In [0]:

```
import pandas as pd
import numpy as np
from scipy.sparse import hstack
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from skmultilearn.problem_transform import LabelPowerset
from sklearn.preprocessing import MultiLabelBinarizer
from collections import Counter

from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn import metrics
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix
from sklearn.model_selection import GridSearchCV
from sklearn.pipeline import make_pipeline
from sklearn.utils import class_weight
import neuralcoref
```

In [0]:

```
from sklearn.model_selection import train_test_split
import gensim
from gensim.scripts.glove2word2vec import glove2word2vec
glove2word2vec(glove_input_file="C:\\Users\\SHAILENDRA\\glove.42B.300d.txt", word2vec_output_file="gensim_glove_vectors.txt")
from gensim.models.keyedvectors import KeyedVectors
glove_model = KeyedVectors.load_word2vec_format("gensim_glove_vectors.txt", binary=False)

import spacy
nlp = spacy.load('C:\\Users\\SHAILENDRA\\Anaconda3\\envs\\tf_gpu\\Lib\\site-packages\\en_core_web_md\\en_core_web_md-2.1.0')
```

!pip install neuralcoref

import en_core_web_lg

!pip install en_core_web_lg

coref = neuralcoref.NeuralCoref(nlp.vocab) nlp.add_pipe(coref, name='neuralcoref')

doc=nlp("The place is new and I like the way guests were managed. It was nice and happening place with live music. Staff was courteous but needs training about the alcohol. I had whiskey sour but it was plain in taste as if there was no whiskey in it. Food menu needs to be updated as there are lots of common things which were missing from the menu.")

doc._.has*coref doc.*.coref_clusters

In [0]:

```
a=pd.read_csv('highland.csv')
#b=pd.read_csv('thikana.csv')
c=pd.read_csv('your-ale-house.csv')
d=pd.read_csv('rabits-burrow1.csv')
#e=pd.read_csv('terttulia-bistro.csv')
#f=pd.read_csv('linkin-barrel.csv')
#g=pd.read_csv('azzurro.csv')
```

In [0]:

```
a.shape
```

Out[0]:

```
(394, 4)
```

In [0]:

```
final=pd.concat([c,d,a])
```

In [0]:

```
#final.drop('Unnamed: 0',axis=1,inplace=True)
final.shape
```

Out[0]:

```
(620, 4)
```

In [0]:

```
final.groupby('res_name').agg('count')
```

Out[0]:

res_name	rating	comment	label
highland	394	394	394
mr-rabbits-bar-burrow	113	113	113
your-ale-house	113	113	113

In [0]:

```
import re
a="Ambiance: 3/5 Food: 3/5 Service: 3/5 "
b='Service- (5/5) Taste - 5/5 Ambience-4/5'
c=' Ambience on terrace 4/5 lighting can be better on terrace Starters 5/5 we had chilli chicken, kebabs, veg Manchurian, and paneer pakodas. It was yumm♥️ Dinner 3.5/5'
d='I would rate it 1.5 Alcohol choices 4 Pricing 3 Ambiance 3.5 Staff courtesy 3.5 Over all rating 3 '
e='if we order drinks after 8 pm then we get 10 percent discount. He only comes to our table at 5 mins to 8 to inform the same. '
f='Quality n Quantity- They have Diverse Menu which is lots to choose from as all are tasty,delicious and of good quality of International level!!!. ------------------------- Rating for Pricing-4.8/5* Rating for hospitality-5/5* its a must visit Here.   '
g='great variety of fresh craft beers served by 2nd Pint Brew Co. This brewery serves some of the best craft beers in town. The manager/owner, also helps you choose appetizers which go well with these beers. Their bartender is a master at cocktails. I’ve occasionally had their Sangria, their LIIT and mojito and he’s never gone wrong! Great job Anand! Rating: 5/5. Overall, one of the finest places to visit in Pune on any day of the week. It is difficult to find places where good food is served with good alcohol, and this place delivers both. Good going guys! I’m definitely a regular! Cheers'
li=[a,b,c,d,e,f,g]

def clean_review(i):
    clean=re.sub('\(([a-zA-Z]|\W|[0-9][0-9]+)+\)','',i)
    clean=re.sub(':(\s|)[0-2](\.[0-5]|)\/5'," is bad.",clean)
    clean=re.sub(':(\s|)[3-5](\.[0-5]|)\/5'," is good.",clean)
    clean=re.sub('(\s|:|-)[0-2](\.[0-5]|)\/5'," is bad.",clean)
    clean=re.sub('(\s|:|-)[3-5](\.[0-5]|)\/5'," is good.",clean)
    clean=re.sub('(\s|:|-)[0-3](\.[0])\/5'," is bad.",clean)
    clean=re.sub('(\s|:|-)[3-5](\.[0-9])\/5'," is good.",clean)
    clean=re.sub('(\s|:|-)[0-3](\.[0-4]|\s|)'," is bad.",clean)
    clean=re.sub('(\s|:|-)[3-5](\.[5]|\s|)'," is good.",clean)
    clean=re.sub('(#|@)[a-zA-Z0-9]+(_|-|[a-zA-Z0-9]+)',"",clean) #remove all the tags relating to # or @
    clean=re.sub('\(',' ',clean)

    clean=re.sub('\)',' ',clean)
    clean=re.sub('\.(\.)+','.',clean)
    clean=re.sub('\!(\!)+','!',clean)
    clean=re.sub('oo(o)+','o',clean)
    #clean=re.sub(':',' is',clean)

    clean=re.sub('[0-9](\.|\))+','',clean)
    clean=re.sub('(d|D)(j|J)','dj',clean)
    clean=re.sub('-',' is ',clean)
    clean=re.sub('\*+','',clean)
    clean=re.sub('\,(,)+',',',clean)
```

```
    clean=re.sub('\s(n|N)\s',' and ',clean)
    clean=re.sub('("|")','',clean)
    #clean=re.sub('\.',' . ',clean)#beacuse some sentences create trouble while parsing
    clean=re.sub('[zZ](z|Z)+','z',clean)
    clean=re.sub('[:;)}]+([OPop]|)\s',' ',clean)#removes smileys like :)
    clean=re.sub('[0-9](:|\))',',',clean)# this removes 5) and 5.

    #clean=re.sub('[0-9](nd|rd|th)','',clean) to remove 2nd,3rd
    clean=clean.replace('ambiance','ambience').replace('Ambiance','Ambience').replace('ambeience','Ambience').replace('₹','rupees').replace('Rs.',' rupees ').replace('mins','minutes')\
          .replace("’m",' am').replace("n’t",' not').replace("'s",' is').replace("'m",' am').replace("n't",' not').replace("’s",' is').replace("dnt",' dont').replace("wl",' will').replace
("its",'it is')\
          .replace("Zgold",'zomato gold').replace("Zomato gold",'zomato gold').replace("Zomato Gold",'zomato gold').replace("handson",'handsome').replace("appricate",'appreciate')\
          .replace('okayish','okay').replace('okish','okay').replace('okeish','okay').replace('A/C','ac').replace('AC','ac').replace('a/c','ac').replace('dinnwr','dinner').replace('cityyy
','city').replace('captain','waiter').replace('Captain','waiter')\
          .replace('!','.').replace('chill','enjoy').replace('legen.wait for it.dary','legendary').replace('Piza','pizza').replace('piza','pizza')
    clean=re.sub(r'[^\x00-\x7F]', '', clean)#removes non ascii hearts and stars keep at last

    return(clean.strip())
#re.sub('!(\s)','',c)
```

In [0]:

```
final.comment=final.comment.apply(lambda x:clean_review(x))
```

In [0]:

```
l=[]
amb=['amience','ambienc','ambiences','ambiemce']
alc=['drink','drinks']
service=['serivce','staff']
for i in final.label:
    for j in i.split(','):
        if(j  in alc):
            j='alcohol'
        if(j.strip() in service):
            #print(j)
            j='service'
        if(j.strip()in amb):
            j='ambience'
        if(j.strip()== 'muisc'):
            j='music'

        l.append(j.strip())
```

In [0]:

```
from sklearn.model_selection import train_test_split
y=final.label
#c.drop(['label'],inplace=True,axis=1)
x_train, x_test, y_train, y_test=train_test_split(final,y,test_size=0.3 ,random_state=42)
```

In [0]:

```
x_test.shape
```

Out[0]:

```
(186, 4)
```

In [0]:

```
v=pd.DataFrame(l)
label_dict=pd.value_counts(v.iloc[:,0]).to_dict()
```

In [0]:

```
from sklearn.feature_extraction.text import CountVectorizer
vectorizer = CountVectorizer(vocabulary=label_dict.keys() ,lowercase=False, binary=True)
categories_one_hot_train = vectorizer.fit_transform(x_train.label.values)
categories_one_hot_test = vectorizer.transform(x_test.label.values)
x_tr=pd.DataFrame(categories_one_hot_train.toarray(),columns=vectorizer.get_feature_names())
```

```
x_te=pd.DataFrame(categories_one_hot_test.toarray(),columns=vectorizer.get_feature_names())
x_train.reset_index(inplace=True)
x_test.reset_index(inplace=True)
final_x_tr=pd.concat([x_train,x_tr],axis=1)
final_x_te=pd.concat([x_test,x_te],axis=1)
final_x_tr.drop(['index','label'],axis=1,inplace=True)
final_x_te.drop(['index','label'],axis=1,inplace=True)
final_x_tr.rating=final_x_tr.rating.apply(lambda x:int(x.split(" ")[1].split('.')[0]))
final_x_te.rating=final_x_te.rating.apply(lambda x:int(x.split(" ")[1].split('.')[0]))
```

In [0]:

```
final_x_tr.comment=final_x_tr.comment.apply(lambda x:clean_review(x))
final_x_te.comment=final_x_te.comment.apply(lambda x:clean_review(x))
```

In [0]:

```
import nltk
sno = nltk.stem.SnowballStemmer('english')
stopwords= ['i', 'me', 'my','this', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've",\
            "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', \
            'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their',\
            'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', \
            'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', \
            'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', \
            'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after',\
            'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further',\
            'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more',\
            'most', 'other', 'some', 'such', 'only', 'own', 'same', 'so', 'than', 'too', 'very', \
            's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', \
            've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn',\
            "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn',\
            "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", \
            'won', "won't", 'wouldn', "wouldn't"]
def l_class(review_list):
    processed=[]
    for re in review_list:
        review=[]
        for i in re.split('.'):

            word=[]
            ii=i.split(" ")
            #print(ii)
            for j in ii:

                if(j !='' and j not in stopwords):
                    word.append(sno.stem(j.strip().lower()))
            #print(sent)
            sentence=" ".join(word)
            review.append(sentence)
        processed.append(('.'.join(review)))
    return processed
```

In [0]:

```

```

In [0]:

```
tr_list=l_class(final_x_tr.comment)
te_list=l_class(final_x_te.comment)
```

In [0]:

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorizer = TfidfVectorizer(max_features=300,min_df=3,ngram_range=(1,1))

text_tfidf_train = vectorizer.fit_transform(tr_list)

text_tfidf_test = vectorizer.transform(te_list)
```

In [0]:

```
text_tfidf_train.shape
```

Out[0]:

```
(434, 300)
```

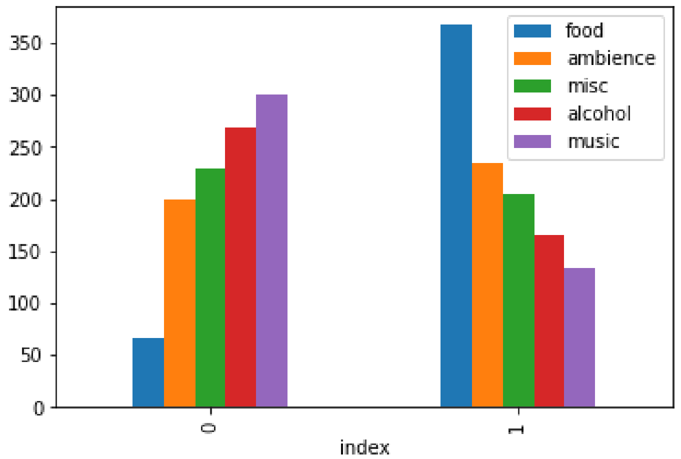
In [0]:

```
import matplotlib.pyplot as plt
import seaborn as sns
d=pd.DataFrame()
for i in ['food','service', 'ambience', 'misc','alcohol', 'music']:
    j=i+'_count'
    print(j)
    d[i]=final_x_tr.groupby(i,as_index=False)['rating'].agg('count').rename(columns={'rating':j})[j]
print(d)
d.reset_index(inplace=True)
#plt.rcParams["figure.figsize"] = (20,20)
d.plot(x='index',y=['food', 'ambience', 'misc','alcohol', 'music'],kind='bar')
#ax = d.plot(x='index', y="A", kind="bar")
#df.plot(x="index", y="B", kind="bar", ax=ax, color="C2")
#df.plot(x="X", y="C", kind="bar", ax=ax, color="C3")
```

```
food_count
service_count
ambience_count
misc_count
alcohol_count
music_count
   food  service  ambience  misc  alcohol  music
0    67      187       199   230      268    301
1   367      247       235   204      166    133
```

Out[0]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x15de2d5e438>
```



In [0]:

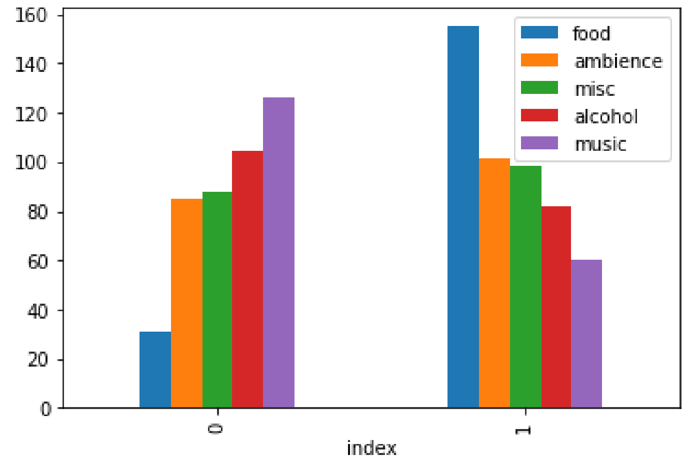
```
import matplotlib.pyplot as plt
import seaborn as sns
d=pd.DataFrame()
for i in ['food','service', 'ambience', 'misc','alcohol', 'music']:
    j=i+'_count'
    print(j)
    d[i]=final_x_te.groupby(i,as_index=False)['rating'].agg('count').rename(columns={'rating':j})[j]
print(d)
d.reset_index(inplace=True)
#plt.rcParams["figure.figsize"] = (20,20)
d.plot(x='index',y=['food', 'ambience', 'misc','alcohol', 'music'],kind='bar')
#ax = d.plot(x='index', y="A", kind="bar")
#df.plot(x="index", y="B", kind="bar", ax=ax, color="C2")
#df.plot(x="X", y="C", kind="bar", ax=ax, color="C3")
```

```
food_count
```

```
food_count
service_count
ambience_count
misc_count
alcohol_count
music_count
   food  service  ambience  misc  alcohol  music
0    31       60        85    88      104    126
1   155      126       101    98       82     60
```

Out[0]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x15eb6cfc208>
```



In [0]:

```python
def meta_features(df):
    df['len_of_comment']=df.iloc[:,2].apply(lambda x:len(x))
    df['no_of_word_comment']=df.iloc[:,2].apply(lambda x:len(x.split(" ")))
    df['no_of_sent_comment']=df.iloc[:,2].apply(lambda x:len(x.split(".")))

meta_features(final_x_tr)
meta_features(final_x_te)
```

In [0]:

```python
def cal(x):
    count=0
    for i in range(0,6):
        if(x[i]==1):
            count+=1
    return count
final_x_tr['no_of_labels']=final_x_tr.iloc[:,[3,4,5,6,7,8]].apply(lambda x:cal(x),axis=1)
final_x_te['no_of_labels']=final_x_te.iloc[:,[3,4,5,6,7,8]].apply(lambda x:cal(x),axis=1)
#final_x_tr.iloc[0:2,[3,4,5,6,7,8]].apply(lambda x:cal(x),axis=1)
```

In [0]:

```python
#scaling all the numerical attributes helped in increasing the accuracy by 2 %
from sklearn.preprocessing import MinMaxScaler
s=MinMaxScaler()
final_x_tr.iloc[:,1]= s.fit_transform(final_x_tr.iloc[:,1].values.reshape(-1,1))
final_x_te.iloc[:,1]= s.transform(final_x_te.iloc[:,1].values.reshape(-1,1))

s=MinMaxScaler()
final_x_tr.iloc[:,9]= s.fit_transform(final_x_tr.iloc[:,9].values.reshape(-1,1))
final_x_te.iloc[:,9]= s.transform(final_x_te.iloc[:,9].values.reshape(-1,1))


s=MinMaxScaler()
final_x_tr.iloc[:,10]= s.fit_transform(final_x_tr.iloc[:,10].values.reshape(-1,1))
final_x_te.iloc[:,10]= s.transform(final_x_te.iloc[:,10].values.reshape(-1,1))


s=MinMaxScaler()
final_x_tr.iloc[:,11]= s.fit_transform(final_x_tr.iloc[:,11].values.reshape(-1,1))
```

```
final_x_te.iloc[:,11]= s.transform(final_x_te.iloc[:,11].values.reshape(-1,1))


s=MinMaxScaler()
final_x_tr.iloc[:,12]= s.fit_transform(final_x_tr.iloc[:,12].values.reshape(-1,1))
final_x_te.iloc[:,12]= s.transform(final_x_te.iloc[:,12].values.reshape(-1,1))
```

tried but tfidf gave better accuracy

from sklearn.feature_extraction.text import CountVectorizer vectorizer = CountVectorizer(max_features=300,min_df=3,ngram_range=(1,1))

text_bow_train = vectorizer.fit_transform(tr_list)

text_bow_test = vectorizer.transform(te_list)

In [0]:

```
from scipy.sparse import hstack
x_tf_idf_tr_stacked=hstack([final_x_tr.iloc[:,[1,9,10,11,12]],text_tfidf_train])
x_tf_idf_te_stacked=hstack([final_x_te.iloc[:,[1,9,10,11,12]],text_tfidf_test])
```

In [0]:

```
class_names=['misc','food','service','ambience','alcohol','music']
```

losses = [] auc = []

class_weights = list(class_weight.compute_class_weight('balanced', y_train.unique(), y_train))

for class_name in class_names:

```
#call the labels one column at a time so we can run the classifier on them
train_target = final_x_tr[class_name]
test_target = final_x_te[class_name]
classifier = SVC(C=1, kernel='linear',probability=True)

cv_loss = np.mean(cross_val_score(classifier, text_tfidf_train, train_target, cv=5, scoring='neg_log_loss'))
losses.append(cv_loss)
print('CV Log_loss score for class {} is {}'.format(class_name, cv_loss))

cv_score = np.mean(cross_val_score(classifier, text_tfidf_train, train_target, cv=5, scoring='accuracy'))
print('CV Accuracy score for class {} is {}'.format(class_name, cv_score))

classifier.fit(text_tfidf_train, train_target)
y_pred = classifier.predict(text_tfidf_test)
y_pred_prob = classifier.predict_proba(text_tfidf_test)[:, 1]
auc_score = metrics.roc_auc_score(test_target, y_pred_prob)
auc.append(auc_score)
print("CV ROC_AUC score {}\n".format(auc_score))

print(confusion_matrix(test_target, y_pred))
print(classification_report(test_target, y_pred))
```

print('Total average CV Log_loss score is {}'.format(np.mean(losses))) print('Total average CV ROC_AUC score is {}'.format(np.mean(auc)))

In [0]:

```
pd.value_counts(x_te.service)
```

Out[0]:

```
1    126
0     60
Name: service, dtype: int64
```

In [0]:

```
pd.value_counts(x_te.alcohol)
```

Out[0]:

```
0    104
1     82
Name: alcohol, dtype: int64
```

In [0]:

```
pd.value_counts(x_te.ambience)
```

Out[0]:

```
1    101
0     85
Name: ambience, dtype: int64
```

In [0]:

```
pd.value_counts(x_te.music)
```

Out[0]:

```
0    126
1     60
Name: music, dtype: int64
```

In [0]:

```
pd.value_counts(x_te.food)
```

Out[0]:

```
1    155
0     31
Name: food, dtype: int64
```

In [0]:

```
from sklearn.utils import class_weight
d={}
for i in class_names:
    print(i)
    class_weights = list(class_weight.compute_class_weight('balanced',
                                                 x_tr[i].unique(),
                                                 x_tr[i]))
    d[i]=class_weights
```

```
misc
food
service
ambience
alcohol
music
```

In [0]:

```
d
```

Out[0]:

```
{'misc': [0.9434782608695652, 1.0637254901960784],
 'food': [0.5912806539509536, 3.2388059701492535],
 'service': [1.160427807486631, 0.8785425101214575],
 'ambience': [1.0904522613065326, 0.9234042553191489],
 'alcohol': [1.3072289156626506, 0.8097014925373134],
 'music': [0.7209302325581395, 1.631578947368421]}
```

In [0]:

```
li=[]
l=[]
amb=['amience','ambienc','ambiences','ambiemce']
alc=['drink','drinks']
service=['serivce','staff']
for i in x_train.label:
    temp=[]
    for j in i.split(','):
        if(j  in alc):
            j='alcohol'
        if(j.strip() in service):
            #print(j)
            j='service'
        if(j.strip()in amb):
            j='ambience'
        if(j.strip()== 'muisc'):
            j='music'

        temp.append(j.strip())
    li.append(set(temp))
x_train['labell']=li

li=[]
l=[]
amb=['amience','ambienc','ambiences','ambiemce']
alc=['drink','drinks']
service=['serivce','staff']
for i in x_test.label:
    temp=[]
    for j in i.split(','):
        if(j  in alc):
            j='alcohol'
        if(j.strip() in service):
            #print(j)
            j='service'
        if(j.strip()in amb):
            j='ambience'
        if(j.strip()== 'muisc'):
            j='music'

        temp.append(j.strip())
    li.append(set(temp))
x_test['labell']=li
```

```
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\ipykernel_launcher.py:21: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\ipykernel_launcher.py:43: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

In [0]:

```
from sklearn.preprocessing import MultiLabelBinarizer

mlb = MultiLabelBinarizer()
y_tr = mlb.fit_transform(x_train.labell)
y_te = mlb.transform(x_test.labell)
```

In [0]:

```
from skmultilearn.problem_transform import LabelPowerset
from sklearn.preprocessing import MultiLabelBinarizer

from sklearn.svm import LinearSVC
from sklearn.linear_model import LogisticRegression
```

```
from sklearn.pipeline import Pipeline
text_clf = Pipeline([('clf', LabelPowerset(LogisticRegression(C=10)))])
text_clf = text_clf.fit(x_tf_idf_tr_stacked, y_tr)
predicted = text_clf.predict(x_tf_idf_te_stacked)

# Calculate accuracy
np.mean(predicted.todense() == y_te)
```

```
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\sklearn\linear_model\logistic.py:432: FutureWarning: Default solver will be changed to 'lbfgs' in 0.22. Specify a solver to silen
ce this warning.
  FutureWarning)
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\sklearn\linear_model\logistic.py:469: FutureWarning: Default multi_class will be changed to 'auto' in 0.22. Specify the multi_cla
ss option to silence this warning.
  "this warning.", FutureWarning)
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\scipy\sparse\lil.py:504: FutureWarning: future versions will not create a writeable array from broadcast_array. Set the writable
flag explicitly to avoid this warning.
  if not i.flags.writeable or i.dtype not in (np.int32, np.int64):
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\scipy\sparse\lil.py:506: FutureWarning: future versions will not create a writeable array from broadcast_array. Set the writable
flag explicitly to avoid this warning.
  if not j.flags.writeable or j.dtype not in (np.int32, np.int64):
c:\users\shailendra\anaconda3\envs\tf_gpu\lib\site-packages\scipy\sparse\lil.py:510: FutureWarning: future versions will not create a writeable array from broadcast_array. Set the writable
flag explicitly to avoid this warning.
  if not x.flags.writeable:
```

Out[0]:

```
0.7231182795698925
```

In [0]:

```
df1=pd.DataFrame(predicted.todense()==y_te,columns=mlb.classes_)
```

In [0]:

```
df1.shape
```

Out[0]:

```
(186, 6)
```

In [0]:

```
#df2=pd.DataFrame(predicted.todense()==np.ones([124,6]),columns=mlb.classes_)

df2=pd.DataFrame(predicted.todense()==np.ones([186,6]),columns=mlb.classes_) # when the data is divided into 0.3 test above for 0.2
```

In [0]:

```
#pd.merge(df1, df2, how='outer',indicator=True)
df3=pd.DataFrame()
df3['alcohol']=df1.alcohol & df2.alcohol
df3['ambience']=df1.ambience & df2.ambience
df3['food']=df1.food & df2.food
df3['misc']=df1.misc & df2.misc
df3['music']=df1.music & df2.music
df3['service']=df1.service & df2.service
#df3
```

In [0]:

```
def label(x):
    l=[]
    labels=mlb.classes_
    for i in range(0,len(x)):
        #print(i)
        if(x[i]==True):
            l.append(labels[i])
    return(l)
final_x_te['predicted_label']=df3.apply(lambda x:label(x),axis=1)
```

In [0]:

```
after_pred_te=pd.DataFrame({'res_name':final_x_te.res_name,'comment':final_x_te.comment, 'predicted':mlb.inverse_transform(predicted)})
```

In [0]:

```
test_reviews=after_pred_te.iloc[1:5,1]
```

In [0]:

```

```

In [0]:

```
from collections import Counter
def aspect_extraction(j):
    #print('**')
    sent=nlp(j)
    sent_dict={}
    noun_count=0
    aspect=['alcohol','novice', 'ambience', 'Food','food', 'misc', 'music', 'service','drinks','cocktails','moctails']
    pos=['yum','awesome','ample','amazing','great','good','epic','pleasant''helpful','nice','decent','better']
    neg=['bearable','disgusting','limited','bad','cheap']
    for token in sent:
        key=''
        value=''

        if(token.dep_=='nsubj' ):# modified as per 1 where if  acomp and advmod is in children of acomp then take advmod
            #print(token.text)
            root=token
            key=token.text
            for k in root.children:
                if(k.dep_=='amod'):
                   # print(k.text)
                    #print('--!!!!--')
                    value=k.text
                    break;

            root=token.head

            if(token.pos_=='PROPN'):
                key='waiter'
                #print(key,token)
            else:
                key=token.text
           # if(token.head.dep_=='conj') commented due to 45 if un commenting check with 45 as well
           #   value=token.head.text

                #print(token,token.dep_,token.head.text)
           # if(token.head.pos_=='VERB'and token.head.dep_=='ROOT'): #added after 41 makes a lot of noise for rest
           #     value=token.head.text
           #     print(key,value)

            for k in root.children:
                   # print(k,k.dep_,k.head)

                if(k.dep_=='nsubj'): #added for 60-61
                    key=k.text
                    #print(key)
                if(k.dep_=='acomp' ):
                    root=k
                    value=k.text
                    #print(value)
                    for l in root.children:
                        if(l.dep_=='advmod'  ):
                            #print(key,l.text)
                            value= l.text+'.'+value
                            break;
                       # elif(l.pos_=='ADJ'): #example staff was friendly but slow
                            #print(key,l.text)
                            value= l.text
                            break;
```

```
            elif(k.dep_=='conj'):
                value=k.text
                #print(value)

            elif(k.dep_=='attr'):  #added for 60-61
                value=k.text
                #print(value)

            elif(k.pos_=='ADV' and k.dep_=='neg'):# try repacing with numbers if good add +1 and bad then -.6
                for i in k.head.children:
                    if(i.dep_=='acomp'or i.dep_=='amod'):
                        value=k.text+'.'+i.text


            if(key!='' and value !='' and value!=0 and key not in sent_dict.keys()):
                sent_dict[key]=value
                #print(sent_dict)
               # if(k.dep_=='amod'):



        #root=token.head#from here till line 72  also 56 to accomodate 93 example

        #if(token.head.dep_=='prep'):
        #    print(root.head.text,root.head.dep_)
        #    if(root.head.dep_=='amod'):
        #        value=token.head.head.text
        #        print(value)

    elif((token.pos_=='NOUN' or token.pos_=='PROPN') and (key=='') and (value=='') ):
        root=token#searchin in its children
        #print(root)
        if(token.head.text in sent_dict.keys()):

            key=token.text
            value=sent_dict[token.head.text]
        elif(token.head.dep_=='amod'):#or token.head.pos_=='VERB'
            value=token.head.text
            key=token.text
            #print(key,value)
        elif(token.head.pos_=='VERB'):#or token.head.pos_=='VERB'
            flag=0
            for i in token.head.children:
                if(i.pos_=='ADV'):
                    flag=1
            if(flag==1):
                value=token.head.text
                key=token.text
        elif((key=='') and (value=='') ):
            key=token.text
           # print(key)
            #if(found==0):
            amod_present=1
            prep_present=0
            for i in root.children:
                if(i.dep_=='amod' or i.pos_=='ADJ'):# check for negation 95 write code for it
                    value=i.text
                    #found=1
                    amod_present=1
                    root=i
                    #print(key,value)


                    for j in root.children:
                        #print(j)
                        if(j.dep_=='npadvmod'): #example 109
                            value=j.text+'.'+root.text
                            #print(key,value)
                            break;
```

```
                # if(i.dep_=='prep'):
                #     prep_present=1


                elif(i.dep_=='advmod' and value!=''):#yu can add sentiment and play with else
                    #print(value)
                    value=value+'.'+i.text
                    found=1
                #print(key,value)

        if((value=='')):
            root=token.head
            key=token.text
            value=''
            for i in root.children:
                if(i.pos_=='ADJ' ):
                    #print(key,i.text)
                    if( value=='' or value==0 ):
                        value=i.text


                        #print(key,value)
                if(i.dep_=='neg' and value!=0 and value!=''):
                    value=i.text+'.'+value
                    #print(key,value)




        if(key!='' and value !='' and value!=0 and key not in sent_dict.keys()):
            sent_dict[key]=value


    elif(token.dep_=='dobj'):   #for example see first sentence
        #print('???')




        root=token
        value=token.text
       # print(root)
        #print(token,token.dep_,token.head.text)
        for k in root.children:
            if(k.dep_=='amod'):
                #print(k,k.dep_,k.head)
                #value=token.head
                value=k.text
                key=k.head.text
                #print(k.text)
            if(k.dep_=='prep'):
                root =k

                for i in root.children:
                    if(i.dep_=='pobj'):
                        key=i.text



               # print(key)
            if(k.dep_=='nsubj'):
                #print(k,k.dep_,k.head)
                #value=token.head
                key=k.text
                value=root.text
                #print(k.text)

                #print('---*')
        if(key!='' and value !='' and value!=0):
                    sent_dict[key]=value
```

```
                    #print(sent_dict)



        elif token.text.lower() in aspect+pos+neg and key=='' and value=='':
            #       sentiment = 1 if token.text in pos else -1
            # if target is an adverb modifier (i.e. pretty, highly, etc.)
            # but happens to be an opinion word, ignore and pass

                #print(token)
                if (token.dep_ == "amod"):
                    #print(token)
                    sent_dict[token.head.text] =token.text
                    #print(token.head.text)
                    #here=0
                if (token.dep_ == "conj") and token.text not in sent_dict.keys() and token.head.text  in sent_dict.keys() :
                        #print(token not in sent_dict.keys())
                        #print(token.text)

                        sent_dict[token.text] = sent_dict[token.head.text]
                    #print(sent_dict[token.head.text])
                if(key!='' and value !='' and value!=0):
                    sent_dict[key]=value




    return(sent_dict)
    print('--------------')
```

In [0]:

```
def list_of_dict_to_dict(d):
    dd={}
    for i in d:
        for j in i.keys():
            dd[j]=i[j]
    return dd
def sent_calculation(x):
    for i in x.keys():
        print(i,x[i])
def similarity(aspect,li):
    temp=0
    c=0
    for i in li:
        try:
            temp+=glove_model.n_similarity([aspect],[i])
            c+=1
        except :
            #print('%s is not in the vocab'%(aspect))
            None



    if(c !=0):
        return(temp/c)
    else:
        return 0
```

In [0]:

```
review="food and service are nice but music is bad."
count=0
final_list=[]
for i in after_pred_te.iloc[0:,1]:
    review=i
    d=[]
    print(count)
    ambience=['ambience','floor','decor','interior','view']
    music=['musician','dance','beats','dj','dn']
```

```
    staff=['staff','server','good','servers','service','hosts','team','owner','owners','management','waiters','raju','bartender','folks']
    not_staff=['we','it','one','they','were','restaurent','which','time','that','i']
    alc=['cocktails','mocktails','drinks','beer','towers']
    food=['food','quantity','kebab','dinner','dessert','gravy','tiramisu','starters','taste','chakna','biryani','flavours','pizza','cuisine','tandoori','wings','kebabs']
    pos=['yum','pretty','luscious','classic','stunning','big','star','kind','strong','pro','amiable','unique','sexy','interested','yummy','awesome','fast','unique','cheap','loved','ample',
'licking','smacking','high','soft','soulful','fab','amazing','something','tempting','love','smimilar','cozy','mannered','pleasing','great','observant','prompt','comfortable','worth','livel
y','good','best','epic','pleasant','clean','courteous','delicious','delecious','legendary','9/10','helpful','beautiful','nice','decent','tasty','quick','cozy','liked','perfect','lovely','
polite','friendly']
    neg=['bearable','wrong','bland','arrogant','low','priced','oily','normal','disgusting','few','limited','bad','slow','better','overpriced','disappointment','slow','costly','ok','okay','
local','okie','loud','hostile','pathetic','average','dull']

    advb=['really','so','too','equally','just','very','amazingly','made','much','extremely','eaqually','quite','pocket','enough']
    neg_advb=['not',]
    for x in review.split('.'):

        y=aspect_extraction(x)
        d.append(y)

    dd=list_of_dict_to_dict(d)# this is our dictionary for a single review
    amb_count=0
    alc_count=0
    food_count=0
    music_count=0
    staff_count=0

    for i in dd.keys():
        #print(i)
        if(i.lower() in food ):
            temp=dd[i].split('.')# the value like awesome.made
            #print(temp)
            if(len(temp)>1):
                if(temp[0] in advb):
                    if (temp[1] in pos):
                        food_count+=1.5
                    else:
                        food_count-=1.5

                if(temp[1] in advb):
                    if (temp[0] in pos):
                        food_count+=1.5
                    else:
                        food_count-=1.5
                elif(temp[0] in neg_advb):
                    if (temp[1] in neg):
                        food_count+=1.5
                    else:
                        food_count-=1.5
                        #print(i,food_count)




            else:
                #print(i,dd[i].lower())
                if(dd[i].lower() in pos):
                    food_count+=1
                    #print('**')
                if(dd[i].lower() in neg):
                    food_count-=1
                    #print(dd[i],i,food_count)
            #print(i,food_count)
        elif(i.lower() in alc):
            temp=dd[i].split('.')# the value like awesome.made
            if(len(temp)>1):
                if(temp[0] in advb):
                    if (temp[1] in pos):
                        alc_count+=1.5
                    else:
                        alc_count-=1.5

                if(temp[1] in advb):
                    if (temp[0] in pos):
```

```
                    alc_count+=1.5
                else:
                    alc_count-=1.5
            elif(temp[0] in neg_advb):
                if (temp[1] in neg):
                    alc_count+=1.5
                else:
                    alc_count-=1.5

        else:
            #print(dd[i])
            if(dd[i].lower() in pos):
                alc_count+=1

            if(dd[i].lower() in neg):
                alc_count-=1
        #print(i,alc_count)
    elif(i.lower() in staff):
        #print("here is %s"%i)
        temp=dd[i].split('.')# the value like awesome.made
        if(len(temp)>1):
            if(temp[0] in advb):
                if (temp[1] in pos):
                    staff_count+=1.5
                else:
                    staff_count-=1.5

            if(temp[1] in advb):
                if (temp[0] in pos):
                    staff_count+=1.5
                else:
                    staff_count-=1.5
            elif(temp[0] in neg_advb):
                if (temp[1] in neg):
                    staff_count+=1.5
                else:
                    staff_count-=1.5
            #print(i,dd[i],staff_count)

        else:

            if(dd[i].lower() in pos):
                staff_count+=1

            if(dd[i].lower() in neg):
                staff_count-=1
            #print(i,dd[i],staff_count)
    elif(i.lower() in music):
        #print("here is %s"%i)
        temp=dd[i].split('.')# the value like awesome.made
        if(len(temp)>1):
            if(temp[0] in advb):
                if (temp[1] in pos):
                    music_count+=1.5
                else:
                    music_count-=1.5

            if(temp[1] in advb):
                if (temp[0] in pos):
                    music_count+=1.5
                else:
                    music_count-=1.5
            elif(temp[0] in neg_advb):
                if (temp[1] in neg):
                    music_count+=1.5
                else:
                    music_count-=1.5

        else:

            if(dd[i].lower() in pos):
                music_count+=1
```

```
                if(dd[i].lower() in neg):
                    music_count-=1

        elif(i.lower() in ambience):
            #print("here is %s"%i)
            temp=dd[i].split('.')# the value like awesome.made
            if(len(temp)>1):
                if(temp[0] in advb):
                    if (temp[1] in pos):
                        amb_count+=1.5
                    else:
                        amb_count-=1.5

                if(temp[1] in advb):
                    if (temp[0] in pos):
                        amb_count+=1.5
                    else:
                        amb_count-=1.5
                elif(temp[0] in neg_advb):
                    if (temp[1] in neg):
                        amb_count+=1.5
                    else:
                        amb_count-=1.5

            else:

                if(dd[i].lower() in pos):
                    amb_count+=1

                if(dd[i].lower() in neg):
                    amb_count-=1



        else:
            similarity_score=similarity(i.lower(),food)
            similarity_score=round(similarity_score,2)
            similarity_score_alc=similarity(i.lower(),alc)
            similarity_score_alc=round(similarity_score_alc,2)
            similarity_score_staff=similarity(i.lower(),staff)
            similarity_score_staff=round(similarity_score_staff,2)
            similarity_score_music=similarity(i.lower(),music)
            similarity_score_music=round(similarity_score_music,2)
            similarity_score_amb=similarity(i.lower(),ambience)
            similarity_score_amb=round(similarity_score_amb,2)
            print(i.lower(),similarity_score,similarity_score_alc,similarity_score_staff,similarity_score_music,similarity_score_amb)
            staff_flag=0
            music_flag=0
            food_flag=0
            alc_flag=0
            amb_flag=0

            flag_list=[similarity_score,similarity_score_alc,similarity_score_staff,similarity_score_music]
            maxi=max(flag_list)

            if(flag_list[0]==maxi):
                food_flag=1
            elif(flag_list[1]==maxi):
                alc_flag=1
            elif(flag_list[2]==maxi):
                staff_flag=1
            elif(flag_list[3]==maxi):
                music_flag=1
            elif(flag_list[4]==maxi):
                amb_flag=1

            #print("the flag seleceted for %s is %s and maxi is %s"%(i,music_flag,maxi))



            if(food_flag==1):
```

```
            temp=dd[i].split('.')# the value like awesome.made

            if(len(temp)>1 and  similarity_score>0.31):

                if(temp[0] in advb):
                    if (temp[1] in pos):
                        food_count+=1.5

                    else:
                        food_count-=1.5


                if(temp[1] in advb):
                    if (temp[0] in pos):
                        food_count+=1.5
                    else:
                        food_count-=1.5
                elif(temp[0] in neg_advb):
                    if (temp[1] in neg):
                        food_count+=1.5
                    else:
                        food_count-=1.5
                #print(i,dd[i],food_count)

            elif(similarity_score>0.31 and dd[i].lower() in pos+neg):
                #print(i)

                if(dd[i].lower() in pos):
                    food_count+=1

                elif(dd[i].lower() in neg):
                    food_count-=1
                #print(i,dd[i],food_count)


        if(alc_flag==1):


            temp=dd[i].split('.')# the value like awesome.made
            if(len(temp)>1 and  similarity_score_alc>0.35):


                if(temp[0] in advb):
                    if (temp[1] in pos):
                        alc_count+=1.5
                    else:
                        alc_count-=1.5

                if(temp[1] in advb):
                    if (temp[0] in pos):
                        alc_count+=1.5
                    else:
                        alc_count-=1.5
                elif(temp[0] in neg_advb):
                    if (temp[1] in neg):
                        alc_count+=1.5
                    else:
                        alc_count-=1.5

            elif(similarity_score_alc>0.35 and dd[i].lower() in pos+neg):
                #print(dd[i],i.lower(),similarity_score)
                #print("%s is here"%i)
                if(dd[i].lower() in pos):
                    alc_count+=1
                if(dd[i].lower() in neg):
                    alc_count-=1
            #print(i,dd[i],alc_count)

        elif(staff_flag==1):
            temp=dd[i].split('.')# the value like awesome.made
```

```
            #print(temp)
            if(len(temp)>1 and  similarity_score_staff>0.40 and i.lower() not in not_staff):


                if(temp[0] in advb):
                    if (temp[1] in pos):
                        staff_count+=1.5
                    else:
                        staff_count-=1.5

                if(temp[1] in advb):
                    if (temp[0] in pos):
                        staff_count+=1.5
                    else:
                        staff_count-=1.5
                elif(temp[0] in neg_advb):
                    if (temp[1] in neg):
                        staff_count+=1.5
                    else:
                        staff_count-=1.5


            elif(similarity_score_staff>0.40 and dd[i].lower() in pos+neg and i.lower() not in not_staff):
                #print(dd[i],i.lower(),similarity_score_staff,staff_count)

                if(dd[i].lower() in pos):
                    staff_count+=1
                if(dd[i].lower() in neg):
                    staff_count-=1

            #print(i,dd[i],staff_count)
        elif(music_flag==1):
            temp=dd[i].split('.')# the value like awesome.made
            #print(len(temp))
            #print(temp)
            if(len(temp)>1 and  similarity_score_music>0.45):


                if(temp[0] in advb):
                    #print(temp[0])
                    if (temp[1] in pos):
                        music_count+=1.5
                    else:
                        music_count-=1.5

                if(temp[1] in advb):
                    if (temp[0] in pos):
                        music_count+=1.5
                    else:
                        music_count-=1.5
                elif(temp[0] in neg_advb):
                    if (temp[1] in neg):
                        music_count+=1.5
                    else:
                        music_count-=1.5


            elif(similarity_score_music>0.45 and dd[i].lower() in pos+neg):

                #print(dd[i],i.lower(),similarity_score)

                if(dd[i].lower() in pos):
                    music_count+=1
                if(dd[i].lower() in neg):
                    music_count-=1

            #print(i,dd[i],music_count)
        elif(amb_flag==1):
            temp=dd[i].split('.')# the value like awesome.made
            #print(len(temp))
            #print(temp)
            if(len(temp)>1 and  similarity_score_music>0.45):
```

```python
                    if(temp[0] in advb):
                        #print(temp[0])
                        if (temp[1] in pos):
                            amb_count+=1.5
                        else:
                            amb_count-=1.5

                    if(temp[1] in advb):
                        if (temp[0] in pos):
                            amb_count+=1.5
                        else:
                            amb_count-=1.5
                    elif(temp[0] in neg_advb):
                        if (temp[1] in neg):
                            amb_count+=1.5
                        else:
                            amb_count-=1.5


                elif(similarity_score_amb>0.45 and dd[i].lower() in pos+neg):

                    #print(dd[i],i.lower(),similarity_score)

                    if(dd[i].lower() in pos):
                        amb_count+=1
                    if(dd[i].lower() in neg):
                        amb_count-=1

                #print(i,dd[i],amb_count)








    #print(dd)
    #print("the food count is :%f"%(food_count))
    #print("the alcohol count is :%f"%(alc_count))
    #print("the staff count is :%f"%(staff_count))
    #print("the music count is :%f"%(music_count))
    #print("the ambience count is :%f"%(amb_count))

    count+=1
    record=(food_count,alc_count,staff_count,music_count,amb_count)
    final_list.append(record)
    #sent_calculation(y)
```

```
0
rahagdale 0 0 0 0 0
waiter 0.27 0.26 0.3 0.13 0.19
it 0.28 0.26 0.43 0.34 0.39
time 0.25 0.24 0.41 0.31 0.36
1
veg 0.34 0.21 0.13 0.06 0.15
lit 0.15 0.23 0.2 0.18 0.33
minutes 0.24 0.22 0.31 0.24 0.28
2
corn 0.28 0.21 0.17 0.15 0.19
restaurant 0.39 0.36 0.37 0.22 0.39
disappointment 0.15 0.09 0.22 0.13 0.15
place 0.3 0.27 0.4 0.3 0.43
3
time 0.25 0.24 0.41 0.31 0.36
manager 0.13 0.15 0.43 0.23 0.25
host 0.19 0.23 0.42 0.3 0.25
it 0.28 0.26 0.43 0.34 0.39
4
lighting 0.14 0.19 0.21 0.21 0.45
```

it 0.28 0.26 0.43 0.34 0.39
desserts 0.43 0.4 0.18 0.14 0.22
brownie 0.25 0.16 0.11 0.1 0.09
place 0.3 0.27 0.4 0.3 0.43
5
street 0.17 0.23 0.27 0.28 0.35
seating 0.17 0.23 0.23 0.12 0.42
area 0.2 0.23 0.36 0.26 0.44
crowd 0.19 0.26 0.32 0.3 0.29
menu 0.36 0.3 0.3 0.2 0.33
host 0.19 0.23 0.42 0.3 0.25
visit 0.22 0.22 0.35 0.25 0.35
6
place 0.3 0.27 0.4 0.3 0.43
roof 0.15 0.2 0.24 0.17 0.4
thing 0.26 0.24 0.41 0.35 0.35
vibe 0.16 0.18 0.19 0.32 0.33
hour 0.24 0.27 0.34 0.28 0.29
rates 0.13 0.15 0.26 0.16 0.24
day 0.27 0.28 0.38 0.31 0.36
7
this 0.24 0.21 0.38 0.3 0.38
air 0.19 0.22 0.28 0.25 0.36
music 0.2 0.25 0.31 0.5 0.34
options 0.25 0.23 0.34 0.19 0.38
8
place 0.3 0.27 0.4 0.3 0.43
9
highland 0.14 0.17 0.21 0.2 0.25
restaurant 0.39 0.36 0.37 0.22 0.39
offer 0.26 0.26 0.39 0.26 0.4
10
this 0.24 0.21 0.38 0.3 0.38
pocket 0.14 0.13 0.23 0.18 0.27
there 0.28 0.3 0.45 0.34 0.4
place 0.3 0.27 0.4 0.3 0.43
budget 0.18 0.17 0.31 0.19 0.3
you 0.26 0.26 0.43 0.35 0.38
valet 0.09 0.14 0.21 0.09 0.21
one 0.27 0.28 0.44 0.36 0.39
mosquitoes 0.07 0.1 0.09 0.03 0.08
chicken 0.46 0.27 0.22 0.18 0.22
roti 0.28 0.1 0.03 0.05 0.03
which 0.27 0.28 0.42 0.32 0.4
rice 0.37 0.22 0.23 0.2 0.21
it 0.28 0.26 0.43 0.34 0.39
crowd 0.19 0.26 0.32 0.3 0.29
issues 0.15 0.16 0.36 0.2 0.27
go 0.27 0.27 0.4 0.33 0.34
11
alcohol 0.2 0.34 0.23 0.17 0.19
offers 0.23 0.26 0.38 0.26 0.43
events 0.18 0.27 0.33 0.3 0.29
12
this 0.24 0.21 0.38 0.3 0.38
13
rabbit 0.2 0.14 0.18 0.17 0.21
options 0.25 0.23 0.34 0.19 0.38
favourite 0.3 0.27 0.27 0.32 0.28
sides 0.26 0.2 0.27 0.19 0.35
kiwi 0.22 0.19 0.19 0.15 0.16
mocktail 0.11 0.24 -0.03 -0.03 -0.06
they 0.27 0.28 0.45 0.31 0.36
14
place 0.3 0.27 0.4 0.3 0.43
all 0.26 0.28 0.43 0.33 0.41
popcorns 0.08 0.15 -0.07 -0.07 -0.07
15
restaurants 0.36 0.34 0.3 0.19 0.31
dish 0.4 0.24 0.25 0.21 0.28
16
road 0.19 0.17 0.27 0.22 0.29

```
place 0.3 0.27 0.4 0.3 0.43
wrap 0.23 0.19 0.2 0.19 0.3
they 0.27 0.28 0.45 0.31 0.36
rice 0.37 0.22 0.23 0.2 0.21
veggies 0.36 0.24 0.12 0.09 0.14
paneer 0.27 0.07 -0.01 -0.03 0.02
curry 0.4 0.2 0.19 0.16 0.16
breads 0.36 0.29 0.12 0.07 0.14
it 0.28 0.26 0.43 0.34 0.39
17
they 0.27 0.28 0.45 0.31 0.36
answer 0.19 0.18 0.35 0.24 0.25
anyone 0.24 0.25 0.41 0.33 0.31
bar 0.28 0.39 0.35 0.28 0.4
experience 0.25 0.25 0.41 0.3 0.4
18
experience 0.25 0.25 0.41 0.3 0.4
prawns 0.33 0.16 0.05 0.03 0.05
pork 0.4 0.22 0.17 0.13 0.14
bunch 0.21 0.23 0.34 0.25 0.26
things 0.26 0.26 0.4 0.3 0.36
tacos 0.34 0.25 0.12 0.09 0.11
potatoes 0.36 0.2 0.16 0.12 0.15
item 0.21 0.15 0.27 0.2 0.33
19
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
20
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
thing 0.26 0.24 0.41 0.35 0.35
one 0.27 0.28 0.44 0.36 0.39
card 0.16 0.16 0.3 0.23 0.27
music 0.2 0.25 0.31 0.5 0.34
experience 0.25 0.25 0.41 0.3 0.4
21
place 0.3 0.27 0.4 0.3 0.43
atmosphere 0.23 0.28 0.29 0.22 0.44
22
roof 0.15 0.2 0.24 0.17 0.4
restaurant 0.39 0.36 0.37 0.22 0.39
finger 0.22 0.22 0.22 0.24 0.24
music 0.2 0.25 0.31 0.5 0.34
mouth 0.23 0.22 0.23 0.2 0.24
ur 0.18 0.14 0.24 0.29 0.21
it 0.28 0.26 0.43 0.34 0.39
location 0.19 0.19 0.34 0.23 0.41
23
place 0.3 0.27 0.4 0.3 0.43
seating 0.17 0.23 0.23 0.12 0.42
beers 0.27 0.47 0.22 0.18 0.18
menu 0.36 0.3 0.3 0.2 0.33
value 0.21 0.2 0.34 0.23 0.33
24
it 0.28 0.26 0.43 0.34 0.39
it 0.28 0.26 0.43 0.34 0.39
25
arrangement 0.16 0.13 0.25 0.22 0.34
seating 0.17 0.23 0.23 0.12 0.42
music 0.2 0.25 0.31 0.5 0.34
night 0.27 0.33 0.35 0.35 0.36
quality 0.24 0.21 0.37 0.28 0.41
i 0.27 0.26 0.41 0.35 0.34
seekh 0.16 -0.04 -0.06 -0.05 -0.02
26
place 0.3 0.27 0.4 0.3 0.43
items 0.26 0.26 0.31 0.21 0.36
27
vibe 0.16 0.18 0.19 0.32 0.33
music 0.2 0.25 0.31 0.5 0.34
experience 0.25 0.25 0.41 0.3 0.4
we 0.28 0.28 0.46 0.32 0.39
```

```
ipl 0.06 0.05 0.13 0.1 0.09
screen 0.15 0.17 0.29 0.23 0.35
28
environment 0.14 0.19 0.36 0.21 0.4
volume 0.12 0.17 0.24 0.27 0.27
music 0.2 0.25 0.31 0.5 0.34
rabdi -0.02 -0.02 -0.14 -0.08 -0.19
disadvantage 0.08 0.01 0.17 0.05 0.09
shade 0.17 0.21 0.17 0.16 0.35
restaurant 0.39 0.36 0.37 0.22 0.39
29
raj 0.09 0.08 0.17 0.16 0.12
music 0.2 0.25 0.31 0.5 0.34
30
rooftop 0.13 0.27 0.2 0.13 0.3
one 0.27 0.28 0.44 0.36 0.39
interiors 0.15 0.18 0.2 0.11 0.48
music 0.2 0.25 0.31 0.5 0.34
they 0.27 0.28 0.45 0.31 0.36
option 0.22 0.19 0.34 0.21 0.35
price 0.22 0.19 0.3 0.25 0.36
side 0.24 0.23 0.34 0.27 0.41
place 0.3 0.27 0.4 0.3 0.43
31
32
quality 0.24 0.21 0.37 0.28 0.41
couches 0.09 0.16 0.09 0.03 0.26
music 0.2 0.25 0.31 0.5 0.34
songs 0.19 0.19 0.25 0.4 0.26
33
this 0.24 0.21 0.38 0.3 0.38
times 0.22 0.23 0.37 0.27 0.31
place 0.3 0.27 0.4 0.3 0.43
vibes 0.11 0.16 0.13 0.28 0.18
you 0.26 0.26 0.43 0.35 0.38
music 0.2 0.25 0.31 0.5 0.34
restaurant 0.39 0.36 0.37 0.22 0.39
34
it 0.28 0.26 0.43 0.34 0.39
experience 0.25 0.25 0.41 0.3 0.4
35
place 0.3 0.27 0.4 0.3 0.43
sureela 0 0 0 0 0
anchor 0.12 0.17 0.23 0.18 0.23
night 0.27 0.33 0.35 0.35 0.36
songs 0.19 0.19 0.25 0.4 0.26
time 0.25 0.24 0.41 0.31 0.36
36
time 0.25 0.24 0.41 0.31 0.36
place 0.3 0.27 0.4 0.3 0.43
37
place 0.3 0.27 0.4 0.3 0.43
time 0.25 0.24 0.41 0.31 0.36
38
time 0.25 0.24 0.41 0.31 0.36
prawns 0.33 0.16 0.05 0.03 0.05
hunan 0.17 0.04 0.04 0.0 0.06
39
40
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
sofas 0.13 0.16 0.1 0.04 0.29
couple 0.25 0.27 0.39 0.3 0.32
course 0.28 0.26 0.39 0.27 0.34
services 0.16 0.19 0.42 0.24 0.32
you 0.26 0.26 0.43 0.35 0.38
nap 0.13 0.12 0.16 0.11 0.17
experience 0.25 0.25 0.41 0.3 0.4
41
roof 0.15 0.2 0.24 0.17 0.4
seating 0.17 0.23 0.23 0.12 0.42
we 0.28 0.28 0.46 0.32 0.39
```

it 0.28 0.26 0.43 0.34 0.39
night 0.27 0.33 0.35 0.35 0.36
42
items 0.26 0.26 0.31 0.21 0.36
bo 0.13 0.09 0.21 0.26 0.18
charge 0.17 0.2 0.35 0.21 0.26
experience 0.25 0.25 0.41 0.3 0.4
43
rates 0.13 0.15 0.26 0.16 0.24
place 0.3 0.27 0.4 0.3 0.43
44
place 0.3 0.27 0.4 0.3 0.43
visit 0.22 0.22 0.35 0.25 0.35
occasion 0.25 0.25 0.29 0.25 0.32
charges 0.11 0.17 0.27 0.15 0.18
45
naan 0.27 0.08 0.02 0.04 0.04
course 0.28 0.26 0.39 0.27 0.34
pot 0.28 0.21 0.19 0.16 0.23
rice 0.37 0.22 0.23 0.2 0.21
46
music 0.2 0.25 0.31 0.5 0.34
shout 0.13 0.17 0.22 0.26 0.16
47
evening 0.27 0.33 0.33 0.3 0.36
rating 0.17 0.15 0.24 0.21 0.29
place 0.3 0.27 0.4 0.3 0.43
location 0.19 0.19 0.34 0.23 0.41
rooftop 0.13 0.27 0.2 0.13 0.3
area 0.2 0.23 0.36 0.26 0.44
parties 0.21 0.28 0.31 0.26 0.28
weather 0.15 0.19 0.24 0.19 0.28
hours 0.21 0.22 0.33 0.25 0.29
dow 0.03 0.09 0.15 0.13 0.09
course 0.28 0.26 0.39 0.27 0.34
48
hiaghland 0 0 0 0 0
you 0.26 0.26 0.43 0.35 0.38
vodka 0.26 0.43 0.15 0.18 0.14
shots 0.19 0.3 0.26 0.23 0.29
49
place 0.3 0.27 0.4 0.3 0.43
dish 0.4 0.24 0.25 0.21 0.28
thanks 0.21 0.17 0.36 0.29 0.3
50
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
hospitality 0.23 0.26 0.33 0.17 0.34
a 0.24 0.23 0.39 0.32 0.37
51
place 0.3 0.27 0.4 0.3 0.43
orders 0.2 0.21 0.3 0.16 0.23
liit -0.02 -0.04 -0.09 -0.06 -0.11
52
music 0.2 0.25 0.31 0.5 0.34
this 0.24 0.21 0.38 0.3 0.38
lot 0.23 0.23 0.39 0.29 0.37
53
menu 0.36 0.3 0.3 0.2 0.33
options 0.25 0.23 0.34 0.19 0.38
place 0.3 0.27 0.4 0.3 0.43
vibes 0.11 0.16 0.13 0.28 0.18
54
place 0.3 0.27 0.4 0.3 0.43
we 0.28 0.28 0.46 0.32 0.39
one 0.27 0.28 0.44 0.36 0.39
jus 0.21 0.15 0.15 0.21 0.11
grps -0.03 -0.03 -0.03 -0.03 -0.04
cz 0.07 0.1 0.11 0.15 0.12
hrs 0.15 0.11 0.2 0.15 0.17
thing 0.26 0.24 0.41 0.35 0.35
your 0.24 0.23 0.39 0.31 0.39

places 0.27 0.29 0.36 0.27 0.38
options 0.25 0.23 0.34 0.19 0.38
55
gate 0.15 0.16 0.23 0.18 0.29
we 0.28 0.28 0.46 0.32 0.39
wheat 0.26 0.21 0.16 0.14 0.16
gro 0.01 0.03 0.05 0.08 0.05
it 0.28 0.26 0.43 0.34 0.39
stout 0.18 0.27 0.15 0.12 0.12
platter 0.35 0.21 0.16 0.11 0.22
chicken 0.46 0.27 0.22 0.18 0.22
course 0.28 0.26 0.39 0.27 0.34
egg 0.3 0.23 0.19 0.19 0.22
naan 0.27 0.08 0.02 0.04 0.04
sizler 0 0 0 0 0
staffs 0.09 0.08 0.27 0.05 0.13
experience 0.25 0.25 0.41 0.3 0.4
56
57
music 0.2 0.25 0.31 0.5 0.34
staffs 0.09 0.08 0.27 0.05 0.13
who 0.21 0.24 0.44 0.35 0.3
i 0.27 0.26 0.41 0.35 0.34
place 0.3 0.27 0.4 0.3 0.43
58
they 0.27 0.28 0.45 0.31 0.36
membership 0.1 0.14 0.3 0.2 0.19
59
organizer 0.07 0.08 0.23 0.17 0.24
60
playlist 0.13 0.14 0.18 0.31 0.18
mood 0.2 0.22 0.22 0.26 0.34
it 0.28 0.26 0.43 0.34 0.39
61
it 0.28 0.26 0.43 0.34 0.39
lot 0.23 0.23 0.39 0.29 0.37
place 0.3 0.27 0.4 0.3 0.43
party 0.24 0.31 0.34 0.34 0.34
rajguru -0.06 -0.04 -0.09 -0.06 -0.07
they 0.27 0.28 0.45 0.31 0.36
we 0.28 0.28 0.46 0.32 0.39
music 0.2 0.25 0.31 0.5 0.34
vibe 0.16 0.18 0.19 0.32 0.33
i 0.27 0.26 0.41 0.35 0.34
fan 0.2 0.18 0.31 0.3 0.31
decors 0.08 0.09 -0.03 -0.03 0.25
music 0.2 0.25 0.31 0.5 0.34
they 0.27 0.28 0.45 0.31 0.36
events 0.18 0.27 0.33 0.3 0.29
go 0.27 0.27 0.4 0.33 0.34
:d 0.2 0.18 0.23 0.27 0.19
62
board 0.16 0.18 0.34 0.21 0.31
it 0.28 0.26 0.43 0.34 0.39
i 0.27 0.26 0.41 0.35 0.34
restaurant 0.39 0.36 0.37 0.22 0.39
day 0.27 0.28 0.38 0.31 0.36
singing 0.19 0.19 0.25 0.38 0.23
everything 0.28 0.26 0.4 0.33 0.4
rooftops 0.05 0.14 0.08 0.03 0.16
place 0.3 0.27 0.4 0.3 0.43
collection 0.17 0.18 0.27 0.25 0.37
them 0.27 0.26 0.42 0.3 0.35
mary 0.15 0.18 0.26 0.25 0.24
bull 0.16 0.18 0.25 0.21 0.21
they 0.27 0.28 0.45 0.31 0.36
which 0.27 0.28 0.42 0.32 0.4
presentation 0.2 0.19 0.34 0.25 0.34
chicken 0.46 0.27 0.22 0.18 0.22
wrap 0.23 0.19 0.2 0.19 0.3
these 0.25 0.28 0.4 0.28 0.36
highland 0.14 0.17 0.21 0.2 0.25

```
veg 0.34 0.21 0.13 0.06 0.15
lazizi 0 0 0 0 0
desserts 0.43 0.4 0.18 0.14 0.22
63
date 0.16 0.15 0.29 0.25 0.25
you 0.26 0.26 0.43 0.35 0.38
place 0.3 0.27 0.4 0.3 0.43
options 0.25 0.23 0.34 0.19 0.38
vegetarians 0.23 0.17 0.11 0.04 0.06
64
usp 0.04 0.08 0.08 0.08 0.04
highland 0.14 0.17 0.21 0.2 0.25
tikke 0 0 0 0 0
vibe 0.16 0.18 0.19 0.32 0.33
i 0.27 0.26 0.41 0.35 0.34
experience 0.25 0.25 0.41 0.3 0.4
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
rooftop 0.13 0.27 0.2 0.13 0.3
high 0.19 0.21 0.33 0.29 0.34
65
place 0.3 0.27 0.4 0.3 0.43
swaad 0 0 0 0 0
interiors 0.15 0.18 0.2 0.11 0.48
people 0.24 0.27 0.43 0.31 0.36
restaurant 0.39 0.36 0.37 0.22 0.39
66
it 0.28 0.26 0.43 0.34 0.39
rice 0.37 0.22 0.23 0.2 0.21
schezwan 0.04 -0.02 -0.1 -0.09 -0.07
noodles 0.37 0.23 0.12 0.13 0.13
you 0.26 0.26 0.43 0.35 0.38
67
neighborhood 0.17 0.21 0.29 0.21 0.32
bar 0.28 0.39 0.35 0.28 0.4
place 0.3 0.27 0.4 0.3 0.43
vibe 0.16 0.18 0.19 0.32 0.33
aswhin 0 0 0 0 0
nupur -0.01 -0.02 -0.01 0.05 -0.03
kunal 0.04 0.03 0.04 0.08 0.01
burrow 0.03 -0.0 0.05 0.04 0.1
night 0.27 0.33 0.35 0.35 0.36
68
atmosphere 0.23 0.28 0.29 0.22 0.44
time 0.25 0.24 0.41 0.31 0.36
69
highland 0.14 0.17 0.21 0.2 0.25
place 0.3 0.27 0.4 0.3 0.43
menu 0.36 0.3 0.3 0.2 0.33
space 0.17 0.23 0.33 0.26 0.44
that 0.25 0.27 0.44 0.34 0.39
hotel 0.2 0.26 0.31 0.19 0.38
roofs 0.09 0.15 0.12 0.05 0.26
70
party 0.24 0.31 0.34 0.34 0.34
swing 0.14 0.16 0.23 0.3 0.27
it 0.28 0.26 0.43 0.34 0.39
we 0.28 0.28 0.46 0.32 0.39
for 0.25 0.25 0.4 0.3 0.35
place 0.3 0.27 0.4 0.3 0.43
roof 0.15 0.2 0.24 0.17 0.4
building 0.16 0.24 0.35 0.22 0.42
screen 0.15 0.17 0.29 0.23 0.35
he 0.23 0.23 0.4 0.35 0.32
special 0.28 0.29 0.38 0.31 0.36
which 0.27 0.28 0.42 0.32 0.4
vodka 0.26 0.43 0.15 0.18 0.14
mocktail 0.11 0.24 -0.03 -0.03 -0.06
detox 0.15 0.18 0.13 0.11 0.12
tea 0.32 0.36 0.25 0.21 0.3
juice 0.31 0.37 0.21 0.2 0.19
drink 0.34 0.54 0.29 0.25 0.27
```

speciality 0.28 0.26 0.16 0.11 0.19
tikkas 0.08 -0.0 -0.08 -0.08 -0.11
skewer 0.22 0.06 0.04 -0.02 0.01
coal 0.1 0.07 0.17 0.1 0.2
time 0.25 0.24 0.41 0.31 0.36
boozing 0.02 0.12 0.0 -0.0 -0.03
experience 0.25 0.25 0.41 0.3 0.4
71
place 0.3 0.27 0.4 0.3 0.43
days 0.22 0.23 0.35 0.28 0.31
it 0.28 0.26 0.43 0.34 0.39
rooftop 0.13 0.27 0.2 0.13 0.3
ones 0.28 0.27 0.38 0.28 0.33
sauce 0.43 0.28 0.18 0.16 0.18
chicken 0.46 0.27 0.22 0.18 0.22
spices 0.37 0.25 0.12 0.11 0.19
noodle 0.34 0.23 0.12 0.12 0.13
pina 0.15 0.24 0.04 0.1 0.03
which 0.27 0.28 0.42 0.32 0.4
it 0.28 0.26 0.43 0.34 0.39
72
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
offers 0.23 0.26 0.38 0.26 0.43
preference 0.21 0.14 0.26 0.15 0.25
73
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
restaurant 0.39 0.36 0.37 0.22 0.39
rooftoop 0 0 0 0 0
74
times 0.22 0.23 0.37 0.27 0.31
variety 0.31 0.32 0.36 0.27 0.38
time 0.25 0.24 0.41 0.31 0.36
brews 0.22 0.34 0.13 0.15 0.09
this 0.24 0.21 0.38 0.3 0.38
music 0.2 0.25 0.31 0.5 0.34
entrance 0.15 0.21 0.24 0.17 0.4
screening 0.09 0.15 0.22 0.15 0.17
aspect 0.15 0.08 0.26 0.2 0.34
they 0.27 0.28 0.45 0.31 0.36
basket 0.26 0.18 0.21 0.16 0.3
presentation 0.2 0.19 0.34 0.25 0.34
platter 0.35 0.21 0.16 0.11 0.22
hummus 0.32 0.17 0.06 0.06 0.08
pita 0.29 0.12 0.1 0.06 0.09
style 0.3 0.26 0.32 0.35 0.48
pomfret 0.08 0.0 0.01 0.02 0.01
fish 0.33 0.24 0.27 0.19 0.28
spices 0.37 0.25 0.12 0.11 0.19
sauces 0.38 0.3 0.1 0.09 0.13
chicken 0.46 0.27 0.22 0.18 0.22
naan 0.27 0.08 0.02 0.04 0.04
nothing 0.26 0.26 0.39 0.33 0.36
cravings 0.2 0.17 0.09 0.07 0.1
butter 0.35 0.23 0.17 0.16 0.21
alcohol 0.2 0.34 0.23 0.17 0.19
craft 0.22 0.28 0.25 0.25 0.33
beers 0.27 0.47 0.22 0.18 0.18
i 0.27 0.26 0.41 0.35 0.34
sangria 0.24 0.3 0.06 0.04 0.12
liit -0.02 -0.04 -0.09 -0.06 -0.11
mojito 0.21 0.28 0.06 0.1 0.06
job 0.18 0.16 0.39 0.26 0.32
anand 0.04 0.03 0.14 0.15 0.09
rating 0.17 0.15 0.24 0.21 0.29
places 0.27 0.29 0.36 0.27 0.38
it 0.28 0.26 0.43 0.34 0.39
alcohol 0.2 0.34 0.23 0.17 0.19
guys 0.22 0.24 0.39 0.33 0.29
75

music 0.2 0.25 0.31 0.5 0.34
hospitality 0.23 0.26 0.33 0.17 0.34
which 0.27 0.28 0.42 0.32 0.4
platter 0.35 0.21 0.16 0.11 0.22
chef 0.35 0.23 0.31 0.27 0.27
seekh 0.16 -0.04 -0.06 -0.05 -0.02
fries 0.38 0.28 0.15 0.13 0.11
course 0.28 0.26 0.39 0.27 0.34
shanghai 0.16 0.15 0.19 0.15 0.21
veg 0.34 0.21 0.13 0.06 0.15
rice 0.37 0.22 0.23 0.2 0.21
garlic 0.35 0.18 0.13 0.1 0.14
76
restaurant 0.39 0.36 0.37 0.22 0.39
it 0.28 0.26 0.43 0.34 0.39
men 0.16 0.18 0.29 0.25 0.25
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
77
they 0.27 0.28 0.45 0.31 0.36
tables 0.19 0.26 0.29 0.17 0.36
which 0.27 0.28 0.42 0.32 0.4
place 0.3 0.27 0.4 0.3 0.43
bar 0.28 0.39 0.35 0.28 0.4
dining 0.31 0.34 0.29 0.19 0.47
menu 0.36 0.3 0.3 0.2 0.33
capriozka 0 0 0 0 0
screwdriver 0.08 0.09 0.09 0.09 0.08
mojito 0.21 0.28 0.06 0.1 0.06
murg 0.07 -0.05 -0.09 -0.1 -0.11
noodles 0.37 0.23 0.12 0.13 0.13
chicken 0.46 0.27 0.22 0.18 0.22
managers 0.1 0.13 0.41 0.13 0.23
it 0.28 0.26 0.43 0.34 0.39
cocktail 0.32 0.46 0.24 0.25 0.28
78
highball 0.09 0.2 0.0 0.03 0.02
place 0.3 0.27 0.4 0.3 0.43
look 0.25 0.24 0.39 0.3 0.45
rooftop 0.13 0.27 0.2 0.13 0.3
meny 0.03 0.04 -0.02 -0.03 -0.04
names 0.19 0.21 0.36 0.29 0.25
mocktail 0.11 0.24 -0.03 -0.03 -0.06
apple 0.23 0.22 0.3 0.22 0.23
fiz -0.01 0.01 0.0 -0.0 -0.06
jalapeos 0 0 0 0 0
ones 0.28 0.27 0.38 0.28 0.33
this 0.24 0.21 0.38 0.3 0.38
starter 0.23 0.17 0.23 0.19 0.2
start 0.24 0.24 0.4 0.3 0.32
menu 0.36 0.3 0.3 0.2 0.33
it 0.28 0.26 0.43 0.34 0.39
this 0.24 0.21 0.38 0.3 0.38
combo 0.23 0.2 0.2 0.27 0.25
cheese 0.43 0.3 0.22 0.19 0.21
nachos 0.32 0.27 0.08 0.03 0.06
plate 0.3 0.2 0.25 0.18 0.33
nachos 0.32 0.27 0.08 0.03 0.06
garnishing 0.14 0.1 0.03 -0.03 0.04
fries 0.38 0.28 0.15 0.13 0.11
combination 0.26 0.24 0.29 0.25 0.36
toppings 0.33 0.21 0.08 0.04 0.13
suey 0.13 0.02 -0.02 0.06 0.01
khao 0.09 0.06 0.03 0.03 0.06
onions 0.33 0.16 0.12 0.08 0.09
groundnuts 0.03 -0.02 -0.07 -0.05 -0.06
onion 0.32 0.16 0.16 0.12 0.13
stem 0.12 0.13 0.17 0.13 0.17
coriander 0.24 0.11 0.03 0.06 0.07
lemon 0.32 0.28 0.18 0.15 0.21
crums 0.01 0.01 -0.09 -0.09 -0.09
curry 0.4 0.2 0.19 0.16 0.16

veggies 0.36 0.24 0.12 0.09 0.14
cheese 0.43 0.3 0.22 0.19 0.21
base 0.22 0.18 0.3 0.22 0.36
topping 0.29 0.23 0.15 0.17 0.17
sauce 0.43 0.28 0.18 0.16 0.18
ice 0.28 0.33 0.28 0.25 0.27
79
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
services 0.16 0.19 0.42 0.24 0.32
time 0.25 0.24 0.41 0.31 0.36
80
place 0.3 0.27 0.4 0.3 0.43
baner -0.05 -0.02 -0.05 -0.03 -0.02
entrance 0.15 0.21 0.24 0.17 0.4
bites 0.28 0.24 0.2 0.16 0.14
licking 0.13 0.11 0.12 0.12 0.14
pomfret 0.08 0.0 0.01 0.02 0.01
options 0.25 0.23 0.34 0.19 0.38
claypot 0.15 0.05 -0.06 -0.07 -0.04
81
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
vibes 0.11 0.16 0.13 0.28 0.18
conversation 0.2 0.23 0.32 0.25 0.29
82
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
tracks 0.14 0.15 0.25 0.37 0.28
which 0.27 0.28 0.42 0.32 0.4
rum 0.24 0.36 0.14 0.17 0.12
juices 0.29 0.35 0.13 0.09 0.13
essence 0.21 0.17 0.21 0.22 0.27
ones 0.28 0.27 0.38 0.28 0.33
83
place 0.3 0.27 0.4 0.3 0.43
rooftop 0.13 0.27 0.2 0.13 0.3
one 0.27 0.28 0.44 0.36 0.39
section 0.18 0.19 0.31 0.23 0.32
you 0.26 0.26 0.43 0.35 0.38
beat 0.2 0.19 0.3 0.43 0.24
part 0.23 0.21 0.38 0.33 0.36
i 0.27 0.26 0.41 0.35 0.34
options 0.25 0.23 0.34 0.19 0.38
this 0.24 0.21 0.38 0.3 0.38
it 0.28 0.26 0.43 0.34 0.39
84
place 0.3 0.27 0.4 0.3 0.43
rates 0.13 0.15 0.26 0.16 0.24
85
baner -0.05 -0.02 -0.05 -0.03 -0.02
party 0.24 0.31 0.34 0.34 0.34
altitude 0.1 0.12 0.15 0.12 0.17
place 0.3 0.27 0.4 0.3 0.43
octane 0.05 0.13 0.08 0.1 0.07
highland 0.14 0.17 0.21 0.2 0.25
i 0.27 0.26 0.41 0.35 0.34
couple 0.25 0.27 0.39 0.3 0.32
experience 0.25 0.25 0.41 0.3 0.4
entry 0.18 0.21 0.33 0.25 0.37
rooftop 0.13 0.27 0.2 0.13 0.3
seating 0.17 0.23 0.23 0.12 0.42
hour 0.24 0.27 0.34 0.28 0.29
note 0.21 0.19 0.36 0.3 0.35
jack 0.17 0.2 0.29 0.3 0.23
it 0.28 0.26 0.43 0.34 0.39
mood 0.2 0.22 0.22 0.26 0.34
vodka 0.26 0.43 0.15 0.18 0.14
gin 0.2 0.35 0.15 0.19 0.1
tequila 0.22 0.36 0.15 0.14 0.12
rum 0.24 0.36 0.14 0.17 0.12
bull 0.16 0.18 0.25 0.21 0.21

friends 0.23 0.28 0.4 0.34 0.3
spices 0.37 0.25 0.12 0.11 0.19
cooking 0.4 0.26 0.27 0.22 0.3
blast 0.17 0.25 0.23 0.26 0.25
beauty 0.21 0.19 0.26 0.26 0.38
chicken 0.46 0.27 0.22 0.18 0.22
this 0.24 0.21 0.38 0.3 0.38
sauce 0.43 0.28 0.18 0.16 0.18
you 0.26 0.26 0.43 0.35 0.38
makhni 0.05 -0.05 -0.06 -0.07 -0.07
sugar 0.27 0.28 0.23 0.23 0.24
balance 0.2 0.2 0.29 0.21 0.31
paneer 0.27 0.07 -0.01 -0.03 0.02
preparations 0.22 0.2 0.18 0.09 0.18
rice 0.37 0.22 0.23 0.2 0.21
they 0.27 0.28 0.45 0.31 0.36
ishpreet 0 0 0 0 0
host 0.19 0.23 0.42 0.3 0.25
he 0.23 0.23 0.4 0.35 0.32
time 0.25 0.24 0.41 0.31 0.36
86
highland 0.14 0.17 0.21 0.2 0.25
it 0.28 0.26 0.43 0.34 0.39
try 0.3 0.27 0.38 0.29 0.32
rates 0.13 0.15 0.26 0.16 0.24
experience 0.25 0.25 0.41 0.3 0.4
i 0.27 0.26 0.41 0.35 0.34
visitor 0.14 0.18 0.3 0.16 0.29
87
menu 0.36 0.3 0.3 0.2 0.33
it 0.28 0.26 0.43 0.34 0.39
we 0.28 0.28 0.46 0.32 0.39
lot 0.23 0.23 0.39 0.29 0.37
88
place 0.3 0.27 0.4 0.3 0.43
street 0.17 0.23 0.27 0.28 0.35
parking 0.15 0.23 0.3 0.14 0.36
valet 0.09 0.14 0.21 0.09 0.21
rooftop 0.13 0.27 0.2 0.13 0.3
collection 0.17 0.18 0.27 0.25 0.37
89
place 0.3 0.27 0.4 0.3 0.43
it 0.28 0.26 0.43 0.34 0.39
doubt 0.21 0.19 0.35 0.3 0.27
place 0.3 0.27 0.4 0.3 0.43
90
night 0.27 0.33 0.35 0.35 0.36
friends 0.23 0.28 0.4 0.34 0.3
experience 0.25 0.25 0.41 0.3 0.4
options 0.25 0.23 0.34 0.19 0.38
marks 0.16 0.15 0.24 0.21 0.25
91
place 0.3 0.27 0.4 0.3 0.43
rooftop 0.13 0.27 0.2 0.13 0.3
dragon 0.17 0.16 0.22 0.22 0.24
corn 0.28 0.21 0.17 0.15 0.19
atmosphere 0.23 0.28 0.29 0.22 0.44
you 0.26 0.26 0.43 0.35 0.38
fan 0.2 0.18 0.31 0.3 0.31
92
highland 0.14 0.17 0.21 0.2 0.25
bars 0.28 0.37 0.26 0.23 0.3
dazling -0.08 -0.04 -0.16 -0.1 -0.16
place 0.3 0.27 0.4 0.3 0.43
area 0.2 0.23 0.36 0.26 0.44
space 0.17 0.23 0.33 0.26 0.44
seating 0.17 0.23 0.23 0.12 0.42
counter 0.2 0.25 0.28 0.16 0.32
sets 0.19 0.21 0.31 0.3 0.37
lights 0.16 0.25 0.24 0.24 0.41
booth 0.15 0.23 0.29 0.23 0.31
times 0.22 0.23 0.37 0.27 0.31

you 0.26 0.26 0.43 0.35 0.38
memories 0.19 0.2 0.23 0.24 0.26
chilly 0.2 0.19 0.13 0.14 0.22
enjoyy -0.07 -0.07 -0.13 -0.07 -0.05
this 0.24 0.21 0.38 0.3 0.38
paneer 0.27 0.07 -0.01 -0.03 0.02
cubes 0.26 0.21 0.11 0.1 0.18
button 0.14 0.13 0.24 0.23 0.28
mushrooms 0.3 0.19 0.12 0.08 0.13
they 0.27 0.28 0.45 0.31 0.36
choice 0.33 0.27 0.39 0.28 0.39
poppers 0.15 0.14 0.04 0.06 0.07
dish 0.4 0.24 0.25 0.21 0.28
anything 0.27 0.25 0.37 0.31 0.33
ones 0.28 0.27 0.38 0.28 0.33
nachos 0.32 0.27 0.08 0.03 0.06
beans 0.33 0.23 0.18 0.17 0.16
salsa 0.31 0.23 0.16 0.3 0.15
jalapeos 0 0 0 0 0
peppers 0.29 0.17 0.12 0.15 0.11
veggies 0.36 0.24 0.12 0.09 0.14
curry 0.4 0.2 0.19 0.16 0.16
consistency 0.22 0.12 0.22 0.1 0.19
aroma 0.26 0.21 0.11 0.09 0.24
it 0.28 0.26 0.43 0.34 0.39
balance 0.2 0.2 0.29 0.21 0.31
spices 0.37 0.25 0.12 0.11 0.19
breads 0.36 0.29 0.12 0.07 0.14
layers 0.18 0.12 0.17 0.16 0.28
grain 0.23 0.17 0.18 0.13 0.24
rice 0.37 0.22 0.23 0.2 0.21
flavors 0.39 0.34 0.17 0.15 0.23
drink 0.34 0.54 0.29 0.25 0.27
pulp 0.15 0.14 0.12 0.12 0.14
juices 0.29 0.35 0.13 0.09 0.13
lime 0.28 0.28 0.16 0.14 0.26
it 0.28 0.26 0.43 0.34 0.39
mary 0.15 0.18 0.26 0.25 0.24
time 0.25 0.24 0.41 0.31 0.36
which 0.27 0.28 0.42 0.32 0.4
thing 0.26 0.24 0.41 0.35 0.35
value 0.21 0.2 0.34 0.23 0.33
93
place 0.3 0.27 0.4 0.3 0.43
comfort 0.22 0.19 0.26 0.18 0.4
essence 0.21 0.17 0.21 0.22 0.27
menu 0.36 0.3 0.3 0.2 0.33
94
it 0.28 0.26 0.43 0.34 0.39
mic 0.08 0.15 0.21 0.31 0.19
lot 0.23 0.23 0.39 0.29 0.37
singers 0.13 0.14 0.23 0.34 0.17
i 0.27 0.26 0.41 0.35 0.34
lil 0.14 0.12 0.2 0.32 0.19
people 0.24 0.27 0.43 0.31 0.36
person 0.22 0.23 0.39 0.31 0.33
repeat 0.16 0.14 0.24 0.23 0.17
ditto 0.14 0.08 0.17 0.17 0.09
crowd 0.19 0.26 0.32 0.3 0.29
vodka 0.26 0.43 0.15 0.18 0.14
cocktail 0.32 0.46 0.24 0.25 0.28
time 0.25 0.24 0.41 0.31 0.36
liitagain 0 0 0 0 0
drink 0.34 0.54 0.29 0.25 0.27
liit -0.02 -0.04 -0.09 -0.06 -0.11
picturepresentation 0 0 0 0 0
difference 0.23 0.18 0.34 0.25 0.29
devil 0.17 0.17 0.23 0.26 0.18
chickenmoderately 0 0 0 0 0
95
rooftop 0.13 0.27 0.2 0.13 0.3
restaurant 0.39 0.36 0.37 0.22 0.39

```
murg 0.07 -0.05 -0.09 -0.1 -0.11
place 0.3 0.27 0.4 0.3 0.43
hangout 0.1 0.18 0.16 0.16 0.14
job 0.18 0.16 0.39 0.26 0.32
96
restaurant 0.39 0.36 0.37 0.22 0.39
97
night 0.27 0.33 0.35 0.35 0.36
place 0.3 0.27 0.4 0.3 0.43
menu 0.36 0.3 0.3 0.2 0.33
time 0.25 0.24 0.41 0.31 0.36
ravioli 0.31 0.13 0.04 0.02 0.04
roast 0.38 0.23 0.16 0.14 0.17
cake 0.38 0.3 0.24 0.24 0.29
course 0.28 0.26 0.39 0.27 0.34
tooth 0.16 0.12 0.16 0.15 0.2
mr 0.16 0.15 0.29 0.27 0.23
98
place 0.3 0.27 0.4 0.3 0.43
things 0.26 0.26 0.4 0.3 0.36
space 0.17 0.23 0.33 0.26 0.44
99
place 0.3 0.27 0.4 0.3 0.43
it 0.28 0.26 0.43 0.34 0.39
fun 0.26 0.31 0.34 0.32 0.37
life 0.21 0.21 0.35 0.32 0.34
highland 0.14 0.17 0.21 0.2 0.25
raj 0.09 0.08 0.17 0.16 0.12
waiter 0.27 0.26 0.3 0.13 0.19
100
display 0.18 0.19 0.29 0.22 0.37
color 0.2 0.17 0.24 0.23 0.37
tables 0.19 0.26 0.29 0.17 0.36
you 0.26 0.26 0.43 0.35 0.38
menu 0.36 0.3 0.3 0.2 0.33
quality 0.24 0.21 0.37 0.28 0.41
people 0.24 0.27 0.43 0.31 0.36
place 0.3 0.27 0.4 0.3 0.43
101
music 0.2 0.25 0.31 0.5 0.34
place 0.3 0.27 0.4 0.3 0.43
102
chicken 0.46 0.27 0.22 0.18 0.22
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
103
baner -0.05 -0.02 -0.05 -0.03 -0.02
road 0.19 0.17 0.27 0.22 0.29
it 0.28 0.26 0.43 0.34 0.39
problem 0.18 0.16 0.38 0.23 0.29
they 0.27 0.28 0.45 0.31 0.36
parking 0.15 0.23 0.3 0.14 0.36
floors 0.14 0.21 0.21 0.16 0.46
rooftop 0.13 0.27 0.2 0.13 0.3
speed 0.16 0.16 0.29 0.24 0.26
mint 0.24 0.22 0.2 0.19 0.23
chutney 0.31 0.16 0.06 0.08 0.06
salad 0.45 0.32 0.2 0.15 0.23
flavor 0.39 0.31 0.19 0.19 0.26
puri 0.13 0.08 0.06 0.05 0.1
sauce 0.43 0.28 0.18 0.16 0.18
wedges 0.19 0.13 0.09 0.11 0.16
fish 0.33 0.24 0.27 0.19 0.28
nothing 0.26 0.26 0.39 0.33 0.36
which 0.27 0.28 0.42 0.32 0.4
bread 0.42 0.28 0.23 0.19 0.21
sausages 0.36 0.26 0.09 0.04 0.06
dough 0.28 0.15 0.14 0.13 0.13
dish 0.4 0.24 0.25 0.21 0.28
one 0.27 0.28 0.44 0.36 0.39
chicken 0.46 0.27 0.22 0.18 0.22
flavors 0.39 0.34 0.17 0.15 0.23
```

roast 0.38 0.23 0.16 0.14 0.17
noodles 0.37 0.23 0.12 0.13 0.13
curry 0.4 0.2 0.19 0.16 0.16
coconut 0.32 0.29 0.15 0.13 0.23
they 0.27 0.28 0.45 0.31 0.36
beers 0.27 0.47 0.22 0.18 0.18
craft 0.22 0.28 0.25 0.25 0.33
place 0.3 0.27 0.4 0.3 0.43
improvement 0.15 0.11 0.3 0.17 0.33
104
they 0.27 0.28 0.45 0.31 0.36
it 0.28 0.26 0.43 0.34 0.39
time 0.25 0.24 0.41 0.31 0.36
105
place 0.3 0.27 0.4 0.3 0.43
we 0.28 0.28 0.46 0.32 0.39
stomach 0.19 0.19 0.16 0.13 0.19
music 0.2 0.25 0.31 0.5 0.34
music 0.2 0.25 0.31 0.5 0.34
106
foot 0.16 0.16 0.24 0.22 0.34
107
place 0.3 0.27 0.4 0.3 0.43
rooftop 0.13 0.27 0.2 0.13 0.3
island 0.2 0.24 0.28 0.22 0.31
tea 0.32 0.36 0.25 0.21 0.3
devil 0.17 0.17 0.23 0.26 0.18
108
tables 0.19 0.26 0.29 0.17 0.36
music 0.2 0.25 0.31 0.5 0.34
guys 0.22 0.24 0.39 0.33 0.29
109
colour 0.22 0.19 0.21 0.21 0.36
quality 0.24 0.21 0.37 0.28 0.41
price 0.22 0.19 0.3 0.25 0.36
110
evening 0.27 0.33 0.33 0.3 0.36
it 0.28 0.26 0.43 0.34 0.39
arrangement 0.16 0.13 0.25 0.22 0.34
that 0.25 0.27 0.44 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
portions 0.23 0.18 0.2 0.1 0.24
quality 0.24 0.21 0.37 0.28 0.41
table 0.25 0.27 0.33 0.24 0.41
job 0.18 0.16 0.39 0.26 0.32
thing 0.26 0.24 0.41 0.35 0.35
111
place 0.3 0.27 0.4 0.3 0.43
chicken 0.46 0.27 0.22 0.18 0.22
skin 0.18 0.15 0.21 0.17 0.28
which 0.27 0.28 0.42 0.32 0.4
112
you 0.26 0.26 0.43 0.35 0.38
order 0.26 0.21 0.35 0.24 0.33
113
place 0.3 0.27 0.4 0.3 0.43
menu 0.36 0.3 0.3 0.2 0.33
hours 0.21 0.22 0.33 0.25 0.29
114
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
time 0.25 0.24 0.41 0.31 0.36
services 0.16 0.19 0.42 0.24 0.32
115
place 0.3 0.27 0.4 0.3 0.43
it 0.28 0.26 0.43 0.34 0.39
nights 0.24 0.32 0.26 0.28 0.28
time 0.25 0.24 0.41 0.31 0.36
songs 0.19 0.19 0.25 0.4 0.26
prices 0.18 0.19 0.27 0.21 0.3
116
place 0.3 0.27 0.4 0.3 0.43

it 0.28 0.26 0.43 0.34 0.39
music 0.2 0.25 0.31 0.5 0.34
it 0.28 0.26 0.43 0.34 0.39
whiskey 0.22 0.37 0.15 0.17 0.13
lots 0.28 0.33 0.37 0.28 0.4
things 0.26 0.26 0.4 0.3 0.36
117
highland 0.14 0.17 0.21 0.2 0.25
weekend 0.24 0.28 0.34 0.29 0.3
bloggers 0.16 0.19 0.31 0.21 0.15
place 0.3 0.27 0.4 0.3 0.43
vibes 0.11 0.16 0.13 0.28 0.18
breeze 0.17 0.26 0.21 0.18 0.31
this 0.24 0.21 0.38 0.3 0.38
i 0.27 0.26 0.41 0.35 0.34
drink 0.34 0.54 0.29 0.25 0.27
side 0.24 0.23 0.34 0.27 0.41
pieces 0.29 0.21 0.25 0.25 0.36
chicken 0.46 0.27 0.22 0.18 0.22
it 0.28 0.26 0.43 0.34 0.39
course 0.28 0.26 0.39 0.27 0.34
makhmali 0 0 0 0 0
rooftop 0.13 0.27 0.2 0.13 0.3
118
cocktail 0.32 0.46 0.24 0.25 0.28
joker 0.07 0.08 0.13 0.18 0.1
which 0.27 0.28 0.42 0.32 0.4
119
restaurant 0.39 0.36 0.37 0.22 0.39
i 0.27 0.26 0.41 0.35 0.34
music 0.2 0.25 0.31 0.5 0.34
minutes 0.24 0.22 0.31 0.24 0.28
which 0.27 0.28 0.42 0.32 0.4
5 0.24 0.26 0.34 0.31 0.36
we 0.28 0.28 0.46 0.32 0.39
world 0.21 0.23 0.36 0.32 0.33
they 0.27 0.28 0.45 0.31 0.36
we 0.28 0.28 0.46 0.32 0.39
place 0.3 0.27 0.4 0.3 0.43
i 0.27 0.26 0.41 0.35 0.34
stars 0.19 0.22 0.28 0.28 0.28
it 0.28 0.26 0.43 0.34 0.39
120
waiter 0.27 0.26 0.3 0.13 0.19
i 0.27 0.26 0.41 0.35 0.34
smell 0.27 0.23 0.18 0.13 0.28
he 0.23 0.23 0.4 0.35 0.32
121
this 0.24 0.21 0.38 0.3 0.38
place 0.3 0.27 0.4 0.3 0.43
enjoying 0.26 0.32 0.28 0.24 0.32
chicken 0.46 0.27 0.22 0.18 0.22
ghee 0.19 0.04 0.02 0.02 0.02
salad 0.45 0.32 0.2 0.15 0.23
cheese 0.43 0.3 0.22 0.19 0.21
122
rabbit 0.2 0.14 0.18 0.17 0.21
find 0.24 0.22 0.41 0.34 0.38
music 0.2 0.25 0.31 0.5 0.34
well 0.3 0.31 0.46 0.35 0.43
123
i 0.27 0.26 0.41 0.35 0.34
city 0.19 0.23 0.35 0.3 0.35
me 0.25 0.25 0.4 0.36 0.34
gem 0.17 0.15 0.23 0.22 0.26
you 0.26 0.26 0.43 0.35 0.38
type 0.21 0.21 0.33 0.29 0.36
people 0.24 0.27 0.43 0.31 0.36
day 0.27 0.28 0.38 0.31 0.36
which 0.27 0.28 0.42 0.32 0.4
rendition 0.13 0.08 0.15 0.23 0.18
time 0.25 0.24 0.41 0.31 0.36

comfort 0.22 0.19 0.26 0.18 0.4
portion 0.2 0.16 0.25 0.17 0.27
places 0.27 0.29 0.36 0.27 0.38
bar 0.28 0.39 0.35 0.28 0.4
menu 0.36 0.3 0.3 0.2 0.33
cocktail 0.32 0.46 0.24 0.25 0.28
hospitality 0.23 0.26 0.33 0.17 0.34
place 0.3 0.27 0.4 0.3 0.43
karaokes 0.01 0.0 -0.08 -0.01 -0.08
music 0.2 0.25 0.31 0.5 0.34
what 0.25 0.24 0.42 0.34 0.37
shot 0.17 0.24 0.27 0.25 0.28
all 0.26 0.28 0.43 0.33 0.41
rb 0.08 0.03 0.15 0.18 0.11
brick 0.17 0.19 0.21 0.19 0.36
it 0.28 0.26 0.43 0.34 0.39
baner -0.05 -0.02 -0.05 -0.03 -0.02
ball 0.18 0.19 0.27 0.28 0.28
124
time 0.25 0.24 0.41 0.31 0.36
music 0.2 0.25 0.31 0.5 0.34
devil 0.17 0.17 0.23 0.26 0.18
dragon 0.17 0.16 0.22 0.22 0.24
murg 0.07 -0.05 -0.09 -0.1 -0.11
it 0.28 0.26 0.43 0.34 0.39
crispies 0.09 0.01 -0.09 -0.04 -0.07
a 0.24 0.23 0.39 0.32 0.37
they 0.27 0.28 0.45 0.31 0.36
way 0.26 0.25 0.41 0.33 0.38
it 0.28 0.26 0.43 0.34 0.39
apple 0.23 0.22 0.3 0.22 0.23
highland 0.14 0.17 0.21 0.2 0.25
nui 0.03 0.07 0.03 0.02 0.02
liit -0.02 -0.04 -0.09 -0.06 -0.11
which 0.27 0.28 0.42 0.32 0.4
we 0.28 0.28 0.46 0.32 0.39
noodles 0.37 0.23 0.12 0.13 0.13
course 0.28 0.26 0.39 0.27 0.34
125
i 0.27 0.26 0.41 0.35 0.34
liveliness 0.07 0.05 0.01 0.0 0.09
which 0.27 0.28 0.42 0.32 0.4
cost 0.21 0.21 0.35 0.23 0.32
it 0.28 0.26 0.43 0.34 0.39
way 0.26 0.25 0.41 0.33 0.38
126
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
we 0.28 0.28 0.46 0.32 0.39
pot 0.28 0.21 0.19 0.16 0.23
which 0.27 0.28 0.42 0.32 0.4
ravioli 0.31 0.13 0.04 0.02 0.04
kachapacha 0 0 0 0 0
prawns 0.33 0.16 0.05 0.03 0.05
it 0.28 0.26 0.43 0.34 0.39
onions 0.33 0.16 0.12 0.08 0.09
rice 0.37 0.22 0.23 0.2 0.21
options 0.25 0.23 0.34 0.19 0.38
experience 0.25 0.25 0.41 0.3 0.4
127
music 0.2 0.25 0.31 0.5 0.34
128
comfortable 0.18 0.21 0.31 0.21 0.43
pro 0.14 0.13 0.29 0.29 0.23
time 0.25 0.24 0.41 0.31 0.36
delicacies 0.37 0.29 0.1 0.07 0.16
129
options 0.25 0.23 0.34 0.19 0.38
time 0.25 0.24 0.41 0.31 0.36
place 0.3 0.27 0.4 0.3 0.43
it 0.28 0.26 0.43 0.34 0.39
130

```
chowk 0.05 0.07 0.02 -0.02 0.06
you 0.26 0.26 0.43 0.35 0.38
sunset 0.19 0.27 0.19 0.2 0.33
side 0.24 0.23 0.34 0.27 0.41
guava 0.18 0.16 0.02 -0.01 0.03
drink 0.34 0.54 0.29 0.25 0.27
paneer 0.27 0.07 -0.01 -0.03 0.02
it 0.28 0.26 0.43 0.34 0.39
dip 0.31 0.23 0.19 0.17 0.22
combination 0.26 0.24 0.29 0.25 0.36
poppers 0.15 0.14 0.04 0.06 0.07
combination 0.26 0.24 0.29 0.25 0.36
nachos 0.32 0.27 0.08 0.03 0.06
beans 0.33 0.23 0.18 0.17 0.16
salsa 0.31 0.23 0.16 0.3 0.15
jalepenos 0.05 -0.01 -0.07 -0.08 -0.09
vegies 0.2 0.1 -0.05 -0.09 -0.05
lover 0.23 0.21 0.23 0.3 0.23
i 0.27 0.26 0.41 0.35 0.34
fan 0.2 0.18 0.31 0.3 0.31
this 0.24 0.21 0.38 0.3 0.38
time 0.25 0.24 0.41 0.31 0.36
chicken 0.46 0.27 0.22 0.18 0.22
mains 0.19 0.18 0.13 0.1 0.16
pieces 0.29 0.21 0.25 0.25 0.36
lazizi 0 0 0 0 0
garnishing 0.14 0.1 0.03 -0.03 0.04
paan 0.09 0.0 -0.04 -0.04 -0.06
jamun 0.11 0.04 -0.06 -0.02 -0.05
treat 0.27 0.27 0.31 0.21 0.27
place 0.3 0.27 0.4 0.3 0.43
131
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
i 0.27 0.26 0.41 0.35 0.34
132
everything 0.28 0.26 0.4 0.33 0.4
photobooth 0.02 0.1 0.07 0.07 0.09
places 0.27 0.29 0.36 0.27 0.38
133
place 0.3 0.27 0.4 0.3 0.43
ones 0.28 0.27 0.38 0.28 0.33
this 0.24 0.21 0.38 0.3 0.38
rooftops 0.05 0.14 0.08 0.03 0.16
value 0.21 0.2 0.34 0.23 0.33
134
place 0.3 0.27 0.4 0.3 0.43
135
couple 0.25 0.27 0.39 0.3 0.32
music 0.2 0.25 0.31 0.5 0.34
lady 0.19 0.17 0.31 0.3 0.27
power 0.17 0.17 0.31 0.26 0.3
136
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
options 0.25 0.23 0.34 0.19 0.38
137
gold 0.16 0.16 0.26 0.25 0.28
bottles 0.22 0.38 0.2 0.13 0.24
this 0.24 0.21 0.38 0.3 0.38
reason 0.22 0.22 0.41 0.32 0.31
vodka 0.26 0.43 0.15 0.18 0.14
that 0.25 0.27 0.44 0.34 0.39
platter 0.35 0.21 0.16 0.11 0.22
138
place 0.3 0.27 0.4 0.3 0.43
that 0.25 0.27 0.44 0.34 0.39
veg 0.34 0.21 0.13 0.06 0.15
platter 0.35 0.21 0.16 0.11 0.22
music 0.2 0.25 0.31 0.5 0.34
something 0.27 0.26 0.39 0.33 0.35
139
```

options 0.25 0.23 0.34 0.19 0.38
place 0.3 0.27 0.4 0.3 0.43
140
place 0.3 0.27 0.4 0.3 0.43
wheat 0.26 0.21 0.16 0.14 0.16
141
this 0.24 0.21 0.38 0.3 0.38
drink 0.34 0.54 0.29 0.25 0.27
govind 0.01 -0.01 0.02 0.02 0.01
evening 0.27 0.33 0.33 0.3 0.36
142
this 0.24 0.21 0.38 0.3 0.38
outlets 0.18 0.2 0.24 0.15 0.24
menu 0.36 0.3 0.3 0.2 0.33
variety 0.31 0.32 0.36 0.27 0.38
numbers 0.16 0.16 0.32 0.22 0.25
retro 0.18 0.21 0.18 0.24 0.35
143
quality 0.24 0.21 0.37 0.28 0.41
dishes 0.46 0.35 0.24 0.18 0.31
144
place 0.3 0.27 0.4 0.3 0.43
time 0.25 0.24 0.41 0.31 0.36
everyday 0.25 0.24 0.3 0.26 0.32
experience 0.25 0.25 0.41 0.3 0.4
mouth 0.23 0.22 0.23 0.2 0.24
you 0.26 0.26 0.43 0.35 0.38
rabbit 0.2 0.14 0.18 0.17 0.21
them 0.27 0.26 0.42 0.3 0.35
attitude 0.16 0.16 0.31 0.23 0.28
i 0.27 0.26 0.41 0.35 0.34
alice 0.13 0.16 0.19 0.24 0.2
burrow 0.03 -0.0 0.05 0.04 0.1
145
place 0.3 0.27 0.4 0.3 0.43
gold 0.16 0.16 0.26 0.25 0.28
146
order 0.26 0.21 0.35 0.24 0.33
dish 0.4 0.24 0.25 0.21 0.28
147
i 0.27 0.26 0.41 0.35 0.34
veggie 0.37 0.22 0.13 0.09 0.14
meals 0.4 0.36 0.28 0.15 0.26
britani -0.08 -0.08 -0.08 -0.09 -0.05
atmosphere 0.23 0.28 0.29 0.22 0.44
148
baner -0.05 -0.02 -0.05 -0.03 -0.02
road 0.19 0.17 0.27 0.22 0.29
this 0.24 0.21 0.38 0.3 0.38
hippie 0.1 0.11 0.08 0.2 0.17
tables 0.19 0.26 0.29 0.17 0.36
seating 0.17 0.23 0.23 0.12 0.42
place 0.3 0.27 0.4 0.3 0.43
option 0.22 0.19 0.34 0.21 0.35
speed 0.16 0.16 0.29 0.24 0.26
kalmi 0 0 0 0 0
corn 0.28 0.21 0.17 0.15 0.19
nanza -0.09 -0.09 -0.14 -0.15 -0.12
murgh 0.13 -0.02 -0.07 -0.08 -0.03
dish 0.4 0.24 0.25 0.21 0.28
drumsticks 0.17 0.05 -0.02 0.08 0.02
spices 0.37 0.25 0.12 0.11 0.19
yogurt 0.36 0.27 0.14 0.11 0.13
onions 0.33 0.16 0.12 0.08 0.09
chicken 0.46 0.27 0.22 0.18 0.22
it 0.28 0.26 0.43 0.34 0.39
balance 0.2 0.2 0.29 0.21 0.31
flavor 0.39 0.31 0.19 0.19 0.26
spot 0.23 0.23 0.33 0.28 0.38
corn 0.28 0.21 0.17 0.15 0.19
flavors 0.39 0.34 0.17 0.15 0.23
which 0.27 0.28 0.42 0.32 0.4

bit 0.25 0.2 0.33 0.3 0.32
heat 0.23 0.2 0.25 0.2 0.3
burst 0.15 0.2 0.18 0.18 0.22
base 0.22 0.18 0.3 0.22 0.36
tasting 0.36 0.41 0.23 0.18 0.24
paneer 0.27 0.07 -0.01 -0.03 0.02
nothing 0.26 0.26 0.39 0.33 0.36
meat 0.41 0.25 0.23 0.17 0.21
thing 0.26 0.24 0.41 0.35 0.35
skewer 0.22 0.06 0.04 -0.02 0.01
introduction 0.15 0.17 0.3 0.23 0.27
rice 0.37 0.22 0.23 0.2 0.21
amount 0.22 0.21 0.32 0.21 0.29
restaurants 0.36 0.34 0.3 0.19 0.31
favorites 0.29 0.25 0.26 0.26 0.31
vibe 0.16 0.18 0.19 0.32 0.33
149
foodies 0.19 0.17 0.13 0.09 0.09
ball 0.18 0.19 0.27 0.28 0.28
place 0.3 0.27 0.4 0.3 0.43
night 0.27 0.33 0.35 0.35 0.36
150
behavior 0.11 0.11 0.28 0.16 0.21
music 0.2 0.25 0.31 0.5 0.34
cafe 0.33 0.33 0.29 0.24 0.34
bar 0.28 0.39 0.35 0.28 0.4
151
starter 0.23 0.17 0.23 0.19 0.2
played 0.18 0.17 0.3 0.35 0.23
place 0.3 0.27 0.4 0.3 0.43
152
space 0.17 0.23 0.33 0.26 0.44
name 0.21 0.19 0.37 0.32 0.3
group 0.15 0.2 0.38 0.29 0.3
time 0.25 0.24 0.41 0.31 0.36
153
class 0.19 0.21 0.34 0.29 0.3
people 0.24 0.27 0.43 0.31 0.36
music 0.2 0.25 0.31 0.5 0.34
u 0.18 0.17 0.3 0.34 0.24
154
music 0.2 0.25 0.31 0.5 0.34
hospitality 0.23 0.26 0.33 0.17 0.34
ones 0.28 0.27 0.38 0.28 0.33
155
monsoon 0.11 0.12 0.09 0.13 0.18
menu 0.36 0.3 0.3 0.2 0.33
experience 0.25 0.25 0.41 0.3 0.4
156
we 0.28 0.28 0.46 0.32 0.39
it 0.28 0.26 0.43 0.34 0.39
music 0.2 0.25 0.31 0.5 0.34
menu 0.36 0.3 0.3 0.2 0.33
hours 0.21 0.22 0.33 0.25 0.29
deals 0.18 0.2 0.27 0.24 0.27
157
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
matches 0.16 0.15 0.26 0.25 0.28
band 0.16 0.17 0.28 0.4 0.25
158
host 0.19 0.23 0.42 0.3 0.25
fish 0.33 0.24 0.27 0.19 0.28
159
bar 0.28 0.39 0.35 0.28 0.4
burrow 0.03 -0.0 0.05 0.04 0.1
bar 0.28 0.39 0.35 0.28 0.4
restaurant 0.39 0.36 0.37 0.22 0.39
interiors 0.15 0.18 0.2 0.11 0.48
music 0.2 0.25 0.31 0.5 0.34
some 0.29 0.3 0.42 0.33 0.39
onion 0.32 0.16 0.16 0.12 0.13

walnut 0.21 0.16 0.14 0.12 0.28
dishes 0.46 0.35 0.24 0.18 0.31
pasta 0.43 0.28 0.17 0.14 0.17
onions 0.33 0.16 0.12 0.08 0.09
shaves 0.07 0.05 0.06 0.03 0.02
rabbit 0.2 0.14 0.18 0.17 0.21
chicken 0.46 0.27 0.22 0.18 0.22
bread 0.42 0.28 0.23 0.19 0.21
you 0.26 0.26 0.43 0.35 0.38
dish 0.4 0.24 0.25 0.21 0.28
this 0.24 0.21 0.38 0.3 0.38
delicacy 0.29 0.13 0.08 0.05 0.15
pieces 0.29 0.21 0.25 0.25 0.36
it 0.28 0.26 0.43 0.34 0.39
which 0.27 0.28 0.42 0.32 0.4
rum 0.24 0.36 0.14 0.17 0.12
affogato 0.05 0.07 -0.07 -0.04 -0.04
pricing 0.13 0.15 0.28 0.16 0.29
rush 0.17 0.16 0.26 0.24 0.24
concern 0.13 0.1 0.33 0.16 0.25
160
they 0.27 0.28 0.45 0.31 0.36
rooftop 0.13 0.27 0.2 0.13 0.3
all 0.26 0.28 0.43 0.33 0.41
they 0.27 0.28 0.45 0.31 0.36
we 0.28 0.28 0.46 0.32 0.39
weather 0.15 0.19 0.24 0.19 0.28
161
i 0.27 0.26 0.41 0.35 0.34
time 0.25 0.24 0.41 0.31 0.36
mood 0.2 0.22 0.22 0.26 0.34
162
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
i 0.27 0.26 0.41 0.35 0.34
options 0.25 0.23 0.34 0.19 0.38
we 0.28 0.28 0.46 0.32 0.39
summer 0.24 0.28 0.32 0.3 0.33
you 0.26 0.26 0.43 0.35 0.38
course 0.28 0.26 0.39 0.27 0.34
163
leaflet 0.06 0.09 0.13 0.02 0.13
billing 0.06 0.06 0.29 0.11 0.14
experience 0.25 0.25 0.41 0.3 0.4
place 0.3 0.27 0.4 0.3 0.43
hour 0.24 0.27 0.34 0.28 0.29
case 0.18 0.17 0.32 0.24 0.3
quality 0.24 0.21 0.37 0.28 0.41
hours 0.21 0.22 0.33 0.25 0.29
ones 0.28 0.27 0.38 0.28 0.33
164
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
rolls 0.3 0.25 0.22 0.2 0.22
experience 0.25 0.25 0.41 0.3 0.4
165
it 0.28 0.26 0.43 0.34 0.39
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
166
nachos 0.32 0.27 0.08 0.03 0.06
167
we 0.28 0.28 0.46 0.32 0.39
lots 0.28 0.33 0.37 0.28 0.4
everything 0.28 0.26 0.4 0.33 0.4
it 0.28 0.26 0.43 0.34 0.39
rooftop 0.13 0.27 0.2 0.13 0.3
nice 0.29 0.29 0.39 0.32 0.44
opinion 0.18 0.16 0.32 0.24 0.27
something 0.27 0.26 0.39 0.33 0.35
vegetables 0.39 0.26 0.17 0.11 0.21
cheese 0.43 0.3 0.22 0.19 0.21

amount 0.22 0.21 0.32 0.21 0.29
sizler 0 0 0 0 0
chutney 0.31 0.16 0.06 0.08 0.06
salad 0.45 0.32 0.2 0.15 0.23
ana 0.1 0.11 0.2 0.22 0.19
place 0.3 0.27 0.4 0.3 0.43
they 0.27 0.28 0.45 0.31 0.36
policies 0.1 0.11 0.31 0.1 0.23
naan 0.27 0.08 0.02 0.04 0.04
manager 0.13 0.15 0.43 0.23 0.25
all 0.26 0.28 0.43 0.33 0.41
168
place 0.3 0.27 0.4 0.3 0.43
pints 0.22 0.32 0.11 0.07 0.06
chicken 0.46 0.27 0.22 0.18 0.22
potatoes 0.36 0.2 0.16 0.12 0.15
enjoyi -0.06 -0.03 -0.13 -0.09 -0.07
something 0.27 0.26 0.39 0.33 0.35
trucks 0.14 0.17 0.23 0.14 0.23
dips 0.24 0.23 0.11 0.12 0.12
side 0.24 0.23 0.34 0.27 0.41
this 0.24 0.21 0.38 0.3 0.38
it 0.28 0.26 0.43 0.34 0.39
paneer 0.27 0.07 -0.01 -0.03 0.02
anything 0.27 0.25 0.37 0.31 0.33
169
place 0.3 0.27 0.4 0.3 0.43
corn 0.28 0.21 0.17 0.15 0.19
they 0.27 0.28 0.45 0.31 0.36
it 0.28 0.26 0.43 0.34 0.39
music 0.2 0.25 0.31 0.5 0.34
option 0.22 0.19 0.34 0.21 0.35
course 0.28 0.26 0.39 0.27 0.34
that 0.25 0.27 0.44 0.34 0.39
issue 0.15 0.15 0.33 0.21 0.27
dragon 0.17 0.16 0.22 0.22 0.24
directions 0.23 0.2 0.27 0.19 0.3
170
place 0.3 0.27 0.4 0.3 0.43
sone 0.02 0.03 0.01 0.04 -0.03
music 0.2 0.25 0.31 0.5 0.34
singer 0.13 0.15 0.25 0.4 0.19
numbers 0.16 0.16 0.32 0.22 0.25
options 0.25 0.23 0.34 0.19 0.38
cheese 0.43 0.3 0.22 0.19 0.21
rolls 0.3 0.25 0.22 0.2 0.22
171
paneer 0.27 0.07 -0.01 -0.03 0.02
beers 0.27 0.47 0.22 0.18 0.18
172
place 0.3 0.27 0.4 0.3 0.43
raj 0.09 0.08 0.17 0.16 0.12
173
part 0.23 0.21 0.38 0.33 0.36
guidance 0.1 0.12 0.31 0.18 0.24
security 0.11 0.13 0.36 0.17 0.3
car 0.16 0.18 0.31 0.23 0.35
park 0.17 0.22 0.29 0.23 0.33
place 0.3 0.27 0.4 0.3 0.43
hangout 0.1 0.18 0.16 0.16 0.14
crowd 0.19 0.26 0.32 0.3 0.29
side 0.24 0.23 0.34 0.27 0.41
music 0.2 0.25 0.31 0.5 0.34
range 0.23 0.26 0.31 0.25 0.38
one 0.27 0.28 0.44 0.36 0.39
salsa 0.31 0.23 0.16 0.3 0.15
chicken 0.46 0.27 0.22 0.18 0.22
style 0.3 0.26 0.32 0.35 0.48
chicken 0.46 0.27 0.22 0.18 0.22
lollipop 0.15 0.12 0.06 0.13 0.1
we 0.28 0.28 0.46 0.32 0.39
prawns 0.33 0.16 0.05 0.03 0.05

size 0.21 0.15 0.27 0.22 0.37
dumplings 0.36 0.2 0.06 0.02 0.07
course 0.28 0.26 0.39 0.27 0.34
it 0.28 0.26 0.43 0.34 0.39
prawn 0.32 0.17 0.06 0.04 0.04
rice 0.37 0.22 0.23 0.2 0.21
menu 0.36 0.3 0.3 0.2 0.33
jamuns 0.04 0.02 -0.09 -0.12 -0.08
which 0.27 0.28 0.42 0.32 0.4
accompaniment 0.22 0.14 0.12 0.23 0.19
it 0.28 0.26 0.43 0.34 0.39
options 0.25 0.23 0.34 0.19 0.38
kind 0.27 0.25 0.41 0.33 0.39
music 0.2 0.25 0.31 0.5 0.34
174
place 0.3 0.27 0.4 0.3 0.43
it 0.28 0.26 0.43 0.34 0.39
that 0.25 0.27 0.44 0.34 0.39
hands 0.19 0.22 0.34 0.3 0.29
rabbit 0.2 0.14 0.18 0.17 0.21
i 0.27 0.26 0.41 0.35 0.34
warmth 0.16 0.16 0.17 0.13 0.33
hospitality 0.23 0.26 0.33 0.17 0.34
work 0.22 0.22 0.43 0.34 0.38
ashwin 0.08 0.02 0.04 0.04 0.03
175
fun 0.26 0.31 0.34 0.32 0.37
vibe 0.16 0.18 0.19 0.32 0.33
mention 0.25 0.25 0.39 0.29 0.3
he 0.23 0.23 0.4 0.35 0.32
place 0.3 0.27 0.4 0.3 0.43
176
highball 0.09 0.2 0.0 0.03 0.02
place 0.3 0.27 0.4 0.3 0.43
level 0.16 0.19 0.35 0.29 0.38
it 0.28 0.26 0.43 0.34 0.39
things 0.26 0.26 0.4 0.3 0.36
cider 0.26 0.3 0.13 0.09 0.1
stout 0.18 0.27 0.15 0.12 0.12
this 0.24 0.21 0.38 0.3 0.38
drink 0.34 0.54 0.29 0.25 0.27
favorites 0.29 0.25 0.26 0.26 0.31
piece 0.25 0.18 0.28 0.29 0.38
you 0.26 0.26 0.43 0.35 0.38
chicken 0.46 0.27 0.22 0.18 0.22
mutton 0.28 0.12 0.06 0.02 0.04
summary 0.14 0.14 0.31 0.18 0.27
hangout 0.1 0.18 0.16 0.16 0.14
177
delicious 0.46 0.39 0.26 0.19 0.28
placed 0.19 0.2 0.33 0.2 0.34
place 0.3 0.27 0.4 0.3 0.43
outside 0.24 0.26 0.38 0.28 0.43
which 0.27 0.28 0.42 0.32 0.4
all 0.26 0.28 0.43 0.33 0.41
quality 0.24 0.21 0.37 0.28 0.41
level 0.16 0.19 0.35 0.29 0.38
rating 0.17 0.15 0.24 0.21 0.29
178
hangout 0.1 0.18 0.16 0.16 0.14
party 0.24 0.31 0.34 0.34 0.34
place 0.3 0.27 0.4 0.3 0.43
crowd 0.19 0.26 0.32 0.3 0.29
worth 0.23 0.24 0.33 0.27 0.3
179
reviews 0.21 0.21 0.29 0.23 0.32
place 0.3 0.27 0.4 0.3 0.43
waste 0.15 0.16 0.28 0.16 0.23
180
rahul 0.05 0.02 0.13 0.13 0.1
guy 0.21 0.23 0.37 0.35 0.27
place 0.3 0.27 0.4 0.3 0.43

```
181
usher 0.08 0.14 0.19 0.27 0.14
what 0.25 0.24 0.42 0.34 0.37
music 0.2 0.25 0.31 0.5 0.34
182
place 0.3 0.27 0.4 0.3 0.43
we 0.28 0.28 0.46 0.32 0.39
atmosphere 0.23 0.28 0.29 0.22 0.44
music 0.2 0.25 0.31 0.5 0.34
is 0.25 0.24 0.41 0.33 0.37
183
place 0.3 0.27 0.4 0.3 0.43
184
highland 0.14 0.17 0.21 0.2 0.25
place 0.3 0.27 0.4 0.3 0.43
hour 0.24 0.27 0.34 0.28 0.29
hangout 0.1 0.18 0.16 0.16 0.14
185
place 0.3 0.27 0.4 0.3 0.43
music 0.2 0.25 0.31 0.5 0.34
disappointing 0.19 0.15 0.25 0.15 0.19
suey 0.13 0.02 -0.02 0.06 0.01
things 0.26 0.26 0.4 0.3 0.36
chicken 0.46 0.27 0.22 0.18 0.22
ghee 0.19 0.04 0.02 0.02 0.02
```

In [0]:

```
#'wrong'.lower() in pos+neg
#after_pred_te.iloc[83:84,1].apply(lambda x :print(x))
final_df=pd.DataFrame(final_list,columns=['food','alcohol','service','music','ambience'])
```

In [0]:

```
after_pred_te.groupby(['res_name'])['res_name'].agg('count')
```

Out[0]:

```
res_name
highland                112
mr-rabbits-bar-burrow    35
your-ale-house           39
Name: res_name, dtype: int64
```

In [0]:

```
after_pred_te.head()
```

Out[0]:

	res_name	comment	predicted
0	your-ale-house	Rahul Rahagdale is Mr. waiter helped us with...	(ambience, food, misc, service)
1	highland	Awesome drinks and Food . Great Veg Starters a...	(alcohol, ambience, food, service)
2	your-ale-house	The waiter tells you something else about the ...	(alcohol, ambience, food, service)
3	highland	One of the finest in Balewadi/Baner area. Grea...	(alcohol, food, misc, service)
4	your-ale-house	I visited this place for an office party. We r...	(ambience, food, misc, service)

In [0]:

```
l=[]
for i in range(0,after_pred_te.shape[0]):
    take=after_pred_te.iloc[i,2]
    search=final_df.iloc[i,].to_dict()
    take=list(take)
    #print(take)
    #print(search)
    d={}
```

```
    for j in take:
        if(j  in search.keys() and j!='misc'):
            d[j]=search[j]
    l.append(d)
```

In [0]:

```
temp_df=pd.DataFrame(l)
temp_df.fillna(999,inplace=True)
```

In [0]:

```
done=pd.concat([after_pred_te,temp_df],axis=1)
```

In [0]:

```
count=0

amb_count_high=112- pd.value_counts(done.iloc[done[done['res_name']=='highland'].index.tolist(),3])[999.0]
amb_count_yah=35- pd.value_counts(done.iloc[done[done['res_name']=='your-ale-house'].index.tolist(),3])[999.0]
amb_count_rab= 39-pd.value_counts(done.iloc[done[done['res_name']=='mr-rabbits-bar-burrow'].index.tolist(),3])[999.0]

food_count_high=112- pd.value_counts(done.iloc[done[done['res_name']=='highland'].index.tolist(),4])[999.0]
food_count_yah=35- pd.value_counts(done.iloc[done[done['res_name']=='your-ale-house'].index.tolist(),4])[999.0]
food_count_rab= 39#-pd.value_counts(done.iloc[done[done['res_name']=='mr-rabbits-bar-burrow'].index.tolist(),4])[999.0]

service_count_high=112- pd.value_counts(done.iloc[done[done['res_name']=='highland'].index.tolist(),5])[999.0]
service_count_yah=35- pd.value_counts(done.iloc[done[done['res_name']=='your-ale-house'].index.tolist(),5])[999.0]
service_count_rab= 39-pd.value_counts(done.iloc[done[done['res_name']=='mr-rabbits-bar-burrow'].index.tolist(),5])[999.0]

alcohol_count_high=112- pd.value_counts(done.iloc[done[done['res_name']=='highland'].index.tolist(),6])[999.0]
alcohol_count_yah=35- pd.value_counts(done.iloc[done[done['res_name']=='your-ale-house'].index.tolist(),6])[999.0]
alcohol_count_rab= 39-pd.value_counts(done.iloc[done[done['res_name']=='mr-rabbits-bar-burrow'].index.tolist(),6])[999.0]

music_count_high=112- pd.value_counts(done.iloc[done[done['res_name']=='highland'].index.tolist(),7])[999.0]
music_count_yah=35- pd.value_counts(done.iloc[done[done['res_name']=='your-ale-house'].index.tolist(),7])[999.0]
music_count_rab= 39-pd.value_counts(done.iloc[done[done['res_name']=='mr-rabbits-bar-burrow'].index.tolist(),7])[999.0]




#done.iloc[[done],3].apply(lambda x:print(x) if(x!=999) else None)
```

In [0]:

```
done.replace(999.0,0,inplace=True)
```

In [0]:

```
#food_count_high 107
#food_count_yah 33
#58.5/107 .5467
#29/33 .8787
#8.5/39
```

In [0]:

```
m=[music_count_high,music_count_rab,music_count_yah]
a=[alcohol_count_high,alcohol_count_rab,alcohol_count_yah]
amb=[amb_count_high,amb_count_rab,amb_count_yah]
f=[112,food_count_rab,food_count_yah]#manual
s=[service_count_high,service_count_rab,service_count_yah]

zzz=pd.DataFrame([a,amb,f,s,m]).T
zzz.columns=['alcohol','ambience','food','service','music']
zzz.index=['highland','mr-rabbits-bar-burrow','your-ale-house']
zzz
```

Out[0]:

alcohol ambience food service music

	alcohol	ambience	food	service	music
highland	42	69	112	69	19
mr-rabbits-bar-burrow	15	31	39	29	13
your-ale-house	14	30	33	27	5

In [0]:

```
zzz1=done.groupby(['res_name']).agg({'alcohol':'sum','ambience':'sum','food':'sum','service':'sum','music':'sum'})
zzz1
```

Out[0]:

	alcohol	ambience	food	service	music
res_name					
highland	26.5	42.5	54.5	47.5	2.0
mr-rabbits-bar-burrow	0.5	10.0	8.5	23.0	2.5
your-ale-house	11.0	13.0	25.0	-2.0	2.0

In [0]:

```
#drawing radar plots to compare
```

In [0]:

```
zzz2=round(zzz1/zzz,2)
zzz2
```

Out[0]:

	alcohol	ambience	food	service	music
res_name					
highland	0.63	0.62	0.49	0.69	0.11
mr-rabbits-bar-burrow	0.03	0.32	0.22	0.79	0.19
your-ale-house	0.79	0.43	0.76	-0.07	0.40

In [0]:

```
labels=np.array(['alcohol','ambience','food','service','music'])
stats=zzz2.iloc[2,:].values
stats1=zzz2.iloc[0,:].values
```

In [0]:

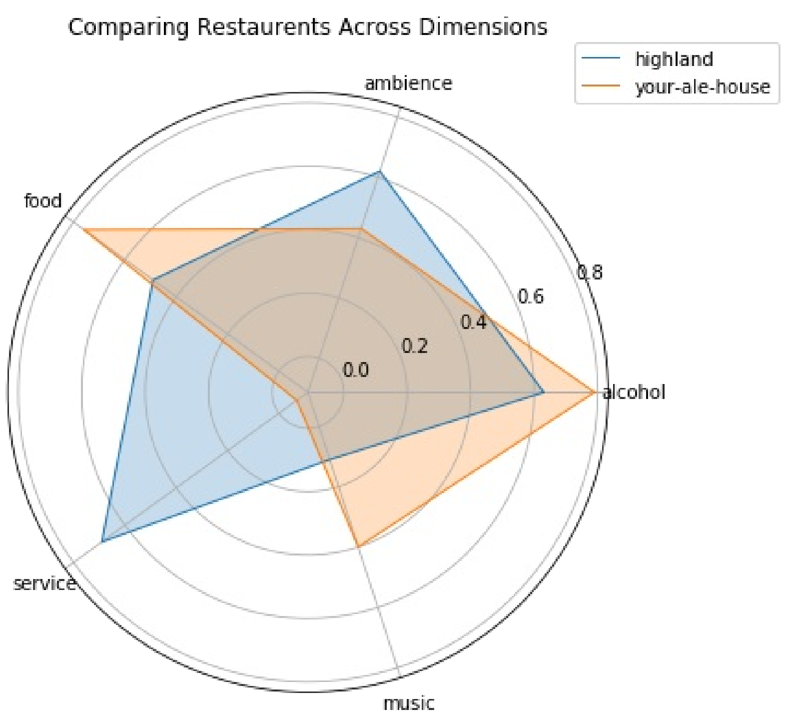
```
# Number of variables we're plotting.
num_vars = len(labels)

# Split the circle into even parts and save the angles
# so we know where to put each axis.
angles = np.linspace(0, 2 * np.pi, num_vars, endpoint=False).tolist()

# The plot is a circle, so we need to "complete the loop"
# and append the start value to the end.
angles += angles[:1]

# ax = plt.subplot(polar=True)
fig, ax = plt.subplots(figsize=(6, 6), subplot_kw=dict(polar=True))

# Helper function to plot each car on the radar chart.
def add_to_radar(car_model, color):
    values = zzz2.loc[car_model].tolist()
    values += values[:1]
    ax.plot(angles, values, linewidth=1, label=car_model)
    ax.fill(angles, values, alpha=0.25)
```

```
    print(values)
add_to_radar('highland', '#1aaf6c')
add_to_radar('your-ale-house', '#429bf4')


# Add title.
ax.set_title('Comparing Restaurents Across Dimensions', y=1.08)
ax.set_thetagrids(np.degrees(angles), labels)

# Add a legend as well.
ax.legend(loc='upper right', bbox_to_anchor=(1.3, 1.1))
```

```
[0.63, 0.62, 0.49, 0.69, 0.11, 0.63]
[0.79, 0.43, 0.76, -0.07, 0.4, 0.79]
```

Out[0]:

```
<matplotlib.legend.Legend at 0x15e5a0ee898>
```



In [0]: