What’s cooking?

- Using recipe ingredients to predict the category of cuisine

IST 565 Data Mining

Mingyu Zhong

Sheida Soleimani

Sara Tugcu

School of Information Studies
SYRACUSE UNIVERSITY

ischool.syr.edu

Introduction

This report will showcase how to go about solving a data mining problem with a categorical output starting from the initial steps of the descriptive analysis, preprocessing the dataset, choosing the models based on the task, conducting the analysis and finally concluding based on the analysis.

The dataset in question is found at Kaggle (Kaggle, 2015a), which is a platform that hosts data science challenges and can be found on the active competition named "What's Cooking?" (Kaggle, 2015b) and it has been made available by the online community site, Yummly (Yummly 2015), which has several features. Amongst other, it allows users to find the perfect recipe based on various search options, such as allergy (e.g. 'Seafood' or 'Gluten'), cook time, tastes (e.g. 'salty' or 'savory') and cuisine (e.g. 'Indian' or 'Chinese'). But with all this data, could we, by asking other questions, use the data to help the users or other target groups? This is what we set out to investigate with this report.

The data set is a transactions data set, which first column contains names of cuisines and the remaining columns consists of ingredients, such as tomatoes, pepper etc., and each row is a dish. A thorough description will be provided in the next section.

The Kaggle competition asks the participant to "predict the category of a dish's cuisine given a list of its ingredients." (Kaggle, 2015b), which is interesting as this seems as the question to ask to make the data mining task a classification problem. But after exploring the data set, yet another hypothesis came to mind, which takes on the approach of association rule mining. To test our hypotheses of whether both approaches will work, this report will seek to answer the two following research question:

- *Can computers distinguish different types of cuisines based on the ingredients? (classification)*
- *What ingredients tend to appear together in recipes? (association rule mining)*

We have chosen this task, as it has a business application and we find it interesting to gain a greater understanding of the underlying technology provided by Yummly, which for example allows the users of the site to find out what can be made based on which ingredients the person has available at home. Furthermore, it allows the user to find what ingredients are needed based on which kind of cuisine he or she would like to cook.

Descriptive Analysis

The data made available is divided into "train" and a "test" data (Kaggle, 2015c), which allows it to be used with several algorithms, for example "classification", which will be explained later in the report. Both datasets are in the open standard format, JSON. The JSON format is accepted by Waikato Environment for Knowledge Analysis (Weka) (Weka, 2015), a software suite which will be the primary software for our data analysis, but as we experienced issues loading the "train" dataset into Weka, we decided to convert (Convert CSV, 2015) it to CSV to gain a better overview, with the hopes of solving the issue(s) resulting in the loading error. This also allows for an easier overview to conduct a descriptive analysis of the "train" dataset. The test data will not be used, as we are not submitting our result to Kaggle, furthermore, we do not have the target class on the test data, so we cannot use it for our classification model.

Even after the conversion, we experienced an error, so we decided to use R, a software for statistical computing and graphics (R Project, 2015), to conduct our descriptive analysis, which is done to get a better understanding of the data set than from just eyeballing it, as a well as understanding which approaches we could use to downsize the data set with the objective to load it in Weka.

The overview led to some questions asked, such as how many of each cuisine are there (below model shows us that the majority of the recipes belongs to italian and mexican cuisine).

The dataset contains 39775 rows and 66 attributes all of the content being nominal. The first attribute in the data set is "ID", which is a unique number for each cuisine, the second attribute is "cuisine" and is the output we want to predict with one of our tasks and is therefore also known as the "target class". It contains 20 different nationalities ranging from Jamaican Japanese. The remaining 64 attributes are respectively the ingredients for all the cuisines, starting from "ingredients/0" and ending with "ingredients/64" with content such as "ground ginger" and "mirin".

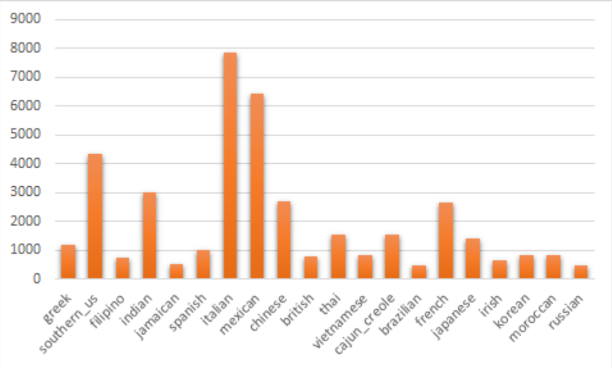

Figure 1: The distribution of each type of cuisine in the original dataset

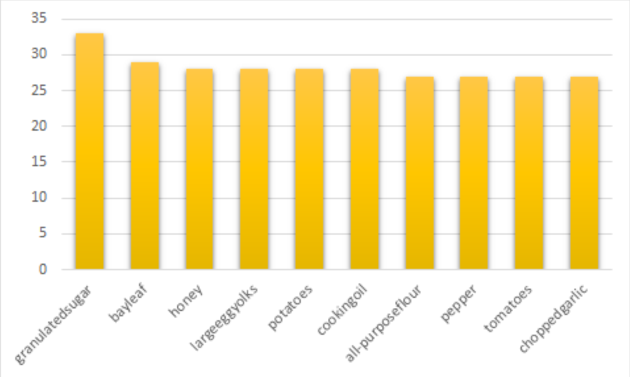

Figure 2: The top 10 most used ingredients

Furthermore, we were interested in knowing the most used ingredients, which is displayed in the chart below.

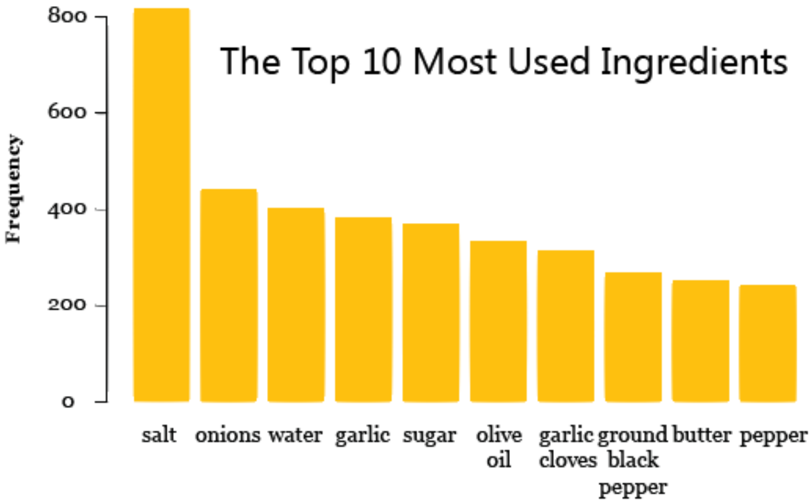

Figure 3: The top 10 most used ingredients

Name	Top 3 Ingredients			Name	Top 3 Ingredients		
brazilian	salt	onions	oliveoil	italian	oliveoil	salt	garlic
british	salt	butter	all-purpose flour	jamaican	salt	ground allspice	onions
cajun creole	salt	onions	garlic	japanese	soysauce	salt	mirin
chinese	soysauce	sesameoil	cornstarch	korean	soysauce	sesame oil	sugar
filipino	salt	garlic	onions	mexican	salt	jalapeno chilies	garlic
french	salt	sugar	oliveoil	moroccan	oliveoil	salt	ground cumin
greek	salt	oliveoil	garlic cloves	russian	salt	onions	sugar
indian	salt	onions	water	southern us	salt	all-purpose flour	butter
irish	salt	butter	all-purpose flour	spanish	salt	oliveoil	onions
vietnamese	fishsauce	sugar	water	thai	fishsauce	garlic	water

Table 1: The top 3 ingredients by cuisine

Based on the initial analysis and our knowledge about the various models, we have settled on firstly trying out the association rule mining and secondly, we will continue with classification models. Our hypothesis is that association rule mining will be able to provide us the answer based on the sequence of ingredients, which can be related to how Amazon uses it for market basket patterns, but instead of predicting an item for a customer to buy, we predict which category the cuisine is.

We have initially chosen two models, as it is important for us to be able to make a comparison of which model works best. Should our initial hypothesis turn out not to be right at all, which will be shown in the result from the analysis, we might need to test a third model.

Preprocessing

After having done the descriptive analysis we go about conducting the preprocessing of the dataset, with the respective algorithms in mind. First, we will go through a general preprocessing which was applied for both methods and afterwards, we will go in depth with the further preprocessing for each respective algorithm. The general preprocessing means using the same part of the dataset and doing a similar general preprocessing but besides this, the preprocessing will be different, as each model needs different inputs.

General preprocessing

First, we tried to load the dataset in Weka with the Rweka package, but as we could only load the data but not generate any models, we decided that we need to choose an approach for downsizing the dataset. We chose a method where we percent-wise cut down (10%, 20%, 30%, 40%, 50%) the data set, and tested the building of models after each cut down. But still we were not able to build models. We therefore went with a second method, which was to cut down the data set so it only contained the first 100 rows of each cuisine. As the dataset is originally randomly organized, the choosing of the rows were already randomized.

Secondly, we removed all the “ “ (space), “““, “-”, “/” and “_” from the downsized data set. The reason for this being that this was a necessary step before converting the data to record data, if not we would have two unique ingredients per every ingredient as some of the ingredients are the same, but with different spelling like “Soy Sauce” and “Soy-Sauce”, which would be incorrect.

Third, we removed the first attribute, the ID column, as this could result in a overfitting issue.

Preprocessing for classification

For classification it is important that we have a similar process for all the classification algorithms, so we will be able to make a valid comparative analysis.

As a first step, we converted the data set with R code[1] to record data, as Weka cannot directly read the transaction data to run classification. This resulted in a change scaling the data set from having 67 attributes to 2318, this changed the data set to be in a binary form, where the first attribute is the “cuisine”, also known as the target class, and the following attributes were ingredients by name, such as “salt” and “pepper”. We did not change any attribute types, as all of them were initially ‘nominal’.

For the comparing of the effect of the preprocessing, we tried various approaches. In the first approach, we did not change anything, to create a baseline for our testing. For the second approach, we deleted the most frequent item “salt”, as this appeared in more than 50% of the instances, which could potentially skew the algorithm, but the accuracy did not change. As a third approach, we used info gain as a parameter, but as there was only 104 attributes out of 2317 attributes which had an info gain being “> 0”, we just kept them and ran the “Naive Bayes[2]” algorithm again and surprisingly saw that the accuracy went down. We therefore went back to our first approach, keeping all of our attributes, including “salt”.

[1] R code uploaded separately to Blackboard

[2] Overall Naïve Bayes had the best accuracy, which is why we used that for testing the second and third preprocessing approach.

Preprocessing for association rules mining

There are two steps in this process. First, we loaded the transaction data into R and ran the association rules. From this, we found that almost all of the rules for each cuisine included a combination of the cuisine and the ingredient "salt". As "salt", is the most popular ingredient in the whole data set, it should not be counted in the association rules as it will skew the dataset, and therefore we deleted "salt" in the whole data set and ran the association rules mining again.

Analysis

Classification

As for classification, we set out to start with the decision tree algorithm (DT), hereafter continue with random forest, naïve bayes, k nearest neighbour (kNN) and support vector machines (SVM) and changed different parameters to get the best accuracy following by other algorithms. Each algorithms we conducted with cross-validation set to 5 folds. The analysis is summarized in the following table.

j48	**default**	**40.25%**
j48	BinarySplits= True	39.50%
j48	ConfidenceFactor= 0.1	39.05%
j48	unpruned = True	38.80%
Random Forest	default	49.15%
Random Forest	seed=100	48.30%
Random Forest	breakTiesRandomly= True	49.25%
Random Forest	debug = True	49.15%
NaiveBayes	default	53.05%
NaiveBayes	useKernelEstimator = True	53.05%
NaiveBayes	useSupervisedDescritization = True	38.10%
NaiveBayes	doNotCheckCapabilities	53.05%
KNN	default	23.70%
KNN	KNN= 3	22.40%
KNN	crossValidate = True	23.70%
KNN	nearestNeighbourSearchAlgorithm = KDTree	23.70%
KNN	nearestNeighbourSearchAlgorithm=filteredNeighbourSearch	23.70%
SVM	default	48.90%
SVM	c= 0.01	36.95%
SVM	c= 0.2	50.55%
SVM	c = 0.5	49.02%
SVM	filterType=standardize training data	49.02%

Table 2: Classification analysis result

Association Rule Mining

As for association rules mining, the first thing we need to decide is the confidence and support. To calculate this, we have to recall that the data set has 100 rows for each cuisine, and 20 cuisines which in total gives us 2000 rows. The support therefore be 0.0005 if the ingredients used only one time in one cuisine (1/2000). We need to choose a starting point that is above this, so we choose to set the support at 0.001 at first.

We found that when we changed the support to 0.01 and confidence to 0.3, all the correlation between LHS and RHS are above 1. This gives us the best result. Thus, we decided to use this settings.

Support	Confidence	Number of Rules
0.001	0.1	367
0.005	0.1	367
0.01	0.1	106
0.01	0.3	36
0.01	0.5	2

Table 3: Association rule mining setting

The result is as follows:
The table and graph shows the 36 rules we generate from association rules mining. The "lhs" gave us the cuisine that we want to cook, "rhs" gave the ingredients that we should buy, the support means the possibility this combination appears, "confidence" means how strong this rule is, and "lift" tell us the correlation between "lhs" and "rhs".
Using the first instance as an example, if we want to cook Brazilian food, we should buy onion. The possibility of this combination appearing in the data set is 0.016, The confidence 0.32 tells us this rule is strong comparing with other rules in the results. The "lift" 1.454545 is above 1, which means, Brazilian and onions is related with each other.

lhs		rhs	support	confidence	lift
1 {brazilian}	=>	{onions}	0.016	0.32	1.45455
2 {british}	=>	{butter}	0.017	0.34	2.70916
3 {british}	=>	{allpurposeflour}	0.015	0.3	2.91262
4 {cajuncreole}	=>	{onions}	0.017	0.34	1.54546
5 {chinese}	=>	{soysauce}	0.023	0.46	4.08889
6 {chinese}	=>	{sesameoil}	0.019	0.38	6.12903
7 {chinese}	=>	{cornstarch}	0.018	0.36	8.37209
8 {chinese}	=>	{sugar}	0.016	0.32	1.72973
9 {chinese}	=>	{garlic}	0.016	0.32	1.67102
10 {chinese}	=>	{water}	0.0155	0.31	1.53846
11 {filipino}	=>	{garlic}	0.022	0.44	2.29765
12 {filipino}	=>	{onions}	0.0205	0.41	1.86364
13 {filipino}	=>	{water}	0.02	0.4	1.98511
14 {filipino}	=>	{soysauce}	0.017	0.34	3.02222

15 {french}	=>	{sugar}	0.015	0.3	1.62162
16 {greek}	=>	{oliveoil}	0.0195	0.39	2.34234
17 {indian}	=>	{onions}	0.0205	0.41	1.86364
18 {irish}	=>	{butter}	0.0185	0.37	2.94821
19 {irish}	=>	{allpurposeflour}	0.0155	0.31	3.00971
20 {italian}	=>	{oliveoil}	0.02	0.4	2.4024
21 {jamaican}	=>	{groundallspice}	0.016	0.32	2.54902
22 {jamaican}	=>	{onions}	0.0155	0.31	1.40909
23 {jamaican}	=>	{garlic}	0.015	0.3	1.56658
24 {japanese}	=>	{soysauce}	0.016	0.32	2.84444
25 {korean}	=>	{soysauce}	0.027	0.54	4.8
26 {korean}	=>	{sesameoil}	0.0245	0.49	7.90323
27 {korean}	=>	{sugar}	0.021	0.42	2.27027
28 {korean}	=>	{garlic}	0.0185	0.37	1.93212
29 {korean}	=>	{greenonions}	0.0165	0.33	3.62637
30 {korean}	=>	{sesameseeds}	0.015	0.3	3.63636
31 {moroccan}	=>	{oliveoil}	0.023	0.46	2.76276
32 {moroccan}	=>	{groundcumin}	0.017	0.34	5.76271
33 {moroccan}	=>	{onions}	0.0155	0.31	1.40909
34 {moroccan}	=>	{garliccloves}	0.015	0.3	1.90476
35 {southernus}	=>	{allpurposeflour}	0.016	0.323232	3.13818
36 {spanish}	=>	{oliveoil}	0.022	0.44	2.64264

Table 4: Association rule mining result

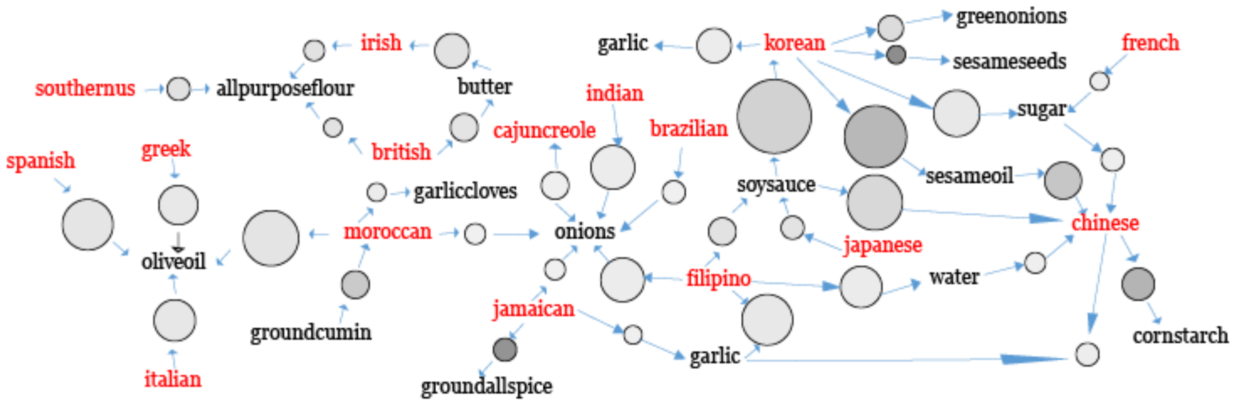

Figure 4: Visualization of association rule mining result

Concluding remarks

Based on this report, we can conclude the following:

- We can conclude that computers can distinguish different types of cuisines based on the ingredients. In our case, Naïve Bayes turned out to be the best classification algorithm for our dataset, even though our initial hypothesis was that the random forest algorithm would work the best, as this algorithm combines several algorithms in one. This might be because Naïve Bayes is good with independent attributes and our dataset's attributes are independent ingredients which are not correlated to each other. Also Naïve Bayes performs well when dependencies of features from each other are similar between features which is true about our dataset attributes (recipes). (Ximing et al, 2015)
- As for association rule mining, we can conclude that we can state which ingredients tend to appear together, as seen in the above example. The algorithm depends on the support and confidence we set as well as the question we ask.
- Though we cannot compare the two models, we can conclude that there is not always one right approach for a data set, it depends on the question you ask.
- Using R with "arules" package, we can only get one item on the RHS as the result. We might learn more about programming to generate rules of combination ingredients for each cuisine.
- A final reflection as a learning outcome, is the that we have experienced how decision making in data mining is very subjective and how important it is with a structured approach, which is initiated with a hypothesis.
- We later found that similar data, as used in this report, is already being used by IBM who created "Chef Watson", which is being used for what IBM calls "cognitive cooking" (IBM Chef Watson, 2015).

References

Convert CSV, 2015. *Convert JSON to CSV.* [online] Available at: < http://www.convertcsv.com/json-to-csv.htm> [Accessed 15 November, 2015].

IBM Chef Watson, 2015. *Ready to do some cognitive cooking?* [online] Available at: <www.ibmchefwatson.com> [Accessed 5 December, 2015]

Kaggle, 2015a. *Kaggle*. [online] Available at: <www.kaggle.com> [Acessed 12 November, 2015].

Kaggle, 2015b. *What's Cooking?*. [online] Available at: <www.kaggle.com/c/whats-cooking> [Accessed 12 November, 2015].

Kaggle, 2015c. *What's Cooking?* [online] Available at:<www.kaggle.com/c/whats-cooking/data> [Accessed 15 November, 2015].

R Project, 2015. *About*. [online] Available at: <https://www.r-project.org/about.html> [Accessed 17 November, 2015].

Weka, 2015. *Weka 3: Data Mining Software in Java.* Available at: <http://www.cs.waikato.ac.nz/ml/weka/> [Accessed 15 November, 2015].

Ximing Li, Jihong Ouyang & Xiaotang Zhou, (2015) *A kernel-based centroid classifier using hypothesis margin*. Journal of Experimental & Theoretical Artificial Intelligence.

Yummly, 2015. *Yummly*. [online] Available at: <http://www.yummly.com/> [Accessed 15 November, 2015].