



# No Free Lunch

Leon Chen, Andrew Reuben,  
Daniel Lai, Samruddhi Somani



# Agenda

- Context
- Exploratory Data Analysis
- Design Choices
- Models
- Takeaways



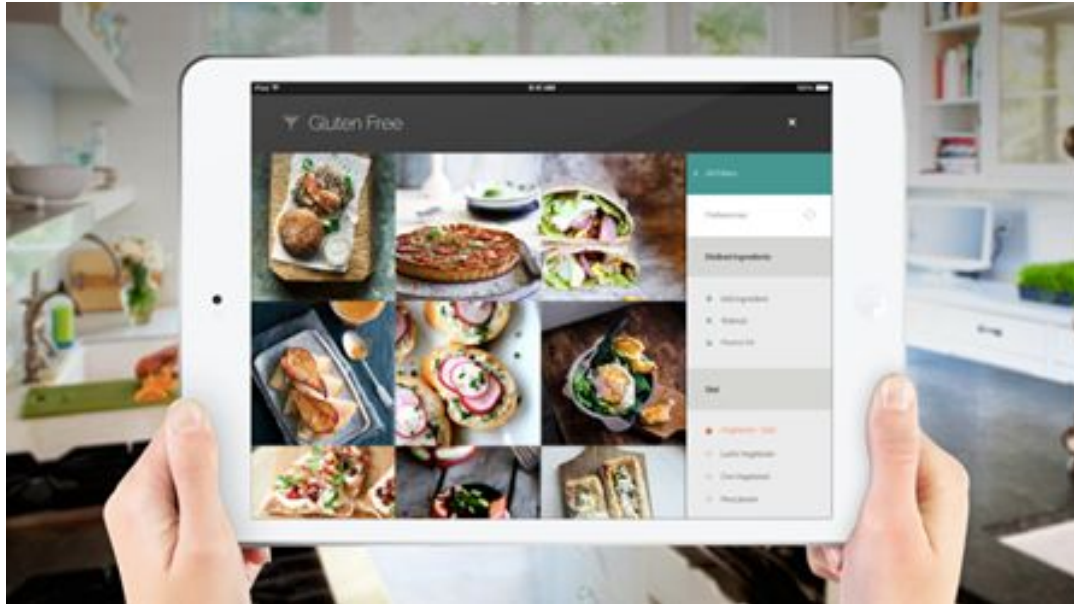
# Context

# Background

- Yummly is a website that:
  - Asks for your cuisine preferences
  - Gathers recipes across the web
  - Presents you with curated recommendations
- Similar to Pinterest, but targeted especially towards food!

# Question

- Given just the ingredients, can cuisines be predicted?



# Data

39,744 recipes in training set

9,944 in test set

Average of 7 ingredients per recipe

6,714 unique ingredients

# Samples

## TRAIN

```
{
  "id": 10259,
  "cuisine": "greek",
  "ingredients": [
    "romaine lettuce",
    "black olives",
    "grape tomatoes",
    "garlic",
    "pepper",
    "purple onion",
    "seasoning",
    "garbanzo beans",
    "feta cheese crumbles"
  ]
}
```

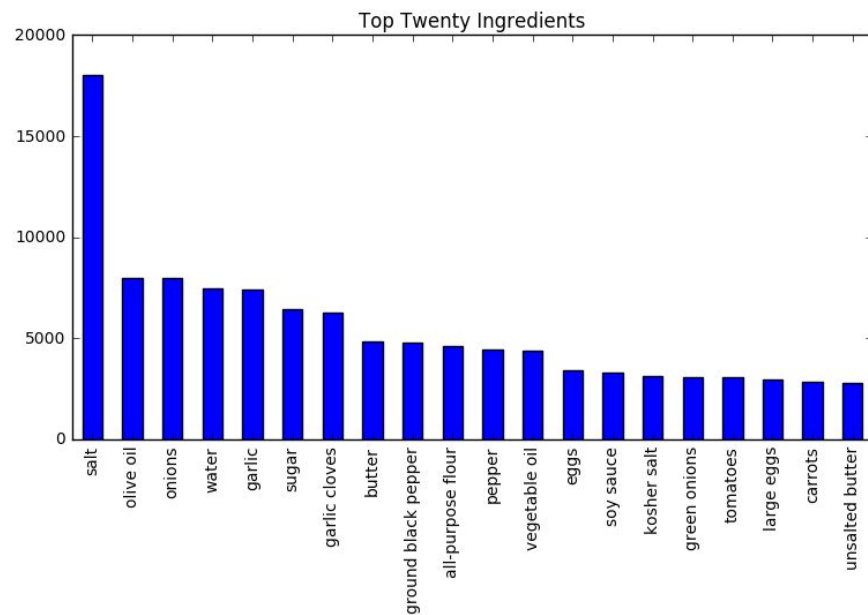
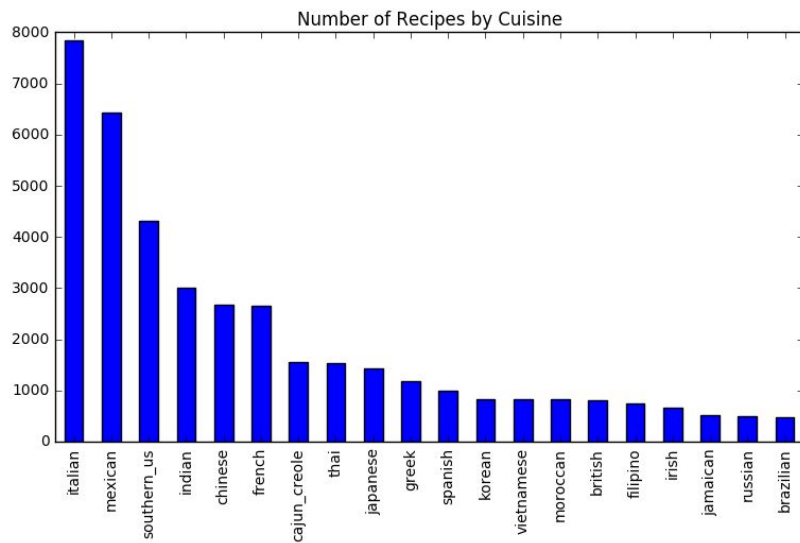
## TEST

```
{
  "id": 18009,
  "ingredients": [
    "baking powder",
    "eggs",
    "all-purpose flour",
    "raisins",
    "milk",
    "white sugar"
  ]
}
```

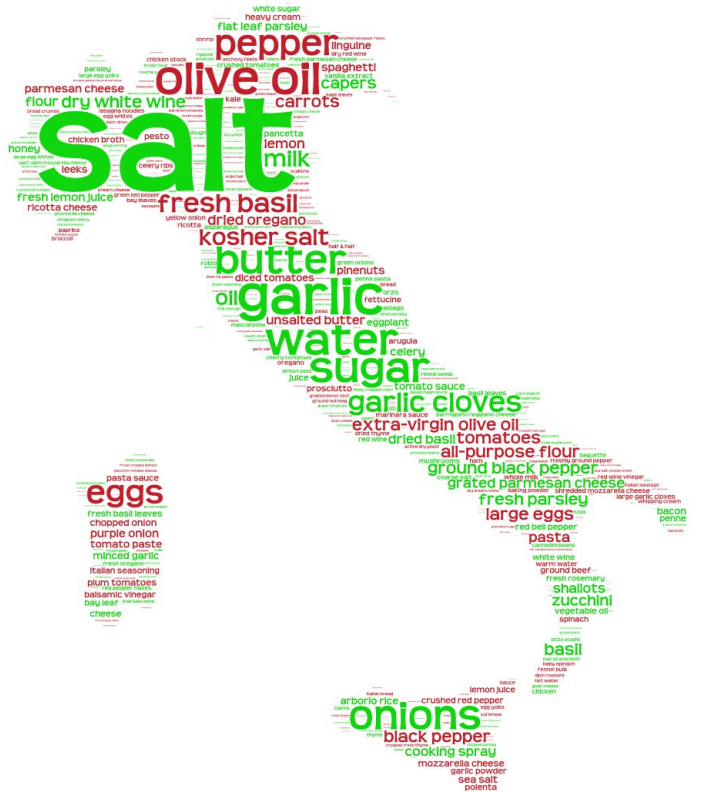
# Exploratory Data Analysis

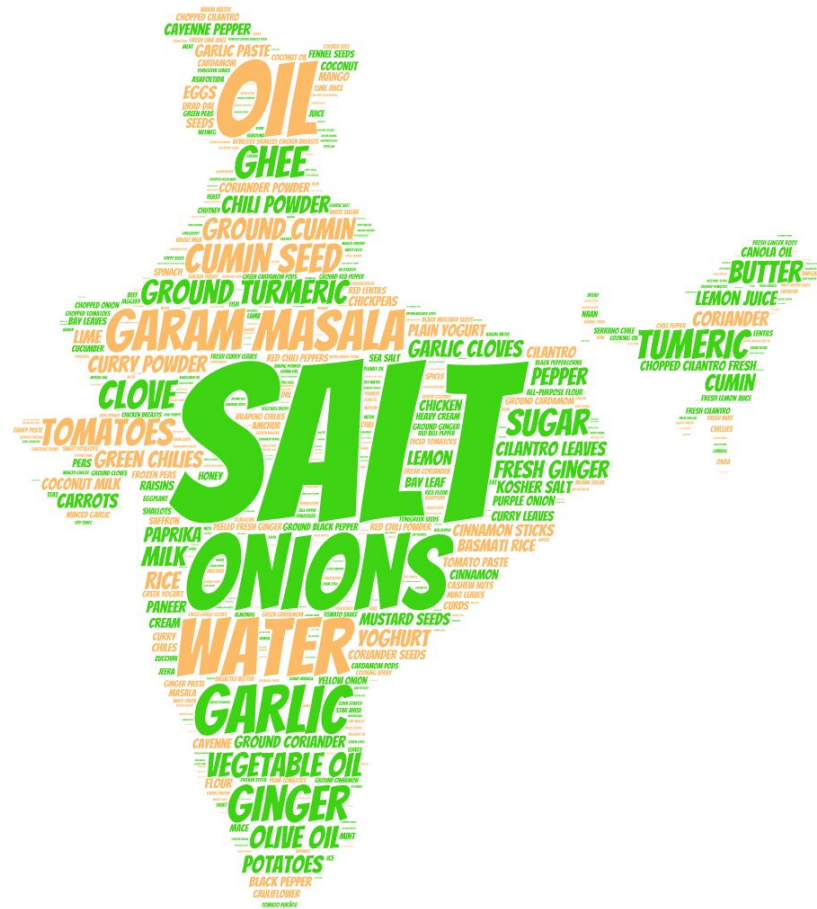


# Aggregates

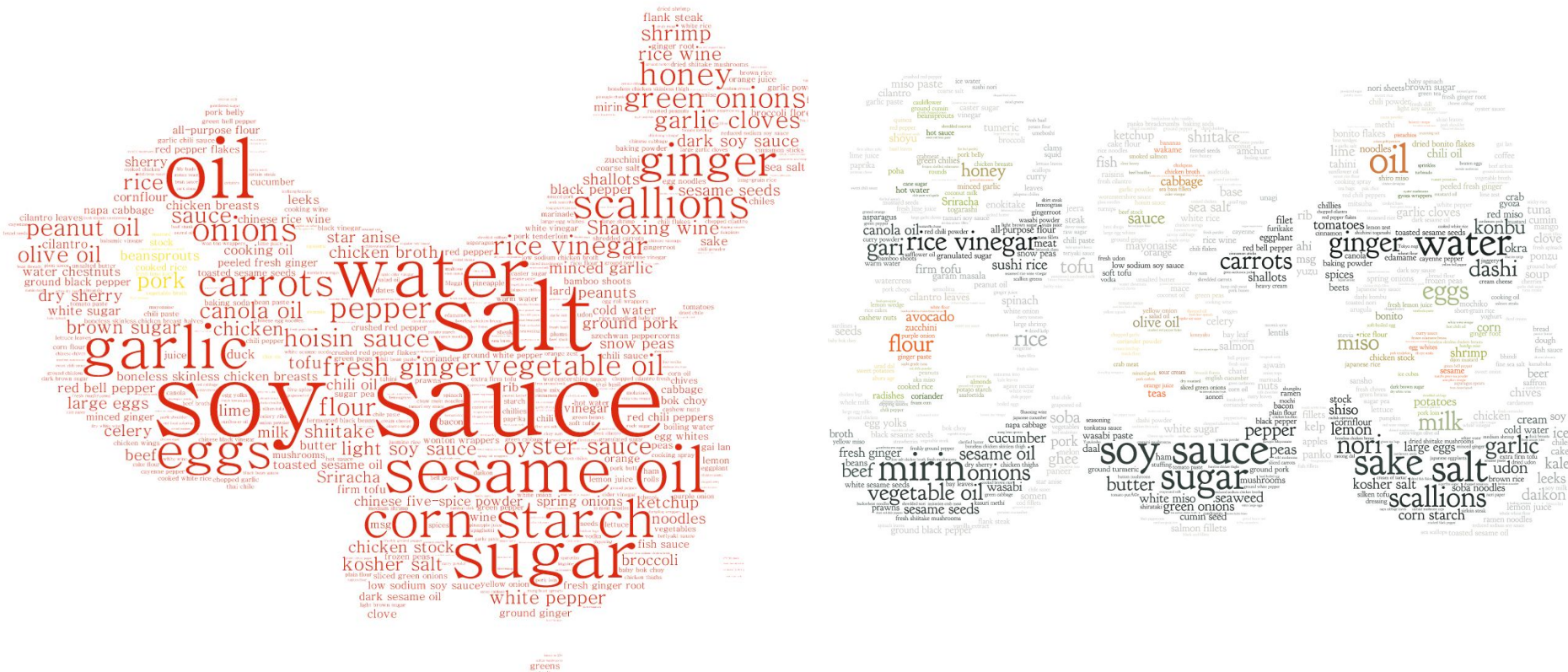


# Cuisines

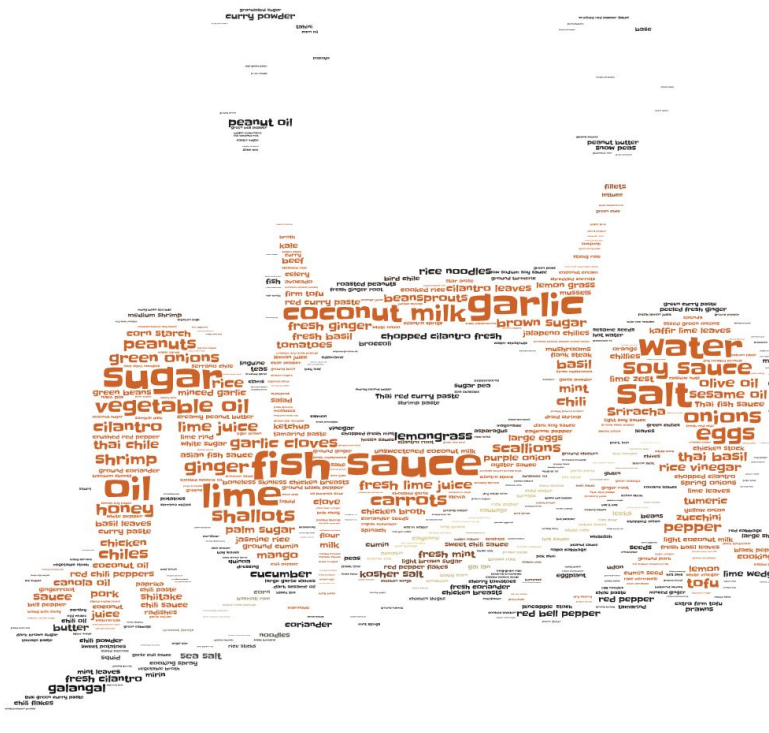


[illegible]

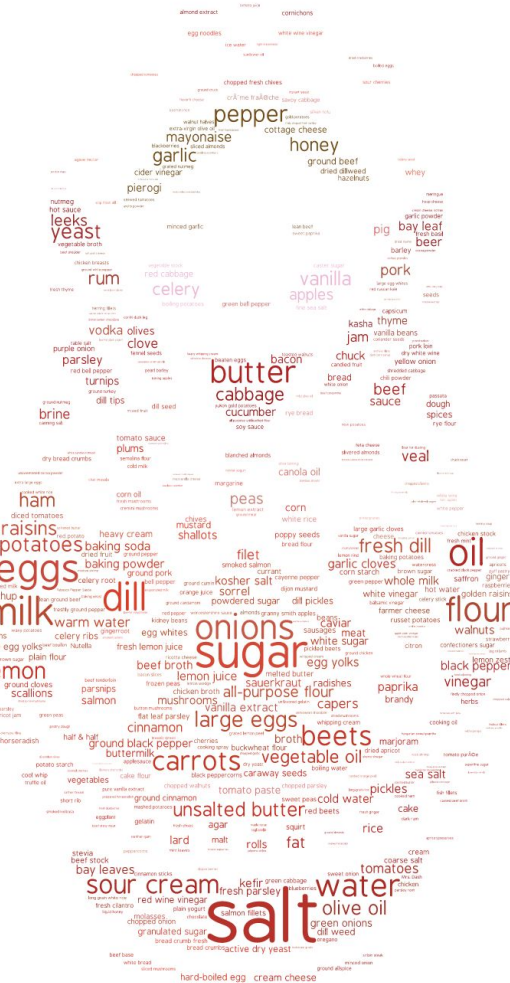
# Cuisines





[illegible]

# Cuisines





# Design Choices



# Bag of “Words”

- List of ingredients unstructured
  - order of ingredients not important
  - no grammatical information lost
  - meaning of one ingredient not affected by adjacent ingredient



# Tokenization

- **Keep original tokens**
  - **Best performance**
- Tokenize by word
  - no improvement over original tokenization
- Lemmatize each original token
  - poor performance in initial testing

```
[u'romaine lettuce',  
u'black olives',  
u'grape tomatoes',  
u'garlic',  
u'pepper',  
u'purple onion',  
u'seasoning',  
u'garbanzo beans',  
u'feta cheese crumbles']
```

# Document Term Matrix

- Scaling
  - **Original Counts**
  - TF-IDF Scaling
- Minimum Documents: 5 to filter spelling errors and uncommon ingredients

# Models

# Establishing a Baseline

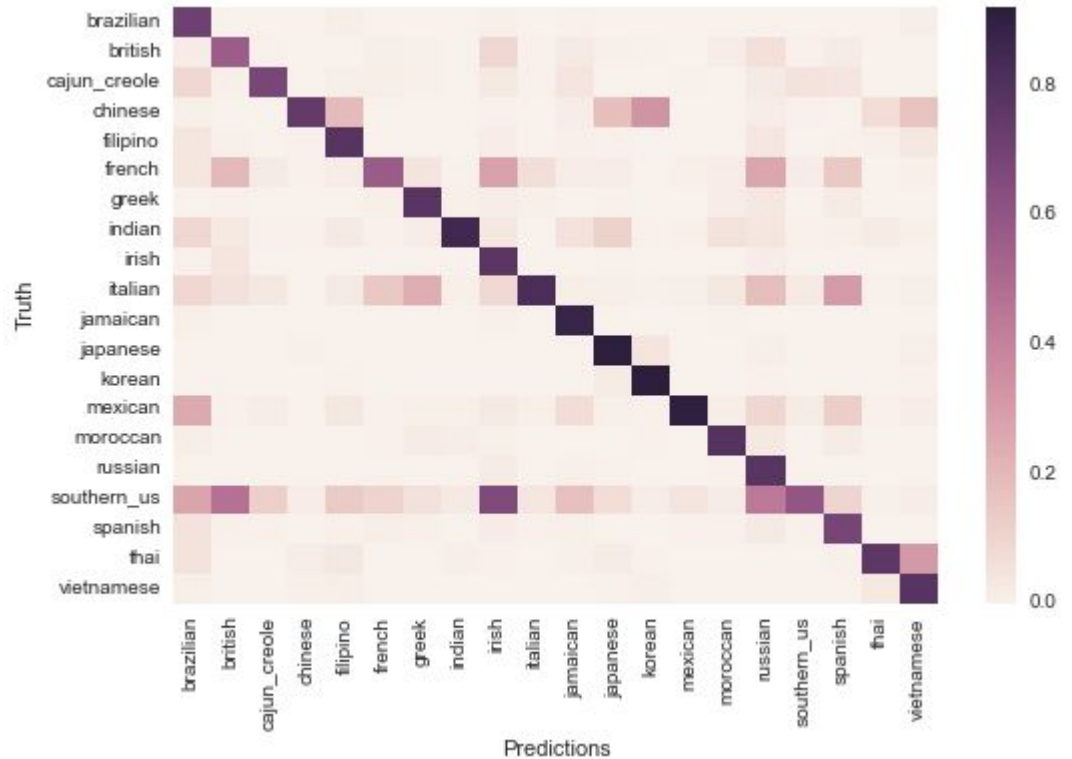
- Null Model: 19% (predict Italian for everything)

# Heatmap Interpretation

- Heatmaps are adjusted to reflect different proportions of each cuisine
- Dark purple diagonal reflects observations where our predictions match the truth--where we are right
- Darker vertical stripes indicate cuisines which are overpredicted.
- Darker horizontal stripes indicate cuisines with which we have particular difficulty.
- Included percentages are test set accuracies.

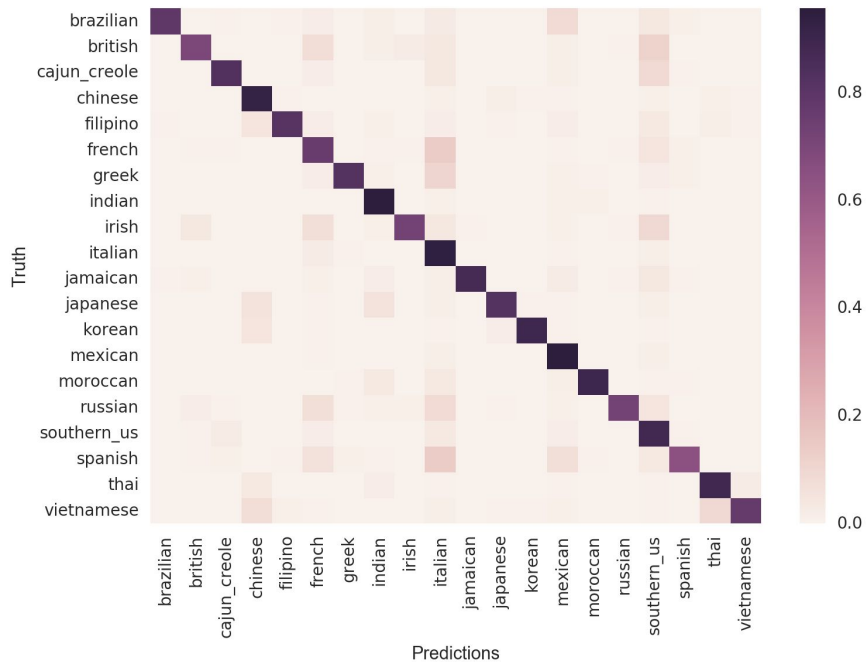
# Naive Bayes

- 73% accuracy on test set

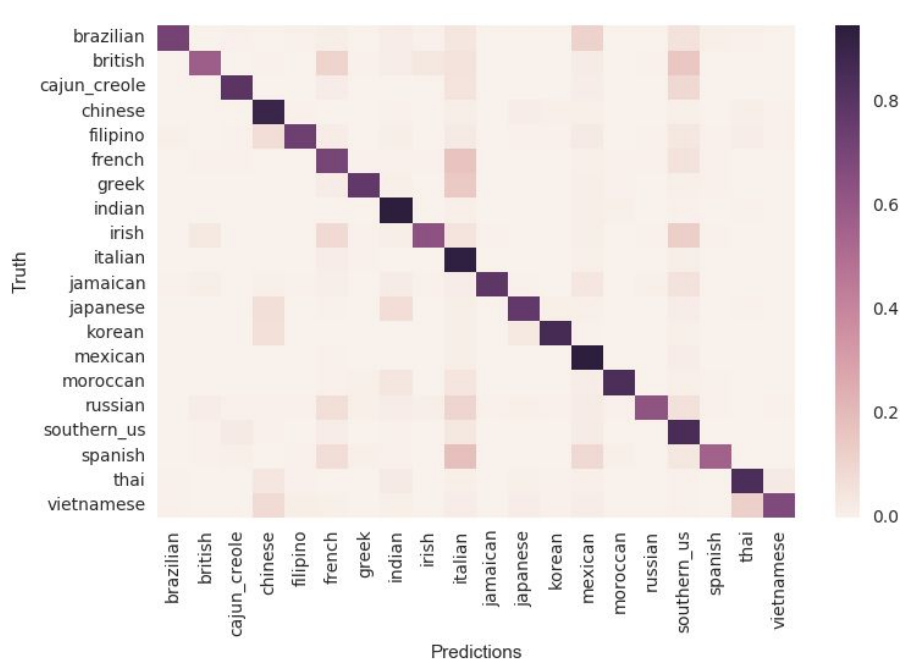


# Linear Models

- Logistic Regression: 78%

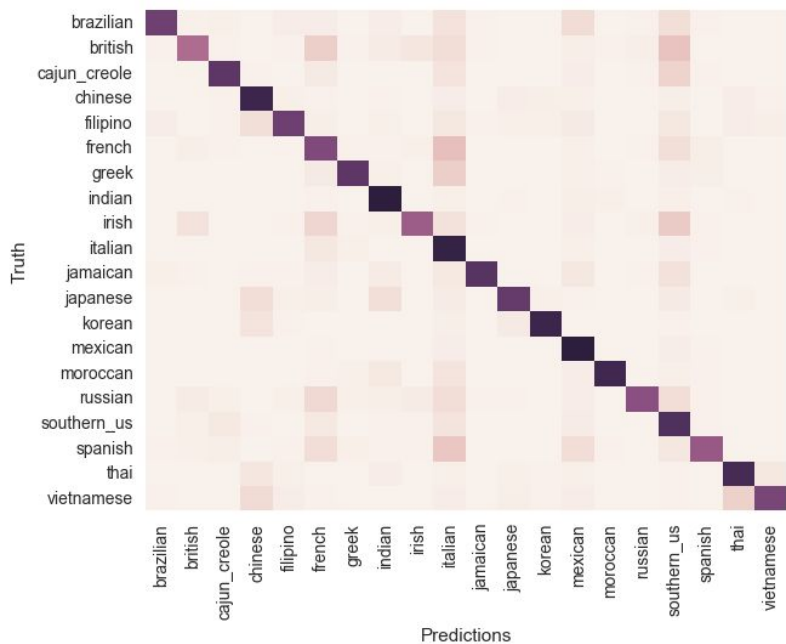


- Support Vector Machines: 78%

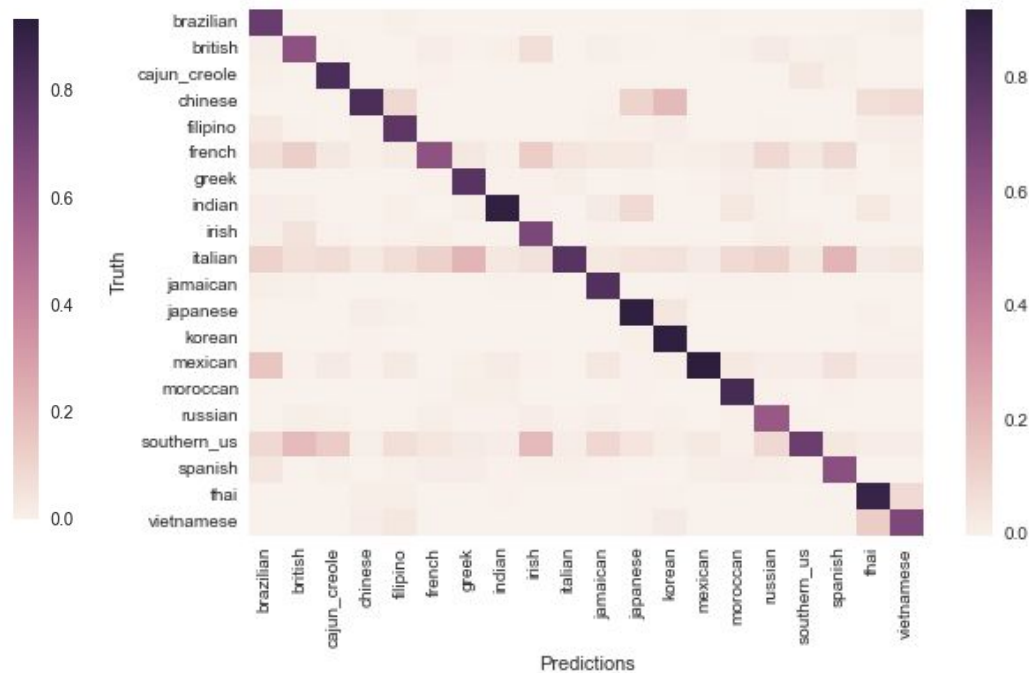


# Dimensionality Reduction Methods

- Latent Semantic Indexing: 71%



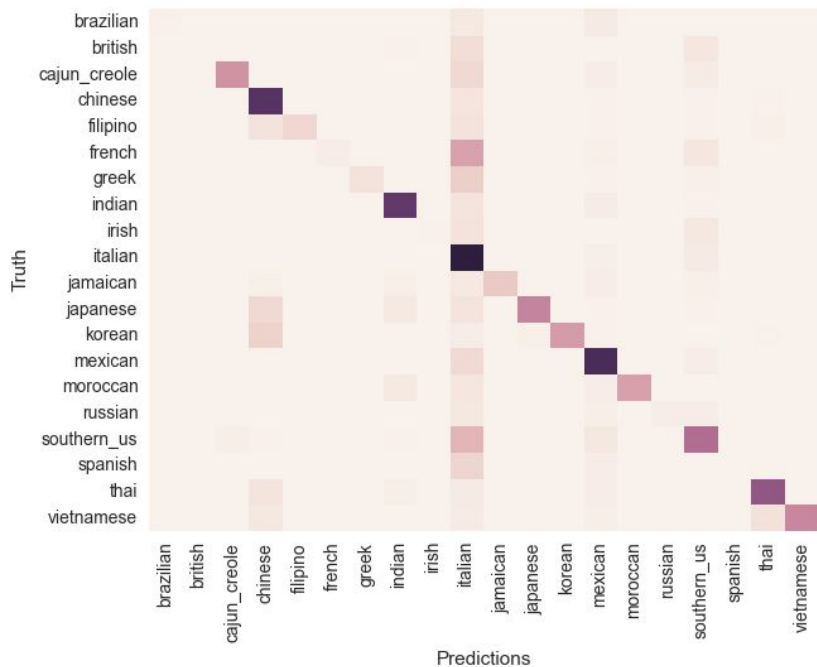
- Linear Discriminant Analysis: 74%



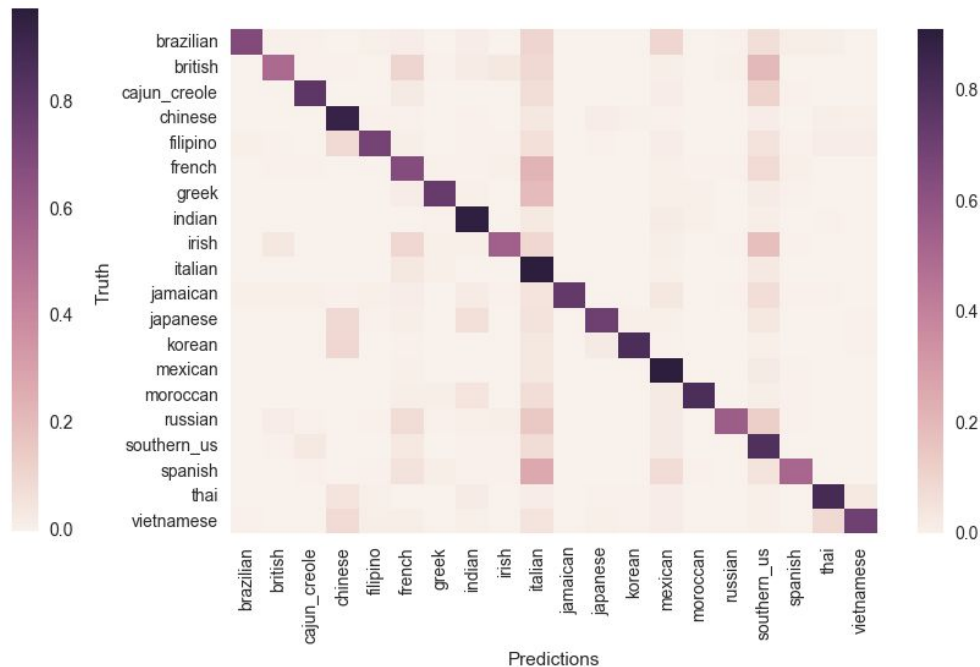


# Tree Based Ensembles

- Random Forests: 55%

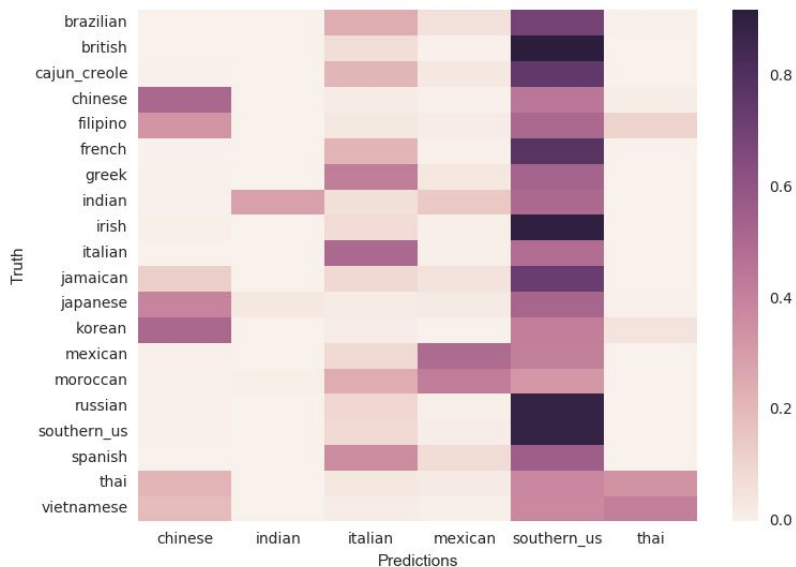


- Gradient Boosting: 76%



# Cooperative Ensemble

- Use decision tree to split dataset into subsets (10 leaves)
- Fit and tune models for each subset
- 35% accuracy--each model always predicted majority class of its node



# Competitive Ensemble

- Individual model accuracies vary between 71% and 78%
- 67% unanimity
- Algorithm: Pick mode. Pick higher prior if tie
- 79% accuracy

XGB	76%
LSI->Logistic Regression	71%
Logistic Regression	78%
SVM	78%
Naïve Bayes	73%
LDA	74%

# Takeaways

# Conclusions

- Yes, with roughly 78% accuracy
- Ingredient amounts and order would be useful
- Top competitors are about 82%
- Are cuisine distinctions meaningful when we're recommending dishes?

# Future Plans

- Custom weights on classes
- Neural networks
- Tune competitive ensemble

# Shoutouts

- Scikit-Learn
- Scipy Sparse Matrices
- XGBoost
- Seaborn
- Tagul