**Z534: Search**

**Project Report**

**Yelp Dataset Challenge**

Report by,

Suhas Jagadish

Supreeth Suryaprakash

Raghuveer Krishnamurthy

Sanjana Agarwal

Contents

[1 Task: Category prediction for Business 3](#_Toc469315929)

[1.1 Research question 3](#_Toc469315930)

[1.2 Flow diagram 3](#_Toc469315931)

[1.3 Text preprocessing 3](#_Toc469315932)

[1.4 Method/Algorithm 4](#_Toc469315933)

[1.5 Experiments and Evaluation 4](#_Toc469315934)

[2 Task: Personalization of user’s dashboard 7](#_Toc469315935)

[2.1 Research question 7](#_Toc469315936)

[2.2 Find similar users to follow 7](#_Toc469315937)

[2.3 Predict user’s favorite category 7](#_Toc469315938)

[2.4 Predict user’s favorite neighborhood 7](#_Toc469315939)

[2.5 Predict Business using HMM 7](#_Toc469315940)

[2.6 Experiments and Evaluation 7](#_Toc469315941)

[3 Conclusion 7](#_Toc469315942)

[4 Future scope 7](#_Toc469315943)

[5 References 7](#_Toc469315944)

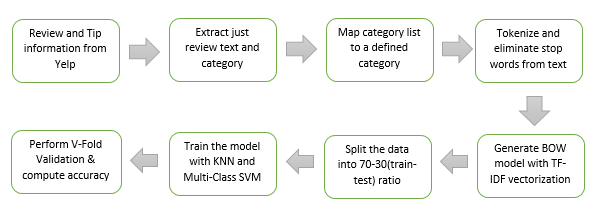
# **Task: Category prediction for Business**

## **Research question**

Each business in Yelp has a category associated with it. A business may belong to single or multiple categories. For example, ['Fast Food', 'Restaurants'] and ['Gluten-Free', 'Asian Fusion', 'Chinese', 'Restaurants'].

Yelp also has the review or tip information provided by the user. Our task is to predict the category of each business by just using the review or tip information.

## **Flow diagram**



## **Text preprocessing**

The data was in .json format and the first thing we did was to convert the json files to csv files. We then eliminated the unwanted information and extracted only the relevant data. For example, we extracted ‘review text’ and ‘business\_id’ from reviews and ‘business\_id’ and ‘category’ from business.

We then compared the business and review files to extract just the ‘Review/tip Text’ and ‘Category’ information because this is all we need to predict the business category based on review text.

When we extracted the business data, we found that the category information was a list of keywords, i.e., ['Fast Food', 'Restaurants'] or ['Gluten-Free', 'Asian Fusion', 'Chinese', 'Restaurants']. We found this a little inefficient to train the model. Hence we made a list of possible categories and a map of keywords to categories, so that we would get a defined category for each business.

For example, the list ['Gluten-Free', 'Asian Fusion', 'Chinese', 'Restaurants'] would be mapped to ‘**Chinese**’ category or ['Burgers', 'Fast Food', 'Restaurants'] would be mapped to ‘**American**’.

Hence the final csv file contained ‘review text’ and corresponding ‘defined category’ information.

## **Method/Algorithm**

Steps:

* Preprocess the data and extract relevant information using the above step.
* For text based feature extraction, we split each review, tokenize it, remove stop words and create a bag of words representation, which is used as features.
* TF-IDF values are computed for each word in a review across the bag of words and a model file is generated. The model file can be thought of as a 2D matrix where x-axis represents each review, y-axis represents the features (bag of words) and the value is the TF-IDF value of the word.
* Use the model file to train KNN and Multi-class SVM algorithms.
* Divide the dataset in the ratio 70-30 where 70% is used to train the model and 30% to test it.
* Use 5-fold cross validation to evaluate the results on the test set.
* Determine precision, recall and F1-score for each class label and finally compute the accuracy of our model.

We implemented the BOW and TF-IDF calculations from scratch, without using any existing libraries. However KNN, SVM and Cross Validation functions were imported from python sklearn package.

## **Experiments and Evaluation**

Considering the entire feature space:

* With KNN, we achieved an accuracy **65.27**% with the value of k set as 5.
* We experimented with different values of and found the highest accuracy rating with k=5. (k is the number of neighbors). The value of k is non-parametric and as [1] defines, general rule of thumb is choosing k is k=sqrt(N)/2 where N is the number of samples in the training set. However with this value of k, the observed accuracy was less than the one with k=5, hence we decided to choose the value as 5.
* With Multi-class SVM, we achieved an accuracy of **80.21**%. We decided to use the LinearSVC implementation of sklearn as this implements “one-vs-the-rest” multiclass strategy.
* The comparison between KNN and SVM is shown below –
* As seen from the above results, linear kernelized SVM outperforms KNN for text classification. This is also evident from [2] where the authors prove that SVMs are well suited for text categorization, mainly because of its high dimensional feature spaces and sparse instance vectors. Furthermore, SVMs do not require any parameter tuning, since they can find good parameter settings automatically.

After dimensionality reduction using Mutual Information (MI):

* As suggested by the Professor, we employed a feature selection or dimensionality reduction technique called **Mutual Information**, which measures the dependency between the variables and reduce the number of features.
* One of the inputs for MI is to set the number of features, say k, beforehand. The MI algorithm will compute the top k features, eliminate the other features and compute the accuracies.
* We used 4 different values for k where k = 80%, 75%, 70% and 60% of the actual number of features, which is our baseline. We found the best accuracy when k was chosen to be 75% of the baseline features.
* The comparison between KNN and SVM after dimensionality reduction is shown below –
* As seen from the above results, the accuracies has decreased, both for KNN and SVM. After the dimensionality reduction, we actually lost some useful information about the review text. While we generated the bag of words model, we already removed stop words to generate the optimized feature space but with further reduction in feature space, it is evident that we lost some useful information about the review/tip text.

# **Task: Personalization of user’s dashboard**

## **Research question**

## **Find similar users to follow**

## **Predict user’s favorite category**

## **Predict user’s favorite neighborhood**

## **Predict Business using HMM**

## **Experiments and Evaluation**

# **Conclusion**

# **Future scope**

# **References**