Restaurant Project

Jeffrey McGovern

October 18, 2016

Introduction

Restaurants have been a favorite area of NLP researchers for years. There are many corpora of restaurant data available. We have created our own corpora by individually visiting restaurants and creating our own corpus in a semi-structured way.

Various algorithms and tools of the Natural Language Toolkit (NLTK) help classify text by either using predetermined testing and training sets, or else by splitting up the dataset randomly into a testing and training set. Usually, the test set is referred to as a proportion of the whole dataset.

After the extraction of features from both the testing and training sets, the Naïe Bayes Classifier built into NLTK trains on the features of the training set and then classifies the test set accordingly. Explained below are the careful selection of the features each exercise employed in order to maximize classification accuracy.

Features tend to fall under a few categories applied to certain pieces of the dataset. For text, we usually extract sentiment and n-gram counts from the relevant and appropriate sections. Some exercises use the NLTK Vader module[1] to determine sentiment using the `polarity_scores` function. The function returns the positive, negative, and neutral valence of the text as a dictionary. The NLTK `ngrams` function extracts n-grams from the text that the NLTK function `word_tokenize` tokenizes. The `FreqDist` object aids the calculation of n-gram counts and we explore various modifications of the counts below.

The class dataset is a collection of 91 reviews by 31 authors from the Fall 2016 CPE-582 class. Authors were tasked with going to a restaurant they have never been to before and rating the restaurant on a 1-5 scale in terms of Food, Venue, Service, and Overall, providing a paragraph to explain each of the ratings. While many of the reviews fit the format, the specification did not align with the example given, and nearly half of the reviews do not follow the required specification.

[1] `nltk.sentiment.vader`

1 Binary Paragraph Rating

*Predict the **binary** rating of each paragraph regardless of subject. Assume correct order for ratings.*

1.1 Methods

Two methods attempt to correctly classify the binary rating of each paragraph:

- Sentiment of the paragraph
- N-Grams from 1 to 5 of the paragraph

For n-gram extraction, punctuation is ignored and all words are considered and converted to lowercase. For sentiment, all three valences are considered. Two datasets are of notable distinction for this exercise: the `data_example` predetermined set and the class's entire set of restaurant reviews split in half at random into testing and training sets.

1.2 Results

For the predetermined dataset, the *sentiment of the paragraph* is the feature that provides the highest classification accuracy, providing 0.688 accuracy, notably better than randomly picking a binary rating for the paragraph. Using n-grams as this feature results in precisely 0.500 accuracy on the predetermined dataset, exactly as good as random.

The class's dataset has somewhat different results. With just sentiment as a feature, the 5 run average for the accuracy was 0.513, going as low as 0.494 and as high as 0.539. Using n-grams garnered 0.494 accuracy for a 5 run average, dipped as low as 0.478 and went as high as 0.511, roughly equivalent to using sentiment. On separate runs of 5, the average accuracy of the n-gram model frequently went below 0.500, while the sentiment model almost exclusively stayed above it. Neither of these results are much better than random chance of picking the binary classifier. Discussion on why this occurred can be found in Exercise 2.

2 Using Review Sentiment for Overall Good or Bad

Use NLTK functions and corpora to discover three interesting phenomena about the restaurant corpus. Use machine learning to prove this. Discuss your results.

Exercise 1 wants us to assume that the paragraphs are in the correct order for ratings. In the discussion of the assignment, two students determined the following[2] by subjectively tagging paragraphs:

> 49 / 91 [53.85%] reviews have the correct order of paragraphs, Food, Service, Venue, Overall
>
> 30 / 91 [32.97%] reviews have either the wrong order of paragraphs or have more than one paragraph on the same topic. It was not uncommon to see a review which had two paragraphs talking almost entirely about the food neglecting one of the topics all together.
>
> 7 / 91 [7.96%] reviews had one or more of their paragraphs labelled as "Other" by us. This means we couldn't figure out what the intended topic for this paragraph was at all. For example if a paragraph talked about the food, service, and venue in the same paragraph but was not making an overall statement about the restaurant it would be labelled "Other".
>
> ...
>
> 3 / 91 [3.30%] reviews had the incorrect number of paragraphs so their order was innately wrong.

When barely half of the reviews have the correct distribution of paragraphs, it makes sense that the results in Exercise 1 are so poor.

Instead of determining the rating of an arbitrary paragraph using a score that has a nearly 50/50 chance of actually corresponding to that paragraph, we decided to look into *classifying binary Overall rating using only the paragraph text*, to see if it performed better on our class's dataset than Exercise **??**.

2.1 Methods

Features extracted from the paragraphs as a whole are as follows:

- Sentiment for all paragraphs as a single text
- Sentiment for each paragraph
- N-Grams from 1 to 5 of all paragraphs as a single text

2.2 Results

For the class's dataset, *sentiment for all paragraphs as a single text* performed the best for classifying the review as binary Overall rating. Usually, for a test proportion of 0.50 (where half the data is used for training), the 5 run average accuracy is around 0.55 to 0.65, with the minimum accuracy usually staying above 0.050. This is better than average, but not by much. Using the other two methods garnered similar results, but

[2] `https://polylearn.calpoly.edu/AY_2016-2017/mod/forum/discuss.php?d=7990`

On the supplied example dataset `data_example`, where the test and training are predetermined, none of the classifiers performed well. Each of them garnered about 25% accuracy, significantly worse than random chance. Fortunately, Exercise 1 does perform well on this tiny dataset.

3 Predicting Overall Ratings

Predict the overall rating of each review $(\mathbf{1}-\mathbf{5})$ *considering all information from the review, except for the final rating number.*

Label the data with the original (1-5). Run a classifier of your choice to predict the actual (1-5) rating of each item. Create your own accuracy function that calculates the real-value prediction error as the RMSE (Root Mean Squared Error) distance of 2 vectors (predicted ratings and actual ratings of the test set). Report your test results as "average RMSE error" for each trial. Report an overall average of 5 trials.

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}(\hat{y}_t - y_t)^2}{n}}.$$

3.1 Methods

We experimented with the following methods to classify the Overall rating of the review.

- The review's Food, Venue, and Service ratings
- Sentiment for all paragraphs as a single text
- N-Grams from 1 to 5 of all paragraphs as a single text

The ratings labels given to the Naïe Bayes Classifier were the string representation of the number, which was converted back to an integer in order to calculated the RMSE after classification. The ratings in the features, however, were actually integers.

3.2 Results

Using the predetermined dataset, the sentiment analysis performed best with a RMSE of 0.612. With sentiment, the RMSE was 1.061 and for the n-gram feature model, the RMSE was 1.620.

For the class's dataset, the best performing set of features were the non-Overall ratings with a 50-run average RMSE of 0.694, with a high of 0.907 and a low of 0.471. That means that the prediction stays generally within a little over half a rating. Sentiment analysis took a long time to run, but a 5 run average RMSE was 1.563, with a minimum at 1.247 and a maximum of 1.563.

The n-gram features ran faster than sentiment analysis, but performed worse, with a 10-run average RMSE of 2.001 with a minimum of 1.549 and a maximum of 2.821. That spans nearly half the possible range of review values, making this the least accurate.

Clearly, simply using the other ratings to predict the overall score is fairly reliable. None of this is surprising. In the specifications for the restaurant review assignments, it states, "Your 'overall' assessment shouldn't be radically out of step with your food/service/venue ratings." This very likely encouraged everyone to score their restaurants in such a way that it aligned with the overall rating.

4 Predicting Authors

> *Predict the* ***author*** *of each review.*
>
> *In the case of authorship attribution, the distance error for one document is binary: 1 if correct, 0 if not correct. Report your test results as "average RMSE error" for each trial. Report an overall average of 5 trials.*

4.1 Methods

A significant amount of experimentation was done with the authorship classification. Decision Tree Classifiers were attempted, but produced usually 0 accuracy. Combining different metric types (e.g. n-grams and sentiment) was also an impossibility, as no amount of weighting with respect to the other metric helped and neither did manual weighting. Between sentiment and n-grams (ignoring case and punctuation), the latter prevailed as the consistently more accurate one. Experimentation with removing the most common ngrams from the model also didn't produce reliably better accuracy.

4.2 Results

For the predetermined dataset, the average accuracy was 0.750 and the RMSE was 0.500. On the class's dataset, 5 runs took long enough that it risked going over the time limit. The average accuracy was 0.317, with a min of 0.217 and a max of 0.391. The average RMSE was 0.825, with a min of 0.780 and a max of 0.885.

5 Authorship Confusion Matrix

Listing 1: Classification list to go with the following confusion matrix.

```
Exercise 4: ACTUAL           CLASSIFIED
Exercise 4: Adam Calabrigo   Logan Williams
Exercise 4: Adam Calabrigo   Logan Williams
Exercise 4: Aditya Budhwar   Aditya Budhwar
Exercise 4: Aditya Budhwar   Aditya Budhwar
Exercise 4: Alanna Buss      Alanna Buss
Exercise 4: Alanna Buss      Alanna Buss
Exercise 4: Brandon Cooper   Miguel Aguilar
Exercise 4: Brandon Livitski Logan Williams
Exercise 4: Brandon Livitski Vivian Fong
Exercise 4: Christian Durst  Logan Williams
Exercise 4: Cody Hunt        Joel Dentici
Exercise 4: Cody Hunt        Miguel Aguilar
Exercise 4: Daniel Kauffman  Logan Williams
Exercise 4: Daniel Kauffman  Logan Williams
Exercise 4: Gavin Scott      Logan Williams
Exercise 4: Gavin Scott      Logan Williams
Exercise 4: Ivan Pachev      Ivan Pachev
Exercise 4: Ivan Pachev      Ivan Pachev
Exercise 4: Jeff McGovern    Logan Williams
Exercise 4: Jeff McGovern    Miguel Aguilar
Exercise 4: Jeremy Kerfs     Jeff McGovern
Exercise 4: Jeremy Kerfs     Logan Williams
Exercise 4: Joel Dentici     Logan Williams
Exercise 4: Joey Wilson      Joel Dentici
Exercise 4: Joey Wilson      Joey Wilson
Exercise 4: Jon Doughty      Logan Williams
Exercise 4: Jonathan Sleep   Logan Williams
Exercise 4: Kishan Patel     Logan Williams
Exercise 4: Logan Williams   Logan Williams
Exercise 4: Logan Williams   Logan Williams
Exercise 4: Michael Williams Miguel Aguilar
Exercise 4: Miguel Aguilar   Miguel Aguilar
Exercise 4: Miguel Aguilar   Miguel Aguilar
Exercise 4: Nupur Garg       Logan Williams
Exercise 4: Nupur Garg       Nupur Garg
Exercise 4: Ryan Gelston     Miguel Aguilar
Exercise 4: Ryan Gelston     Ryan Gelston
Exercise 4: Ryan Smith       Logan Williams
Exercise 4: Sage Maxwell     Sage Maxwell
Exercise 4: Sage Maxwell     Sage Maxwell
Exercise 4: Sam Lakes        Brandon Livitski
Exercise 4: Sean Bayley      Logan Williams
Exercise 4: Timothy Chu      Timothy Chu
Exercise 4: Timothy Chu      Timothy Chu
Exercise 4: Vivian Fong      Jeremy Kerfs
Exercise 4: Vivian Fong      Vivian Fong
Exercise 4: 0.391
Exercise 4: Average RMSE Error: 0.780
```

Listing 2: Example classification list and confusion matrix for classifying authorship using n-grams and Naïe Bayes Classifier.

```
Exercise 4: Confusion Matrix:
                 |         B                                 M                  |
                 |         r C   D                           i                  |
                 | A A   B a h   a               J   L c M                      |
                 | d d   r n r   n     J         o   o h i                      |
                 | a i   a d i   i     e J J     n K g a g   R   S              |
                 | m t A n o s   e G I f e o J J a i a e u   y   a   S T V |
                 |   y l d n t   l a v f r e o o t s n l e N a R g   e i i |
                 | C a a o   i C   v a   e l e n h h     l u n y e S a m v |
                 | a   n n L a o K i n M m   y   a a W W   p   a   a n o i |
                 | l B n   i n d a n   c y D   D n n i i A u G n M m   t a |
                 | a u a C v   y u   P G   e W o     l l g r e   a   B h n |
                 | b d   o i D   f S a o K n i u S P l l u   l S x L a y   |
                 | r h B o t u H f c c v e t l g l a i i i G s m w a y   F |
                 | i w u p s r u m o h e r i s h e t a a l a t i e k l C o |
                 | g a s e k s n a t e r f c o t e e m m a r o t l e e h n |
                 | o r s r i t t n t v n s i n y p l s s r g n h l s y u g |
-----------------+-----------------------------------------------------------+
  Adam Calabrigo |<.>. . . . . . . . . . . . . . . . . 2 . . . . . . . . . . |
  Aditya Budhwar | .<2>. . . . . . . . . . . . . . . . . . . . . . . . . . . |
     Alanna Buss | . .<2>. . . . . . . . . . . . . . . . . . . . . . . . . . |
  Brandon Cooper | . . .<.>. . . . . . . . . . . . . . . 1 . . . . . . . . . |
Brandon Livitski | . . . .<.>. . . . . . . . . . . . . . 1 . . . . . . . . . 1 |
 Christian Durst | . . . . .<.>. . . . . . . . . . . . . 1 . . . . . . . . . . |
       Cody Hunt | . . . . . .<.>. . . . . 1 . . . . . . 1 . . . . . . . . . |
 Daniel Kauffman | . . . . . .<.>. . . . . . . . . . . . 2 . . . . . . . . . . |
     Gavin Scott | . . . . . .<.>. . . . . . . . . . . . 2 . . . . . . . . . . |
     Ivan Pachev | . . . . . .<2>. . . . . . . . . . . . . . . . . . . . . . . |
   Jeff McGovern | . . . . . .<.>. . . . . . . . . . . . 1 . 1 . . . . . . . . |
    Jeremy Kerfs | . . . . . 1<.>. . . . . . . . . . . . 1 . . . . . . . . . . |
    Joel Dentici | . . . . . .<.>. . . . . . . . . . . . 1 . . . . . . . . . . |
     Joey Wilson | . . . . . 1<1>. . . . . . . . . . . . . . . . . . . . . . . |
     Jon Doughty | . . . . . .<.>. . . . . . . . . . . . 1 . . . . . . . . . . |
  Jonathan Sleep | . . . . . .<.>. . . . . . . . . . . . 1 . . . . . . . . . . |
    Kishan Patel | . . . . . .<.>1 . . . . . . . . . . . . . . . . . . . . . . |
  Logan Williams | . . . . . .<2>. . . . . . . . . . . . . . . . . . . . . . . |
Michael Williams | . . . . . .<.>1 . . . . . . . . . . . . . . . . . . . . . . |
  Miguel Aguilar | . . . . . .<2>. . . . . . . . . . . . . . . . . . . . . . . |
      Nupur Garg | . . . . . 1 . .<1>. . . . . . . . . . . . . . . . . . . . |
    Ryan Gelston | . . . . . 1 .<1>. . . . . . . . . . . . . . . . . . . . . |
     Ryan Smith | . . . . . 1 . . . .<.>. . . . . . . . . . . . . . . . . . |
    Sage Maxwell | . . . . . .<2>. . . . . . . . . . . . . . . . . . . . . . |
       Sam Lakes | . . . . . .<.>. . . . . . . . . . . . . . . . . . . . . . |
     Sean Bayley | . . . . . 1 . . . . . .<.>. . . . . . . . . . . . . . . . |
     Timothy Chu | . . . . . .<2>. . . . . . . . . . . . . . . . . . . . . . |
     Vivian Fong | . . . . . .<1>. . . . . . . . . . . . . . . . . . . . . . |
-----------------+-----------------------------------------------------------+
(row = reference; col = test)
```

Listing 3: An overall confusion matrix for a run of 5 classifications. It is unclear if this is actually useful or appropriate, since it does not add up to the expected amount in the dataset.

```
Exercise 4: Overall Confusion Matrix:
                 |         B                                 M                       |
                 |         r C   D                           i                       |
                 | A A   B a h   a               J J   L c M                 T   |
                 | d d   r n r   n     J         o u   o h i N               o   |
                 | a i   a d i   i     e J J     n s K g a g i   R   S       b   |
                 | m t A n o s   e G I f e o J J a t i a e u c   y   a   S T i V |
                 |   y l d n t   l a v f r e o o t i s n l e o N a R g   e i a i |
                 | C a a o   i C   v a   e l e n h n h     l l u n y e S a m s v |
                 | a   n n L a o K i n M m   y   a   a W W   e p   a   a n o   i |
                 | l B n   i n d a n   c y D   D n P n i i A   u G n M m   t B a |
                 | a u a C v   y u   P G   e W o   o   l l g M r e   a   B h l n |
                 | b d   o i D   f S a o K n i u S s P l l u a   l S x L a y e   |
                 | r h B o t u H f c c v e t l g l t a i i i r G s m w a y   i F |
                 | i w u p s r u m o h e r i s h e i t a a l t a t i e k l C s o |
                 | g a s e k s n a t e r f c o t e g e m m a i r o t l e e h c n |
                 | o r s r i t t n t v n s i n y p o l s s r n g n h l s y u h g |
-----------------+-----------------------------------------------------------+
  Adam Calabrigo |<3>. . . . . 1 . . . 1 . . . 2 . 1 . . . . . . . . . . . |
  Aditya Budhwar | .<4>. . . . . 1 . . . . . 1 . 1 . |
     Alanna Buss | . .<3>1 . 1 . . . . . 1 . . . 2 . . . . . . . . . . . . |
  Brandon Cooper | . . .<2>. . . . . 1 . 1 . . . 1 . . . 2 . . . . . . . . . . |
Brandon Livitski | . . . .<.>. . 1 . . 1 . . . . . 1 . 2 . . . . . . . . . 2 1 |
 Christian Durst | . . . . .<.>. . . . . 1 . 2 . 1 . . . . . 1 . 3 . |
       Cody Hunt | 1 . . . . .<.>. . . . 1 . . . 4 . . . 3 . . . . . . . . . . |
 Daniel Kauffman | . . . . . .<6>. . 1 . . . . . 2 . . . . . . . . . . . . |
     Gavin Scott | . . . . . .<1>. . . . . 1 . 2 . 4 . 1 . . . . . . . . . . |
     Ivan Pachev | . . . . . 1 .<2>. . . . 1 . . . . . 1 . . . . . . . . . 1 . |
   Jeff McGovern | . . . . . .<3>. . . . . 1 . 1 . . . . . . . 3 . |
    Jeremy Kerfs | . . . . . 2<.>. . . . . 4 . 2 . . . . . . . . 1 . |
    Joel Dentici | . . . . . 1 .<.>. . . . . 1 . . . . . . . . . 2 . |
     Joey Wilson | . . . . . 4<1>. . . . 1 . . . 1 . . . . . 1 1 . |
     Jon Doughty | . . . . . .<2>. . . 2 . 3 . . . . . . . . 1 . |
  Jonathan Sleep | . . . . 1 . . . . . .<1>. . 1 . . . . . . . . . 1 2 . |
  Justin Postigo | . . . . . 1 . . . . . .<5>. . . 2 . . . . . . . . . . . |
    Kishan Patel | . . . . . 1 . . . . . .<.>3 . . . . . . . . . . . . |
  Logan Williams | . . . . . .<4>. . . . . 5 . |
Michael Williams | . . . . . 1 . . . . . 2 . 2<1>1 . . . . . 1 . |
  Miguel Aguilar | . 1 . . . . . 1 . . .<6>. . . . . |
   Nicole Martin | . . . . . 2 . . . . . 1 1<.>1 . . . . . 2 . |
      Nupur Garg | . . . . . 1 . . .<3>. . . . . 3 . |
    Ryan Gelston | . . . . . 1 . 1 . .<3>. . . . . 4 . |
      Ryan Smith | . . . . . 1 . . 1 1 . . . . . 1 . . . 1 .<2>. . . . . 1 |
    Sage Maxwell | . . . . . 1 . . 1 . . . . . 1 . . . . . .<3>. . 1 . . |
       Sam Lakes | . . . . 1 . . . . . 1 . . . . . .<.>. . 1 . |
     Sean Bayley | . . . . . 2 . . . . . .<1>. . . |
     Timothy Chu | . . . . . 1 . . . . . 1 . 1 . . . . . .<6>. . |
  Tobias Bleisch | . . . . . 1 . . . . . .<7>. |
     Vivian Fong | . . . . . 1 . . . . . 4<4>|
-----------------+-----------------------------------------------------------+
(row = reference; col = test)

Exercise 4: Runs: 5 Average     : 0.317 Max: 0.391 Min: 0.217
Exercise 4: Runs: 5 Average RMSE: 0.825 Max: 0.885 Min: 0.780
```