FINAL PROJECT
YELP DATA CHALLENGE

Hardik Chapanera, Pranab Bhadani, Ruchi Gupta Neema, Snehal Vartak

INDIANA UNIVERSITY BLOOMINGTON

TASK 1

Recommend business to users

Recommendation systems: Our Approach

<u>Customer Query</u>

Represent Customer as
Feature Vector

Food
Service
Ambience
Cost

Maximize
Cosine similarity

<u>Restaurant</u>

Restaurant 1:
Feature Vector

Food
Service
Ambience
Cost

Restaurant 2:
Feature Vector

Food
Service
Ambience
Cost

Feature Vector Selection

- Using text mining, we extracted top 15 words from the review text based on POS tagging.
- Finally we took 5 features as food, service, ambience, cost and misc.

```
Out[44]: ['place',
          'food',
          'time',
          'service',
          'pizza',
          'White',
          'order',
          'people',
          'Castle',
          'fries',
          'beer',
          'way',
          'restaurant',
          'staff',
          'experience',
          'Vegas',
          'burgers',
          'sliders',
          'menu',
          'day']
```

Food
Service
Ambience
Cost
Misc

Representation of Customer/ Restaurant as Vector from Reviews

Procedure from Review to Feature Vector

One Review → Multiple sentences → Sentiment analysis for each sentence → Predict the category to which sentence belongs

Example

Cheeseburger was delicious. Fries was awesome.

→

Sentence1
Cheeseburger was delicious

Sentence 2
Fries was awesome

→

Sentiment Sentence 1
Positive

Sentiment Sentence
Positive

→

Category Sentence 1
Food

Category Sentence 2
Food

Category Prediction for Each Sentence

Step 1: Train Word Embeddings

```
model.most_similar(["salmon"])
```

```
[(u'whitefish', 0.7713552713394165),
 (u'halibut', 0.7040623426437378),
 (u'trout', 0.7014901638031006),
 (u'scallop', 0.6834375858306885),
 (u'tuna', 0.6816917657852173),
 (u'saba', 0.6587077379226685),
 (u'hamachi', 0.6518269777297974),
 (u'swordfish', 0.6413819789886475),
 (u'mackerel', 0.6379843950271606),
 (u'tilapia', 0.6306815147399902)]
```

Step 2: Extract Noun from Sentences

Sentence

```
contentArray =['Those little \
     hamburger and cheeseburgers \
               were delicious!']
```

Noun

```
['hamburger', 'cheeseburgers']
```

Step 3: Similarity between Nouns and Features

Similarity with hamburger

```
food      -  0.811020091176
service   -  1.02561366372
ambience  -  1.09910923243
cost      -  0.930332280695
```

Similarity with cheeseburgers

```
food      -  0.841149821877
service   -  1.01916900091
ambience  -  1.02966656536
cost      -  0.89333409816
```

Step 4: Assign features for sentence

Sentence

```
contentArray =['Those little \
     hamburger and cheeseburgers \
               were delicious!']
```

Assign Feature: FOOD

Creating Restaurant Feature Vector and Customer Feature vector

Restaurant Feature Vector : Review 1

Review ID	Sentence ID	Reviews	Sentiment	food	service	ambience	cost	misc
1	1	Crave those crazy squares!!	1	1	0	0	0	0
1	2	Back home in Texas, my dad would crave them and have to settle for the frozen-aisle version.	1	1	0	0	0	0
1	3	This place is a bit of a show in the middle of the night as most people are drunk and sloppy while ordering lol.	1	0	0	1	0	0
1			**1**	**1**	**0**	**1**	**0**	**0**

Restaurant Feature vector is given by the mean of each review vector

Customer Feature vector

$$C = \frac{\sum_{i=1}^{n} w_i * R_i}{\sum_{i=1}^{n} w_i}$$

w_i: *Customer rating of restaurant i*

R_i: *Restaurant i feature vector*

Recommendation systems: Our Approach

Customer Query

Represent Customer as Feature Vector

Food
Service
Ambience
Cost

Maximize Cosine similarity

Restaurant

Restaurant 1: Feature Vector

Food
Service
Ambience
Cost

Restaurant 2: Feature Vector

Food
Service
Ambience
Cost

RMSE Value = 1.63

Algorithm 2: Collaborative Filtering Approach

1. Item Based Similarity

   – Calculate the similarity between item-item using cosine and pearson similarity.

Algorithm	Similarity	RMSE
Item Similarity Model	Cosine	3.84
KNN Basic	Cosine	1.0963
KNN With Means	Cosine	1.0312
KNN Basic	Pearson	1.1538
KNN With Means	Pearson	1.0837
Ranking Factorization		1.13671

2. User Based Similarity Evaluation -

Algorithm	Similarity	RMSE
User Similarity Model	Cosine	3.74
KNN Basic	Cosine	1.0833
KNN With Means	Cosine	1.0386
KNN Basic	Pearson	1.1374
KNN With Means	Pearson	1.0843

TASK 2

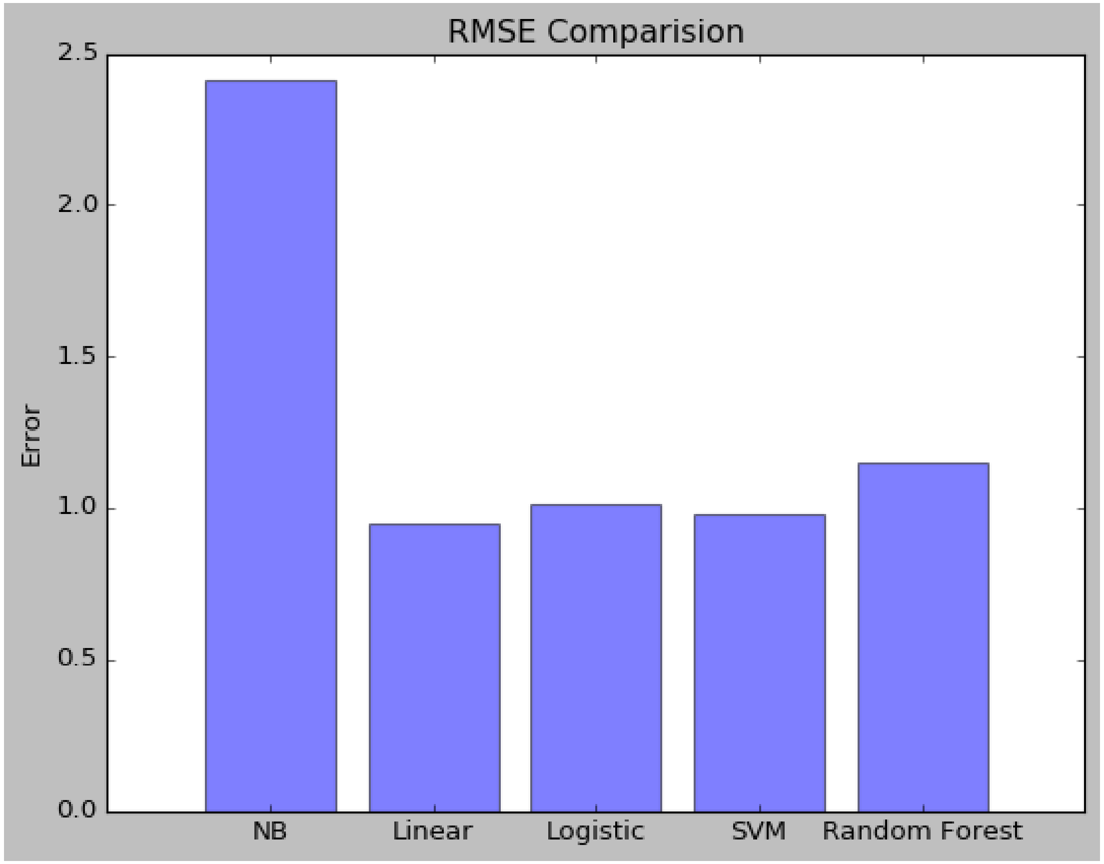
Predicting bad reviews for a Business

Machine Learning Approach

- Create features based on ratings
  - Apply classical machine learning models
  - works well with less training data.
- Applied Naive Bayes, Linear Regression, Logistic Regression, SVM and Random Forest
- Testing another model where features are tf-idf scores of reviews

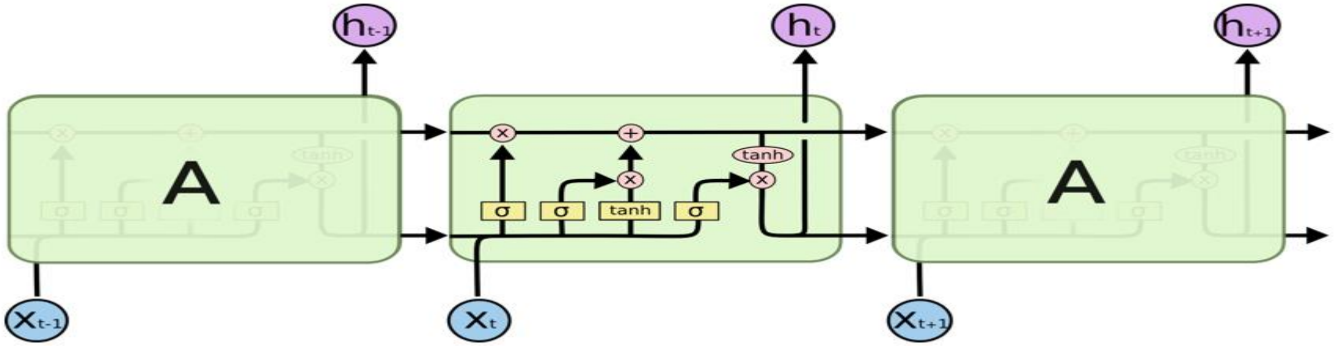
Results

RMSE Values



Approach 1

1. LSTM (Recurrent Neural Network)



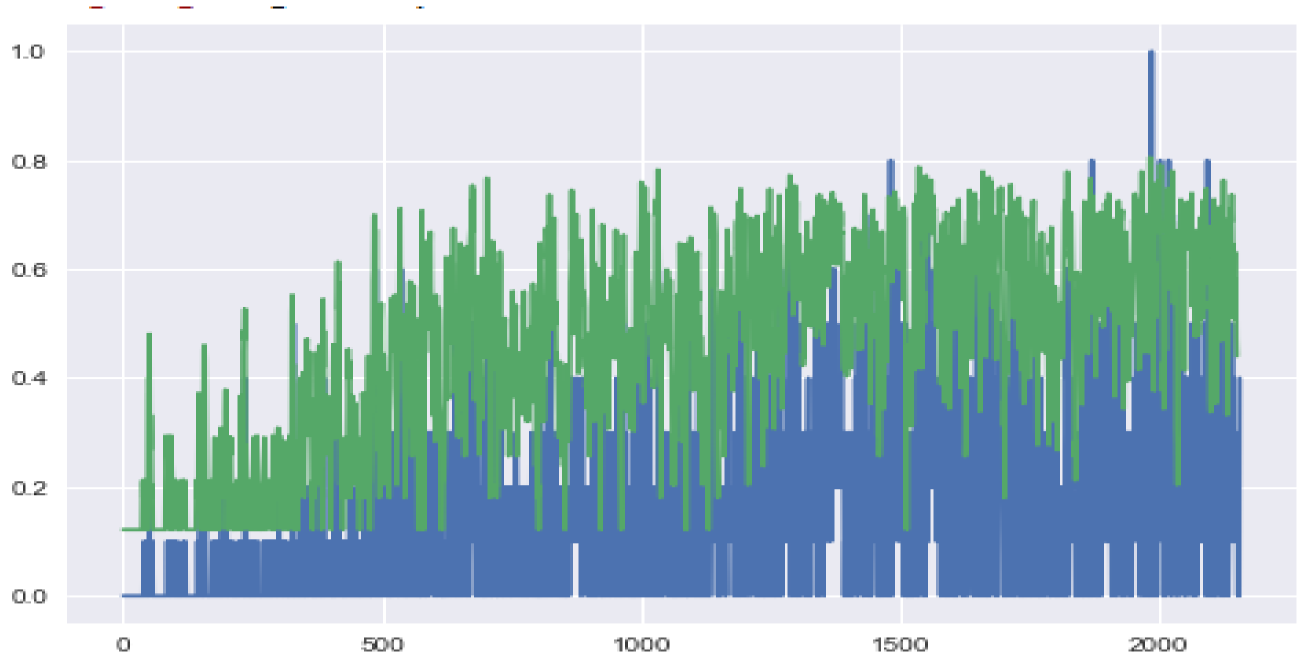
2. We preprocess the data to find number of reviews per day per Business

3. Parameter used : timestep = 4, unit = 10, hidden layer = 1

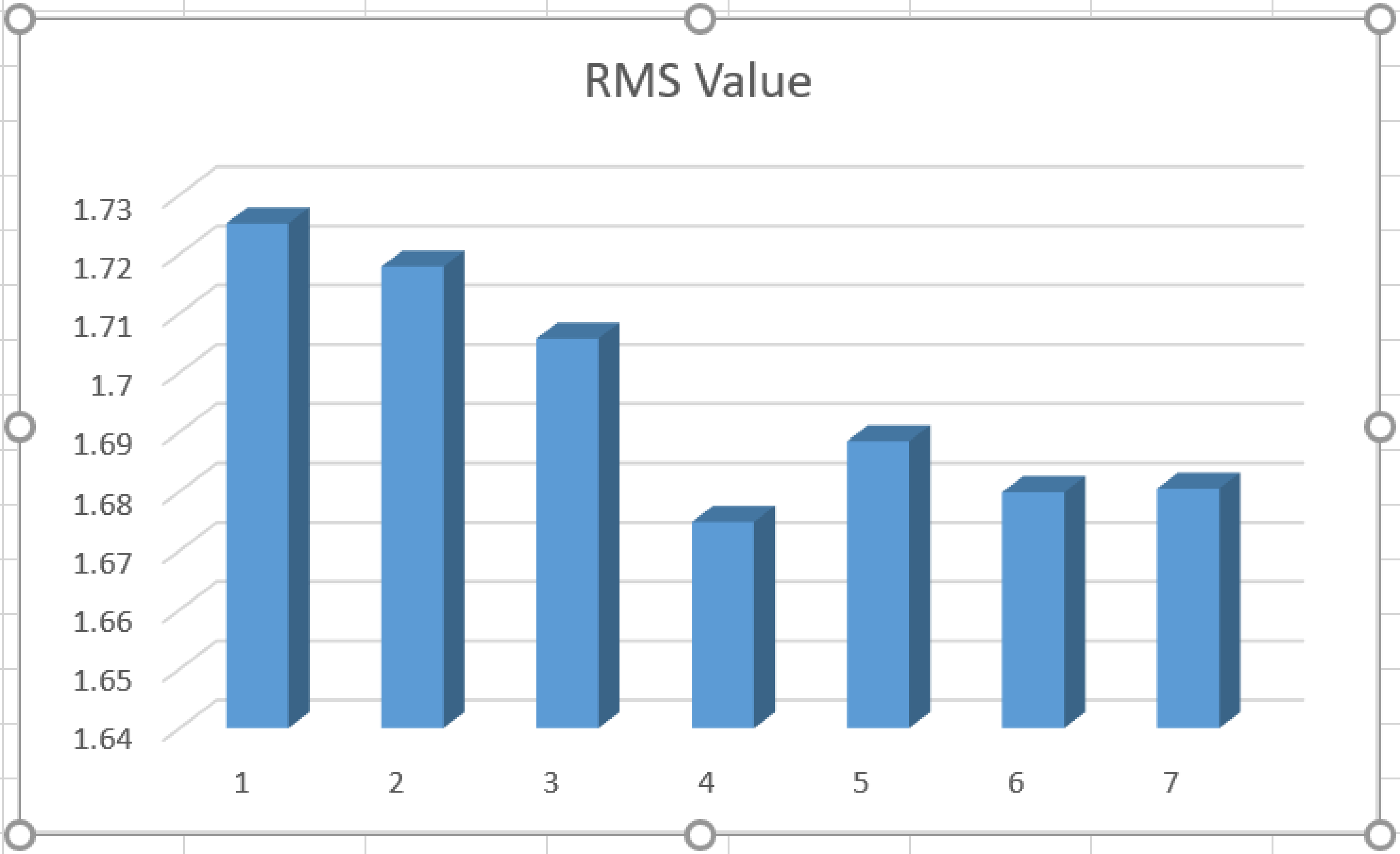
4. Result - 1.4614 RMSE

Results

Predicted Results



Result Continued..



Approach 2

- Auto regressive Integrated Moving Average (ARIMA)
  - Statistical approach
  - works well with less training data.
  - Data need to be stationary

- Test for stationarity

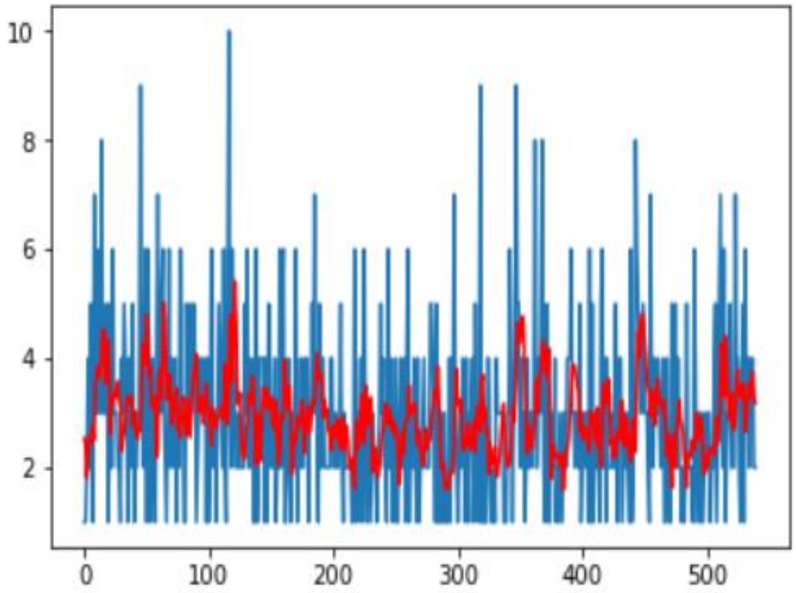
- Dickey Fuller's test (t-stat < critical value)

```
Results of Dickey-Fuller Test:
Test Statistic                   -4.089946
p-value                           0.001006
#Lags Used                       28.000000
Number of Observations Used    2669.000000
Critical Value (1%)              -3.432802
Critical Value (5%)              -2.862624
Critical Value (10%)             -2.567347
dtype: float64
```

Results

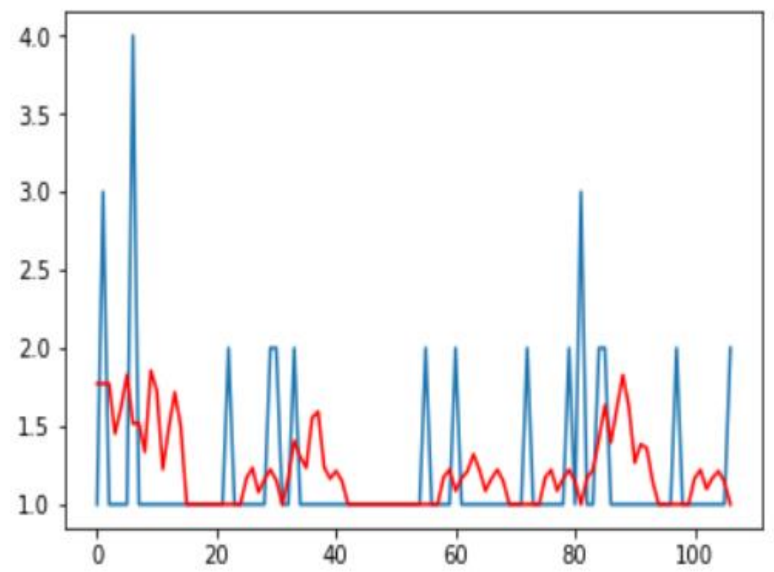
Total Review Prediction

RMSE : 3.075



Negative Review Prediction

RMSE: 0.274



THANK YOU