Cuisine Classification from Ingredients

Boqi Li, Mingyu Wang
CS 229, Stanford University

Introduction

The goal of the project is to categorize a dish's cuisine type (Indian, Italian, Chinese, etc.), by analyzing its ingredient list. The dataset for the project is provided by Yummly. In the dataset, there are 20 types of cuisines. The team need to build a dictionary of ingredients and use several multi-class classification models to predict the type of cuisines given a new dish.

Given the dataset with more than 39000 examples, the total number of ingredients with different names is 6714. There are many ingredients which are basically the same but with slight difference in names. Therefore the ingredients is largely redundant. The team came up with several methods to reduce the number of features and use them in several different classification models to compare their performance.

Since there are 20 classes in the cuisine set. The team used several multiclass classification models including **OVA multiclass Naïve Bayes**, **OVA R2-regulized logistic regression**, **OVA multi-class SVM**, **Multi-class SVM with crammer method**, and **K nearest neighbor**.

Pre-processing and feature selection

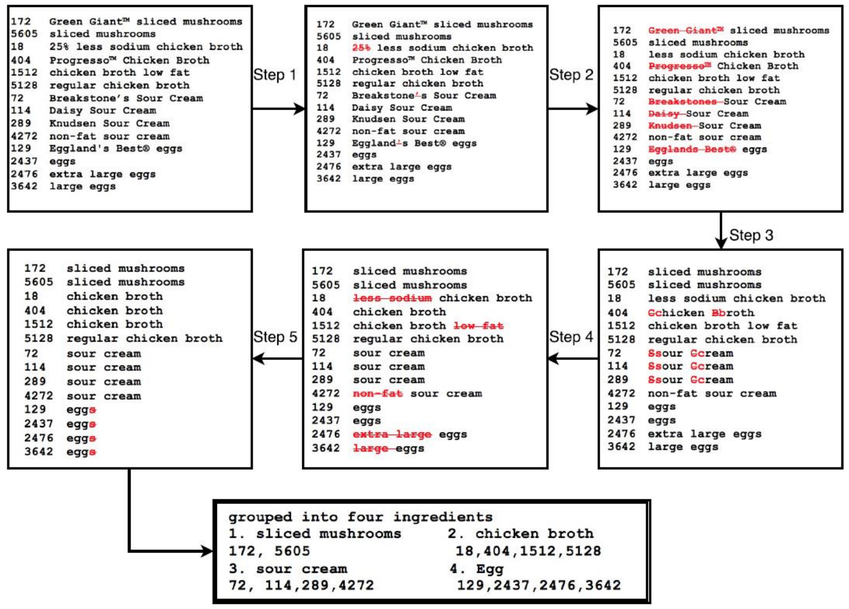

Figure 1. Feature selection Schematic

Five steps in feature selection

- Step 1. remove punctuation, digits, content inside parenthesis
- Step 2. remove brands
- Step 3. convert to lower case and remove stop words
- Step 4. Remove low-content adjectives
- Step 5. Porter Stemmer Algorithm

# of features: 6714 –> 5353
# of cuisines: 20

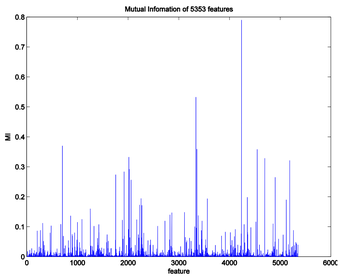

Figure 2. Mutual Information of reduced ingredients

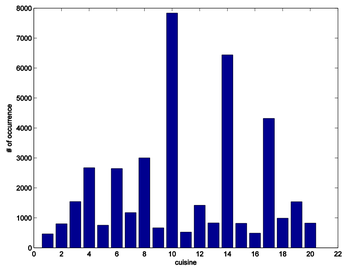

Figure 3. Cuisine distribution

Classification algorithms and performance

- Multi-class text classification problem can be reduced to a series of binary classification problems.
- Error-Correcting Output Coding (ECOC)[1], which trains several numbers of different binary classifiers, and then uses the outputs of these classifiers to predict the label for a new example.
- The code matrix R is the one-verse-all (OVA) code matrix. And it is applied to train a series of Naïve Bayes, SVM and logistic regression classifiers.
- Multiclass SVM introduced by Crammer and Singer[2]
- K-nearest-neighbor algorithm [3]
- 80% of the dataset as training set
- 20% of the dataset as test set
- Key words Combination method only search for same key words to combine.
- Combination & Reduction does both Key words Combination and descriptive words reduction.

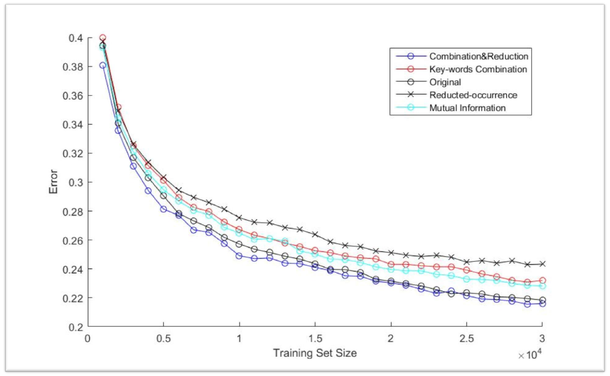

Figure 4. Test Error for Logistic Regression with different feature methods

Table 1. Overall performance for each classification algorithms

	OVA NB	OVA SVM	OVA Logistic	SVM by Crammer	Knn
Original features	0.623	0.751	0.772	0.746	0.661
Reduced-occurrence	0.670	0.742	0.757	0.732	0.683
Mutual Information	0.713	0.757	0.771	0.743	0.697
Key words Combination	0.653	0.750	0.768	0.740	0.690
Combination &Reduction	0.622	0.765	0.784	0.754	0.714

Observations and future work

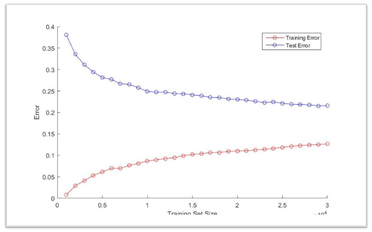

Figure 5. Training error and test error on logistic regression with Combination & Reduction

Chart 1. Error label rate mislabeled into each class

Chart 2. Error label rate of each class mislabeled into other classes

Observations:

- After feature selection, the reduced feature set didn't improve the performance much.
- Chart 1 and Chart 2 show that cuisine types that have small training examples have larger error rates.

Future works:

- Better natural language processing schematic in the context of recipes
- Performance may suffer from the lack of training examples of several cuisines: Brazilian, Jamaican and Russian. Additional training examples in these classes would be beneficial.
- Additional models would be tried like "bags of words", etc.

References & Acknowledgements

The project team gratefully acknowledges Kaggle.com Yummly for making the data for this project publicly available, as well as Andrew Ng and the CS 229 staff for their advice and guidance.

[1] T. Dietterich, G. Bakiri, "Solving Multiclass Learning Problems via Error-Correcting Output Codes " Journal of Artificial Intelligence Research , 1995.
[2] K. Crammer, Y. Singer, "On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines, " Journal of Machine Learning Research, 2001.
[3] D. Trudgian,Z. Yang, "Spam Classification Using Nearest Neighbour Techniques, " Proceedings of the 5th international conference on intelligent data engineering and automated learning, pp.578–585.Revised, 2004.

boqili@stanford.edu mingyuw@stanford.edu