Stefan Sebastian, 242

Abstract

Contents

1 Dataset **3**

2 Parameters **4**

3 Evaluation metrics **4**

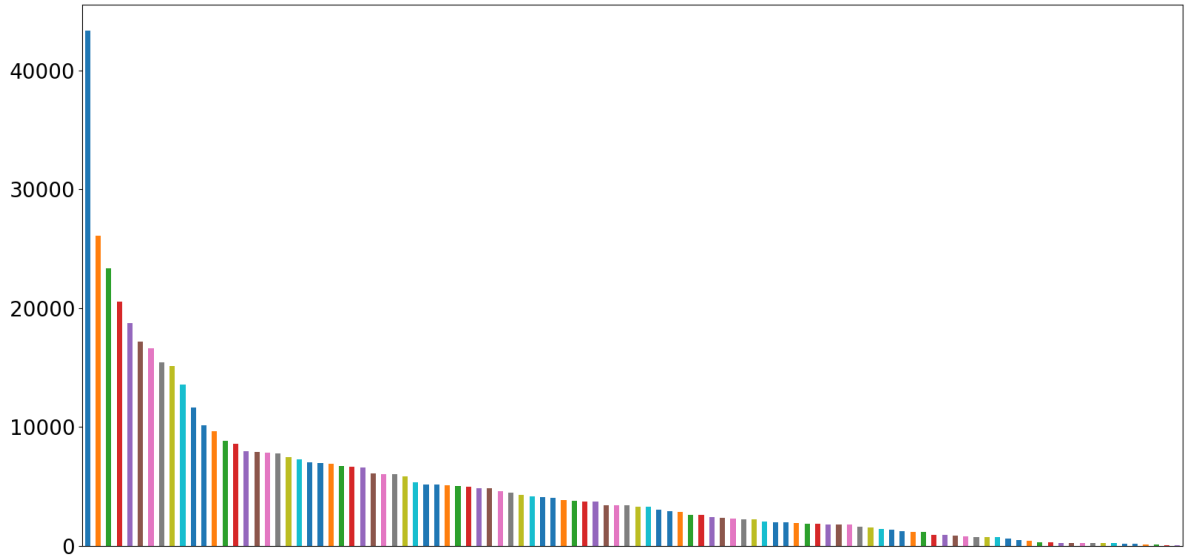

Figure 1: A plot of the number of reviews per beer style

1 Dataset

The dataset used for this experiment is a collection of beer reviews taken from the BeerAdvocate website. The original dataset was made up out of 1.5 million user reviews, from 33387 different users, collected between 1998 and 2011. It is not available at this time but I found a subset of around 500 thousand reviews on data.world[1].

The data is in csv format, each row containing various information like: beer name, beer style, alcohol content, scores for taste, appearance, aroma and a textual review. The only columns considered for this experiment were the beer style and the text review.

On exploratory data analysis, 407 duplicate and 119 missing reviews were found. Also there are 104 distinct beer styles and the amount of data for each of them is quite imbalanced as can be see in figure 1, where each bar represents a beer style with a size proportional to the number of reviews for it.

The first step in preprocessing this dataset was to remove rows containing duplicates and missing values in the columns that are relevant for the experiment: beer style, text review. Next, some of the styles for which there wasn't a lot of data available were dropped, as they would be outliers for clustering. The threshold for this cut was chosen arbitrarily at 7000 reviews.

Upon inspection of the beer styles present in the dataset, a fine granularity was observed. For example, the difference between American Pale Ale and American India Pale Ale is minimal and not very rigorous. This and other similar cases would only confuse the classifier. For this reason, a mapping was made for each style to a broader category, using the taxonomy published by BusinessInsider[2].

The final mapping can be seen in figure 2. Finally, a balanced dataset is built by selecting the minimum value for which we have an equal distribution of reviews per beer style, which is around 9500 per style. The analyzed dataset is made through random sampling of 9500 values for each of the remaining beer styles.

Figure 2: Mapping of beer styles from the original dataset

2 Parameters

The most important parameter used in the experiment is k, the number of clusters. This was chosen to be the number of distinct beer styles in the dataset, 9, as the reviews for beers of the same type should be similar to each other.

The number of features for each data point is equivalent to how many word frequencies are considered. This was determined experimentally by running the basic algorithm with some different feature values. The results, shown in table 2, indicate that until around 1000 there is the biggest growth in performance.

Nr features	Precision	Recall	F1
200	0.420	0.481	0.449
700	0.501	0.554	0.526
1000	0.537	0.532	0.534
1500	0.515	0.528	0.521

Two methods were tested for seed initialization: random points and kd-tree based. The run with kd-trees obtained a 0.47 F1 score while, random initialization got a 0.53 F1 score. This might be caused by finding good starting points through randomness or that the large number of features is not a good fit for the kd-tree data structure.

3 Evaluation metrics

The direction chosen for evaluation is how well the clusters found separate the initial dataset by beer types. In order to do this, each cluster is labeled with the most common style of beer among its elements. This turns our problem intro a multi-label classification one.

The metrics used are the ones described by Beleites at al.[3] for multi-class problems. First of all, a confusion matrix is computed, which is a lxl matrix where l is the number of features. Each element (x, y) in the confusion matrix has a value meaning how many times a data point with label x has been classified as having label y. Precision and recall are then calculated for each label, simulating the binary evaluation.

Precision, also called positive predictive value, is a fraction that represents how many elements classified with a label have been assigned correctly. Recall, also known as sensitivity, is the fraction of instances correctly assigned to a class and all instances from that class. For the binary case, precision and recall can be calculated as in figure 1, where tp, fp, fn stand for true positives, false positives and false negatives. This can be extended to the confusion matrix, for each row, as follows: precision is the fraction of the value on the diagonal and the sum of values on the column, recall is the value on the diagonal divided by the sum of values on the row. The precision and recall of the multi-label classifier can then be computed as the means of the values obtained from the previous step.

$$precision = \frac{tp}{tp + fp} \quad recall = \frac{tp}{tp + fn} \tag{1}$$

F1 score is a combined measure of precision and recall, used to provide a single measurement for the system. It represents the harmonic mean of the two values, as in figure 2, where mp and mr are mean precision and mean recall.

$$F1 = \frac{2 * mp * mr}{mp + mr} \tag{2}$$

References

[1] "Beer advocate reviews." https://data.world/petergensler/beer-advocate-reviews, 2011.

[2] M. Stanger and S. Gould, "Everything you need to know about beer, in one chart." https://www.businessinsider.com/different-types-of-beer-2014-12, 2014.

[3] C. Beleites, R. Salzer, and V. Sergo, "Validation of soft classification models using partial class memberships: An extended concept of sensitivity and co. applied to the grading of astrocytoma tissues.," *Chemometrics and Intelligent Laboratory Systems*, vol. 122, pp. 12–22, 2013.