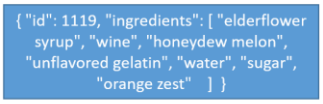
Capstone Project Proposal

Use recipe ingredients to categorize the cuisine

- **Domain Background**
  - The topic is “use recipe ingredients to categorize the cuisine”, which is from **Kaggle**: https://www.kaggle.com/c/whats-cooking-kernels-only
  - Domain is **Cooking**.
- **Problem Statement**
  - Given a list of ingredient of a cuisine, predict the category of the cuisine, such as Indian, Korean, etc.
- **Datasets and inputs**
  - There are two datasets, training dataset and testing dataset.
    - The training dataset contains around 40000 instances, each instance is data of a cuisine, containing cuisine id, cuisine label (the category of the cuisine), and a list of ingredient of the cuisine.
    - The testing dataset contains around 10000 instances, each instance is data of a cuisine, containing cuisine id, and a list of ingredient of the cuisine (There is no label in testing dataset.)
  - Both training dataset and testing dataset is JSON format.
  - The difference between training dataset and testing dataset is that training dataset has labels, but testing dataset has no label.
- **Solution statement**
  - The topic is multi-class classification, and I will use supervised machine learning to build model to predict.
    - **Preprocessing**
      - Make each ingredient be a feature. The value of the feature is zero if the cuisine does not use the ingredient, and the value of the feature is one if the cuisine use the ingredient. For example, if a list of ingredient of a cuisine do not contain ‘cream’, the value of the feature ‘cream’ of this instance is zero.
      - Example of a cuisine:



id	elderflower syrup	wine	honeydew melon	unflavored gelatin	water	sugar	orange zest	· · ·	garlic
1119	1	1	1	1	1	1	1	· · ·	0

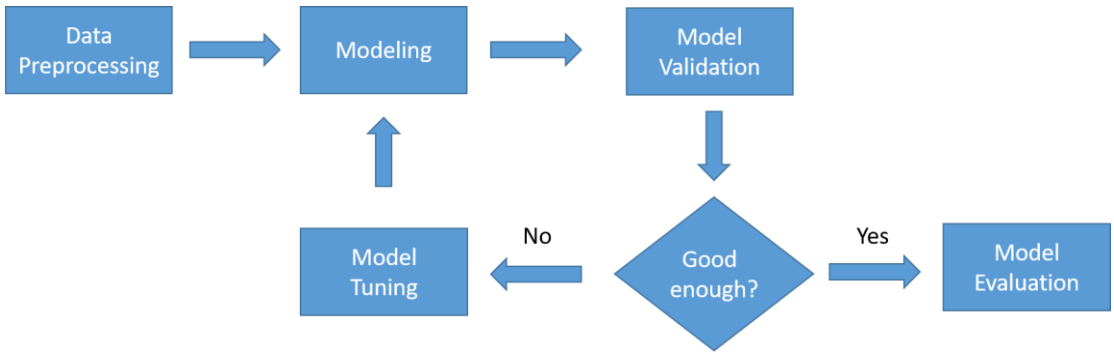
- **Modeling**
  - Use dataset with labels to build model. Use model and data without label to predict the label of the data.
  - I would like to try ensemble methods, especially decision tree-based methods, such as random forest and gradient boosting decision trees.

- **Benchmark model**
  - I would like to try methods which are not ensemble methods, such as a single decision tree model.

- **Evaluation metrics**
  - **Dataset**
    - The testing dataset provided by Kaggle has no label. I cannot evaluate model by testing dataset locally, and I would try to submit to Kaggle to evaluate. If this way cannot work, I would use part of training dataset to evaluate model locally.
    - I would keep part of training dataset as testing dataset in beginning. If it turns out that I can submit to Kaggle to evaluate, I would use whole training dataset to train and validate.
  - **Metrics**
    - Accuracy
      - $Accuracy = \frac{number\ of\ instances\ which\ are\ correctly\ predicted}{number\ of\ all\ instances}$
    - Average Prediction
      - The topic is multi-class classification. For each class, calculate a Prediction value of the class. And calculate average of Prediction in all classes.
      - $Prediction = \frac{number\ of\ true\ positive}{number\ of\ true\ positive\ +\ number\ of\ false\ positive}$
    - Average Recall
      - The topic is multi-class classification. For each class, calculate a Recall value of the class. And calculate average of Recall in all classes.
      - $Recall = \frac{number\ of\ true\ positive}{number\ of\ true\ positive\ +\ number\ of\ false\ negative}$

- Average F1-Measure
  - The topic is multi-class classification. For each class, calculate a F1-Measue value of the class. And calculate average of F1-Measue in all classes.
  - $F1 - Measure = \frac{2 \times Prediction \times Recall}{Prediction + Recall}$
- Training time
  - The elapsed time to train model.
- Testing time
  - The elapsed time to test model.

- **Outline of Project Design**
  - Overall flowchart



  - Brief step of training and testing