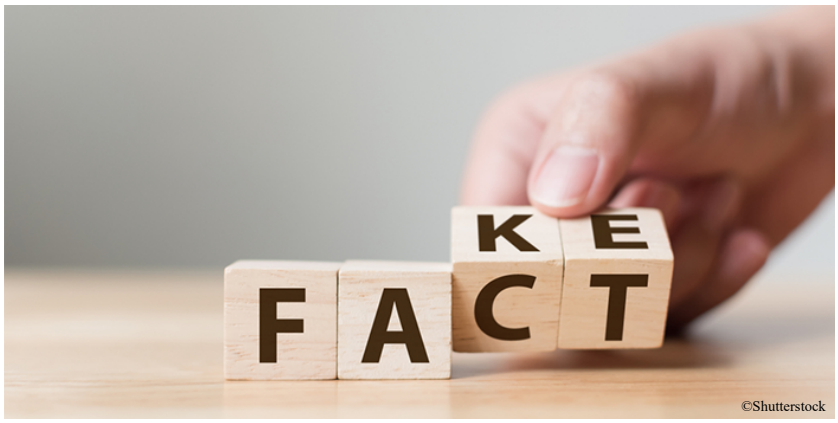
John Fields
Dr. Ami Gates
IST707 - Homework #8
9/5/19



Introduction

Fake product reviews have existed on the internet since the late 1990's when websites like Epinions became popular[i]. This trend has continued and a 2011 New York Times article profiled a marketer who would write two positive reviews for a product/service for $5.[ii] In 2011, these types of reviews were typically manually entered but now it's much easier to use technology to write these reviews. In addition to using machine learning to combat fake reviews, the Federal Trade Commission (FTC) fined a weight-loss company $12.8 million for authorizing the publication of fake reviews.[iii] However, there is some positive news in a 2018 journal paper, *Fake Review Detection Using Classification*. In this paper, Neha Chowdhury and Anala Pandit provide a review of how they used machine learning to detect 98-99% of fake product reviews.[iv]

In addition to predicting fake reviews in the restaurant review data provided, a second goal is to analyze the reviews to predict which are positive and which are negative. This task (called sentiment analysis) is typically much easier than identifying fake reviews and can provide a restaurant owner or retailer with interesting insights beyond the subjective ratings of 1 to 5 stars which are typical of most reviews.

The goal of this paper is to analyze the restaurant review dataset to predict those reviews which are fake and to determine the sentiment of the review (positive or negative).

Analysis and Models

About the Data

The restaurant review dataset includes 92 food reviews which include a "Lie" label (true or false), "Sentiment" label (positive or negative) and the review text. The data was provided in a csv file format that was imported for the analysis. The "Lie" and "Sentiment" columns are both complete and do not require any additional processing except to convert the t/f and p/n values to 0 or 1. The review text is much more varied and has special characters like /, ! or ? and numbers that must be removed from the data.

Below are the steps used to clean the text data and prepare it for the analysis:

1. Merge the text which is in multiple columns in the csv file.
2. Remove the two rows with blank text reviews except for a "?".
3. Remove special characters "/" and "!"
4. Create column names of "Lie", "Sentiment", and "Review".
5. Convert the labels "Lie" and "Sentiment" to factors.
6. Tokenize the text by breaking up the sentences into words that will become the input for the models.
7. Convert the text to a Document Term Matrix with these options
    a. Remove stopwords, punctuation, numbers and separators.
    b. Change upper case to lower case.
    c. Stem the words.
    d. Implement the following formulas for minimum and maximum word frequency:
        i. minTermFreq = ndocs * 0.01
        ii. maxTermFreq = ndocs * 1
8. Partition the data one dataset for the Sentiment and another for Lie Detection. Then split these sets into 70% for training and 30% for testing. It was also very important to pick random samples of the data since the csv file is sorted with the negative reviews first and the positive reviews last.
9. Create separate datasets with the Lie label and the Partition label.

After the steps above, the training data includes 63 examples and the test data includes 39 examples as shown in Figures 1 and 2 below.

column 0: rownames	Lie	Sentiment	Review
	t	n	This restaurant is quite popular recently. Went there ...
71	t	p	i really like this one chicken wings restaurant. it is a ...
72	t	p	Ruby Tuesday is my favorite America Style Restauran...
45	t	n	The food at Lemongrass Restaurant was a mixed bag...
87	t	p	Blue Monkey Cafe is my favorite Japanese restaurant ...
3	f	n	After I went shopping with some of my friend_ we we...
86	t	p	Can t say too much about it. Just_ try it buddy ...
59	f	p	I went into the restaurant_ it decorated comfortably ...
37	t	n	The worst restaurant experience of my life happened...
2	f	n	i really like this buffet restaurant in Marshall street. t...
28	t	n	The worst restaurant I have ever been to ended up by...
7	f	n	I went to ABC restaurant two days ago and I hated th...
74	t	p	The best restaurant I have ever been was a small cor...
9	f	n	OMG. This restaurant is horrible. The receptionist did...
32	t	n	The restaurant looked good from the outside. My fat...

Figure 1- Sample review training data (70%)

	Lie	Sentiment	Review
13	f	n	I entered the restaurant and a waitress came by with ...
10	f	n	Yesterday_ I went to a casino-restaurant called NoF...
29	t	n	I went to this restaurant where I had ordered for the ...
77	t	p	I went to this awesome restaurant in San Francisco (I ...
16	f	n	In each of the diner dish there are at least one fly in i...
66	f	p	RIM KAAP One of the best Thai restaurants in town s...
21	f	n	Usually_ I use Yelp to find restaurant. The Yelp would...
15	f	n	This is the last place you would want to dine at. The ...
30	t	n	I went to XYZ restaurant last week and I was very dis...
54	f	p	I went to Joey s and had the best lasagna on the pla...
49	f	p	In my favorite restaurant Yuenan Restaurant. The no...
82	t	p	The restaurant looked pretty good_ the people aroun...
61	f	p	This place was one of the best restaurant I have been...
24	t	n	Pizza Hut Syracuse_ NY The only thing worth going h...
43	t	n	Restaurant : Samrat Food Ordered : Dal Tadka_ Baiga...

Figure 2 - Sample review testing data (30%)

With the data in separate training and test datasets, it is also important to determine how evenly the label data is distributed between these sets. Figures 3-6 show that there is an even distribution of values and no additional normalization is necessary.

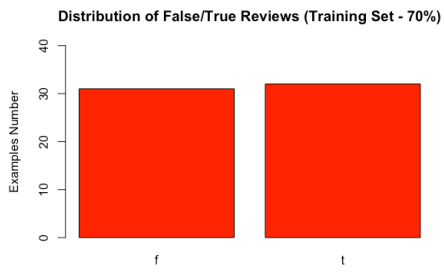


Figure 3 - False/true reviews in training

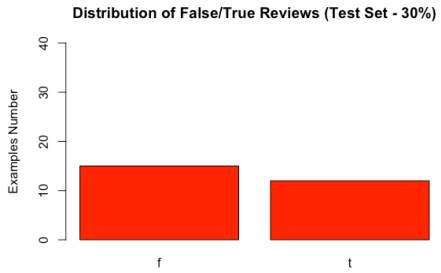


Figure 4 - False/true reviews in test

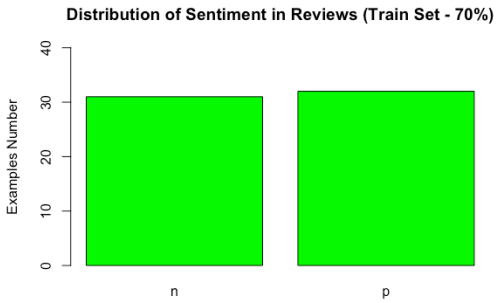


Figure 5 - Sentiment (negative/positive) in training

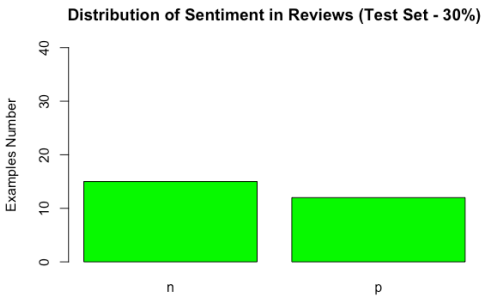


Figure 6 - Sentiment (negative/positive) in test

With the words in a document term matrix, the most frequent and least frequent words can be identified.

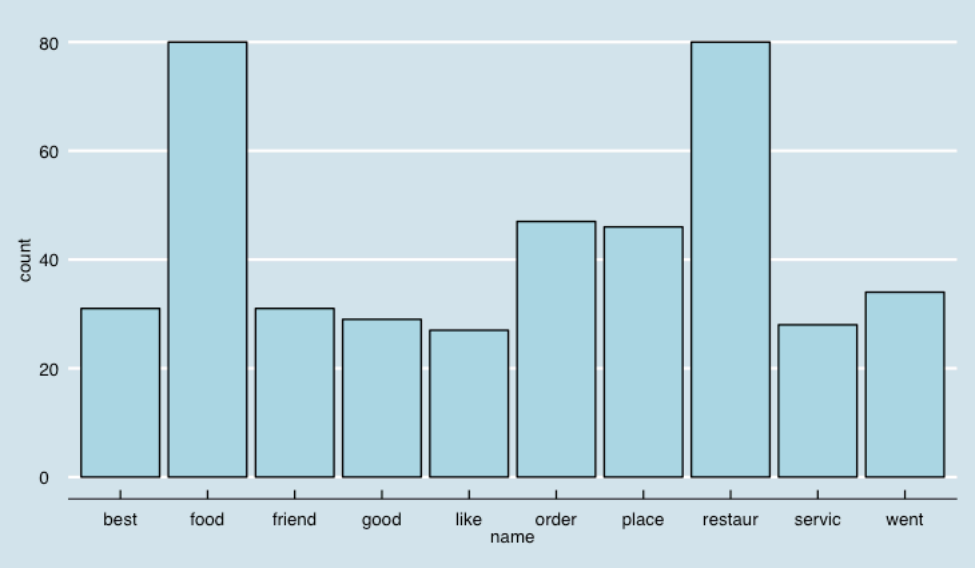


Figure 7 - Top 10 words by frequency

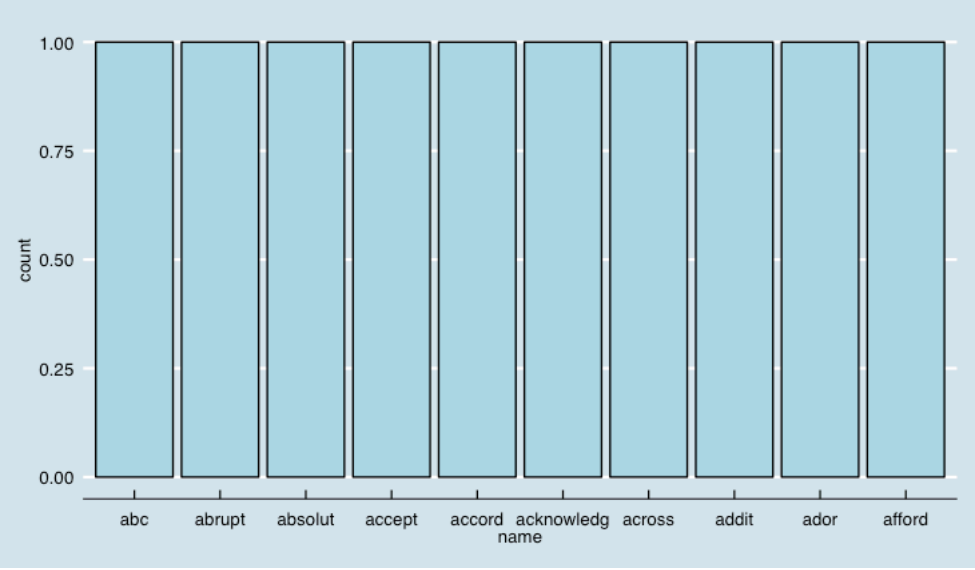


Figure 8 - Sample of least frequent words starting with "a"

The steps required for the exploratory data analysis, cleansing and processing are now complete and the data is ready for modeling.

Model 1 - Multinomial Naive Bayes

Naive Bayes is a simple classifier algorithm that is based on the work of English theologian and mathematician Thomas Bayes (1702-1761).[v] Bayes Rule is based on the probability of events and is summarized with this formula:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Each of the models in the analysis will include the following steps:

1. Confusion Matrix that shows the correct and wrong predictions.
2. Statistics for the model showing the data that will be summarized in the results section - accuracy, sensitivity (recall), positive predictive value (precision).
3. Chi-squared test showing X-squared, degrees of freedom and p-values.

Model 1 - Multinomial Naive Bayes - Sentiment

	Actual	
Predictions	1	2
1	8	2
2	7	10

Figure 9 - Naive Bayes confusion matrix for sentiment with test data

Accuracy = 66.7%

```
               Accuracy : 0.6667
                 95% CI : (0.4604, 0.8348)
    No Information Rate : 0.5556
    P-Value [Acc > NIR] : 0.1667

                  Kappa : 0.352

 Mcnemar's Test P-Value : 0.1824

            Sensitivity : 0.5333
            Specificity : 0.8333
         Pos Pred Value : 0.8000
         Neg Pred Value : 0.5882
             Prevalence : 0.5556
         Detection Rate : 0.2963
   Detection Prevalence : 0.3704
      Balanced Accuracy : 0.6833

       'Positive' Class : 1
```

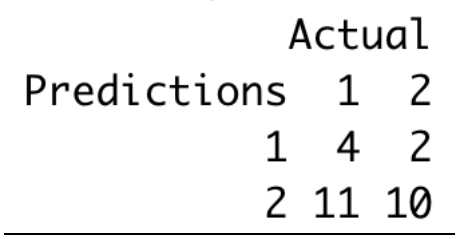
Figure 10 - Naive Bayes statistics for sentiment with test data

```
        Pearson's Chi-squared test with Yates' continuity correction

data:  table1
X-squared = 2.432, df = 1, p-value = 0.1189
```

Figure 11 - Chi-squared test for Naive Bayes sentiment with test data

Model 1 - Multinomial Naive Bayes - Lie Detection



Predictions	Actual 1	2
1	4	2
2	11	10

Figure 12 - Naive Bayes confusion matrix for lie detection with test data

Accuracy = 51.9%

```
               Accuracy : 0.5185
                 95% CI : (0.3195, 0.7133)
    No Information Rate : 0.5556
    P-Value [Acc > NIR] : 0.7206

                  Kappa : 0.093

 Mcnemar's Test P-Value : 0.0265

            Sensitivity : 0.2667
            Specificity : 0.8333
         Pos Pred Value : 0.6667
         Neg Pred Value : 0.4762
             Prevalence : 0.5556
         Detection Rate : 0.1481
   Detection Prevalence : 0.2222
      Balanced Accuracy : 0.5500

       'Positive' Class : 1
```

Figure 13 - Naive Bayes summary statistics for lie detection with test data

```
        Pearson's Chi-squared test with Yates' continuity correction

data:  table2
X-squared = 0.024107, df = 1, p-value = 0.8766
```

Figure 14 - Chi-squared score for Naive Bayes lie detection with test data

Model 2 - Support Vector Machine

> *"A Support Vector Machine (SVM) is a discriminative classifier formally defined by a separating hyperplane. In other words, given labeled training data (supervised learning), the algorithm outputs an optimal hyperplane which categorizes new examples. In two-dimensional space this hyperplane is a line dividing a plane in two parts where in each class lay in either side."*[vi]

The SVM analysis will utilize the Linear kernel for testing since this kernel works best on sparse text with many features.[vii]

Model 2 - Support Vector Machine - Sentiment

```
             Predicted Class
Actual Class  1  2
           1 12  3
           2  1 11
```

Figure 15 - SVM linear confusion matrix for sentiment with test data

Accuracy = 85.2%

```
Call:
svm(formula = trainNBsent$Sentiment ~ ., data = trainNBsentnolabel, kernel = "linear", scale = FALSE)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  linear
       cost:  1
      gamma:  0.001086957

Number of Support Vectors:  52
```

Figure 16 - SVM linear statistics for sentiment with test data

```
        Pearson's Chi-squared test with Yates' continuity correction

data:  table3
X-squared = 10.995, df = 1, p-value = 0.0009137
```

Figure 17 - Chi-squared score for SVM sentiment with test data

Model 2 - Support Vector Machine - Lie Detection

		Predicted Class	
Actual Class		1	2
	1	5	10
	2	2	10

Figure 18 - Linear SVM confusion matrix for lie detection with test data

Accuracy = 55.6%

```
Call:
svm(formula = trainNBlie$Lie ~ ., data = trainNBlienolabel, kernel = "linear", scale = FALSE)


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  linear
       cost:  1
      gamma:  0.001086957

Number of Support Vectors:  62
```

Figure 19 - SVM linear statistics for lie detection with test data

```
        Pearson's Chi-squared test with Yates' continuity correction

data:  table4
X-squared = 0.2917, df = 1, p-value = 0.5891
```

Figure 20 - Chi-squared score for SVM lie detection

Model 3 - Support Vector Machine - Sentiment and Lie Detection

After testing the Naive Bayes and SVM models, the SVM had better results and additional tuning will be performed in Model 3 with the goal of higher accuracy, precision and recall.

Changing the cost value for the Sentiment SVM to .6 improved the accuracy from 85% to 89%. A similar change in the Lie Detection SVM had no impact and the accuracy remained at 59%.

```
Confusion Matrix and Statistics

          Reference
Prediction  1  2
         1 13  1
         2  2 11

               Accuracy : 0.8889
                 95% CI : (0.7084, 0.9765)
    No Information Rate : 0.5556
    P-Value [Acc > NIR] : 0.0002236

                  Kappa : 0.7769

 Mcnemar's Test P-Value : 1.0000000

              Precision : 0.9286
                 Recall : 0.8667
                     F1 : 0.8966
             Prevalence : 0.5556
         Detection Rate : 0.4815
   Detection Prevalence : 0.5185
      Balanced Accuracy : 0.8917

       'Positive' Class : 1
```

Accuracy = 88.9%

Now that the models have been tested and the accuracy levels calculated, the results of the predictions will be discussed in the next section.

Results

For an analysis of restaurant reviews, it is not surprising that the word "food" appears more than 80 times in the text of the reviews. A closer analysis of the positive and negative words from the sentiment analysis provides more interesting information words provide clues to the false reviews in the dataset.

Figure 21 - Word cloud of most frequent words in restaurant reviews

Feature selection for the analysis of lie detection and sentiment is a key factor to producing good results. Through an information gain function, the lists in Figure 23 and 24 were produced to show the most important lie detection words on the left and sentiment words on the right. The words "music", "want", "give", "will" and "later" show up in both lists. Words like "best" and "great" show up in the sentiment list which is expected. However, the lie detection list is more varied. This is probably due to the more challenging task of choosing specific words to determine if someone is lying.

Information Gain - Top Features

```
  made 0.241628971
 music 0.241628971
indian 0.241628971
  want 0.231971709
  give 0.203051125
  will 0.203051125
 later 0.203051125
  hour 0.203051125
  wasn 0.203051125
  half 0.203051125
  bill 0.195238935
finish 0.195238935
waiter 0.195238935
  huge 0.195238935
   big 0.195238935
   fun 0.195238935
  idea 0.195238935
  keep 0.195238935
```

Figure 22 - Information gain features from lie detection data

```
      best 0.341778323
       bar 0.283532869
     great 0.283532869
      said 0.262010953
      beer 0.241628971
     music 0.241628971
     whole 0.241628971
       ask 0.241628971
    dinner 0.241628971
       hot 0.241628971
      love 0.241628971
       ate 0.241628971
 recommend 0.241628971
   without 0.241628971
      just 0.231971709
      want 0.231971709
     found 0.203051125
      give 0.203051125
      will 0.203051125
     later 0.203051125
```

Figure 23 - Information gain features from sentiment data

In addition to the chi-squared tests performed on each model, the "residuals" for this test also offer interesting insights into the value placed on each word.

Chi-Squared Test - Top Features

place	best	good	can	around	ever	want	price	take	later
-0.353	-0.299	-0.211	-0.18	-0.18	-0.148	-0.148	-0.148	-0.148	-0.115

will	began	get	life	order	time	flyer	help	threw	waitress
0.829	0.879	0.879	0.879	0.879	0.879	0.931	0.931	0.931	0.931

Figure 24 - Chi-squared features sorted by residuals

When utilizing the 90 reviews to make predictions, lie detection proved to be much more challenging and the best model could only predict correctly 56% of the time (not much better than a coin toss). However, sentiment produced better results with Naive Bayes at 67% accuracy and SVM at 85%. With additional tuning of the cost value to .6, the SVM for sentiment improved to 89%. The precision recall and chi-squared values were all much better in the SVM.

Category	Parameter Setting	Overall Accuracy	Precision	Recall/Sensitivity	Chi Squared p-value
Sentiment	Naive Bayes	.67	.80	.53	.1189
Sentiment	SVM - Linear Kernel, cost = 1	.85	.92	.80	.0009
Sentiment	SVM - Linear Model, cost =.6	.89	.93	.87	.0002
Lie	Naive Bayes	.52	.67	.27	.8766
Lie	SVM - Linear Kernel, cost = 1	.56	.71	.33	.5891

Table 1 - Comparison of model statistics

A comparison of the Chi Square values shows the sentiment models had low p-values so the null hypothesis of the independence assumption can be rejected for the SVM Linear Kernals. The p-values of .0009 and .0002 respectively are below the typical threshold of .05. For the other values above .05, the null hypothesis cannot be rejected.

Conclusion

The use of machine learning techniques has proved to be beneficial for solving many challenging problems like determining the positive or negative sentiment of online food reviews. Making a prediction about whether a review is genuine or fabricated is a more challenging task for most statistical models.

Using several of the popular techniques for predictions, this analysis was able to determine if a review was positive or negative with 89% accuracy. This capability is very useful for restaurants and other industries to develop marketing and customer service plans using this information. For example, when using a 1 star to 5 star scale for ratings of restaurants, it is often difficult to understand the true values given the subjectivity of the ratings. One restaurant guest may be a very difficult customer and never give higher than a 4 while another customer may

always give a 5 (even when they were not completely satisfied). Using sentiment analysis of these reviews is a way to uncover these nuances in the reviews to gain a deeper understanding of customer behavior.

Spotting fake reviews online is much more difficult and a growing problem for retailers and marketers. A 2019 CBS News report cited a Fakespot report which estimated that 52% of Walmart reviews and 30% of Amazon reviews are fake.[viii] Unfortunately, the analysis for this paper does not provide an easy solutions to solve this problem. The algorithms and models used were only able to spot the fake reviews 56% of the time. However, there is good news on the horizon since there is a tremendous amount of research occurring for text mining and natural language processing. Google unveiled a new technology in November 2018 that has great promise for advancing the understanding of text and human speech by computers.[ix] However, until this technology is proven for tasks such as spotting fake reviews, you should be skeptical of that fabulous review you just read on Yelp* for that new restaurant in your town. There is a good chance that it might not be written by a person or computer that actually dined at the restaurant.

[i] Editor, Wikipedia. “Customer Review.” *Wikipedia*, Wikimedia Foundation, 23 July 2019, en.wikipedia.org/wiki/Customer_review.

[ii] Streitfeld, David. “In a Race to Out-Rave, 5-Star Web Reviews Go for $5.” *The New York Times*, The New York Times, 19 Aug. 2011, www.nytimes.com/2011/08/20/technology/finding-fake-reviews-online.html.

[iii] Roesler, Peter. “FTC Fines Company $12.8 Million for Fake Amazon Reviews.” *Inc.com*, Inc., 4 Mar. 2019, www.inc.com/peter-roesler/ftc-fines-company-128-million-for-fake-amazon-reviews.html.

[iv] Chowdhary, Neha, and Anala Pandit. “Fake Review Detection Using Classification.” *International Journal of Computer Applications*, vol. 180, no. 50, 2018, pp. 16–21., doi:10.5120/ijca2018917316.

[v] Britannica, The Editors of Encyclopaedia. “Thomas Bayes.” *Encyclopædia Britannica*, Encyclopædia Britannica, Inc., www.britannica.com/biography/Thomas-Bayes.

[vi] Patel, Savan. “Chapter 2 : SVM (Support Vector Machine) - Theory.” *Medium*, Machine Learning 101, 4 May 2017, medium.com/machine-learning-101/chapter-2-svm-support-vector-machine-theory-f0812effc72.

[vii] Kaestner, Celso Antonio Alves. “Support Vector Machines and Kernel Functions for Text Processing.” *Revista De Informática Teórica e Aplicada*, vol. 20, no. 3, 2013, p. 130., doi:10.22456/2175-2745.39702.

[viii] Picchi, Aimee. “Buyer Beware: Scourge of Fake Reviews Hitting Amazon, Walmart and Other Major Retailers.” *CBS News*, CBS Interactive, 28 Feb. 2019, www.cbsnews.com/news/buyer-beware-a-scourge-of-fake-online-reviews-is-hitting-amazon-walmart-and-other-major-retailers/.

[ix] Metz, Cade. “Finally, a Machine That Can Finish Your Sentence.” *The New York Times*, The New York Times, 18 Nov. 2018, www.nytimes.com/2018/11/18/technology/artificial-intelligence-language.html.