not get as well informed suggestions if they started trying descriptions very different to the ones they had previously been using. The issue of 'cold start' is less of an issue for this system, as it means the only influence on classifying an input description is the description itself.

A website demoing the functionality described in this paper is accesible at:

https://sommeclassifier.herokuapp.com/

If the webpage does not load, it may be suspended by the hosting company. A screenshot of the website is stored in the same directory as this *pdf*.

Variety	Description	Points	Price	Country	Province	Winery	Designation
Sauvignon Blanc	*"The match for the red wine under the same name, this wine comes from organically grown vines in the village of Champtin. It is rich, full of white currant as well as crisp apple. The tight texture and intense minerality mark it out as a wine to age, while its elegance underlines its fine future. Drink from 2018."*	95	$73.00	France	Loire Valley	Pascal Jolivet	Sauvage
White Blend	*"Made with Malvasia Toscano and Trebbiano grapes (unlike the slightly more complex Occhio di Pernice Vin Santo, which is made from Sangiovese, this is a delightfully rich and concentrated sweet wine that boasts unreal levels of intensity and personality. The wine is smooth and round with lingering notes of brown sugar, roasted almond, melted sugar, dried apricots and figs."*	97	$170.00	Italy	Tuscany	Avignonesi	$375ml$
Sparkling Blend	*"This is an incredibly beautiful and yes, joyful, sparkler, made from equal parts Chardonnay and Pinot Noir. This particular offering was disgorged in July 2016. It positively impresses in minerality, complemented by a more fulsome sense of lemon, apple and hazelnut on the creamy palate, leaving a rich trail of silk on the finish."*	98	$250.00	United States	California	Iron Horse	Joy! Estate Bottled

Figure 6: A wine recommendation for each variety predicted for the first example in Figure 5

References

[1] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

[2] The pandas development team. pandas-dev/pandas: Pandas, February 2020.

[3] Guido Van Rossum and Fred L. Drake. *Python 3 Reference Manual.* CreateSpace, Scotts Valley, CA, 2009.