End Mark Prediction?

Eric Mark Martin, Jonathan Zwiebel
ericmarkmartin@stanford.edu, jzwiebel@stanford.edu
CS 229: Machine Learning - Autumn 2018

Introduction

Practically all of our mobile devices use some form of autocorrect, predictive typing, or diction to complete our sentences for us. Yet, if you open your phone and type a sentence your device will almost certainly punctuate it (if at all) with a period whether it's "Talk to you later.", "Come over!" or "You up?". Our project creates and compares a number of models based on techniques learned in CS 229 to predict the end mark of a English sentence.

Problem Definition

The goal of our project is to be able to correctly punctuate variable-length English sentences with one of three end marks: periods, question marks, or exclamation marks (denoted PERIOD, QMARK, EXPOINT in this poster). We want to punctuate sentences drawn from the distribution of English sentences so we did not reweight our data to have equal proportions of each punctuation mark.

```
i came ; i saw ; i conquered → .
do you have <NUMBER> pickles → ?
<PROPER>, stop now → !
```

For our baseline we used proportional guessing, a model that would randomly guess an end mark using proportions taken from the training set. For our oracle we used human-level assessment, giving a number of people sentences from our test set, and asking them to predict the end mark.

We evaluated five different classes of models – logistic regression, naive bayes, SVM, random forests, and fully connected neural networks – and compared their performance. Each model was evaluated over matching 90-5-5 train-dev-test splits and scored using standard classification metrics.

Models and Results

Proportional Guessing - Baseline

An estimated distribution of labels is found using the training set. Predictions are draw as random samples from this distribution.

	P	Q	E
P	1568	140	190
Q	174	16	8
E	192	19	29

	Prec	Rec	F1	Supp
P	0.81	0.81	0.81	1932
Q	0.08	0.08	0.08	198
E	0.13	0.12	0.12	240
Macro	0.34	0.34	0.34	2370

Logistic Regression (2 Models)

A standard multiclass logistic regression was run with 20005 features. We tested both binary and bag-of-words feature vectors and found binary feature vectors to be our strongest logistic regression model.

	P	Q	E
P	1874	26	32
Q	116	72	10
E	166	13	61

	Prec	Rec	F1	Supp
P	0.87	0.97	0.92	1932
Q	0.65	0.36	0.47	198
E	0.59	0.25	0.36	240
Macro	0.70	0.53	0.58	2370

L2 loss, 100 iterations, trained with stochastic average gradient

Naive Bayes (3 Models)

A standard naive bayes model designed for multiclass applications. We tested multinomial with bag-of-words, bernoulli with binary vectors, and gaussian with bag-of-words distributions and found bernoulli to perform the best.

	P	Q	E
P	1606	23	303
Q	76	47	75
E	73	4	163

	Prec	Rec	F1	Supp
P	0.92	0.83	0.87	1932
Q	0.64	0.24	0.35	198
E	0.30	0.68	0.42	240
Macro	0.62	0.58	0.54	2370

Bernoulli distribution, uniform prior

Human Level - Oracle

Over a reduced test set we asked humans to classify sentences using the same tokenization scheme as given to the learned models.

	P	Q	E
P	151	1	10
Q	3	7	1
E	12	2	5

	Prec	Rec	F1	Supp
P	0.91	0.93	0.92	162
Q	0.70	0.64	0.67	11
E	0.31	0.26	0.29	19
Macro	0.64	0.61	0.62	192

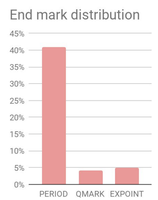
Random Forests (4 Models) - Best

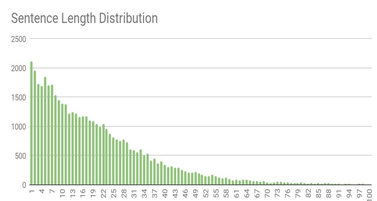
A standard random forest model aggregating 100 decision trees trained on bag-of-words feature vectors. Each tree was allowed to train until each leaf was pure. We also attempted stopping the constituent trees at depths 5, 10, and 50, but the model model highly over classified periods. As shown by the cross-validation metrics, bagging prevented the model from overfitting, even without a limit on tree depth.

	P	Q	E
P	1909	11	22
Q	133	56	9
E	170	11	62

	Prec	Rec	F1	Supp
P	0.87	0.98	0.92	1932
Q	0.79	0.34	0.47	198
E	0.68	0.26	0.37	240
Macro	0.78	0.53	0.59	2370

gini loss, 100 estimators, trained to unlimited depth





Support Vector Machine (2 Models)

A standard SVM using a radial basis function (RBF) kernel. We tried both a Support Vector Classifier (SVC) and a Stochastic Gradient Descent Classifier (SGD). We also attempted a TFIDF vectorizer for both types of SVMs. We found SGD with a bag-of-words to work best.

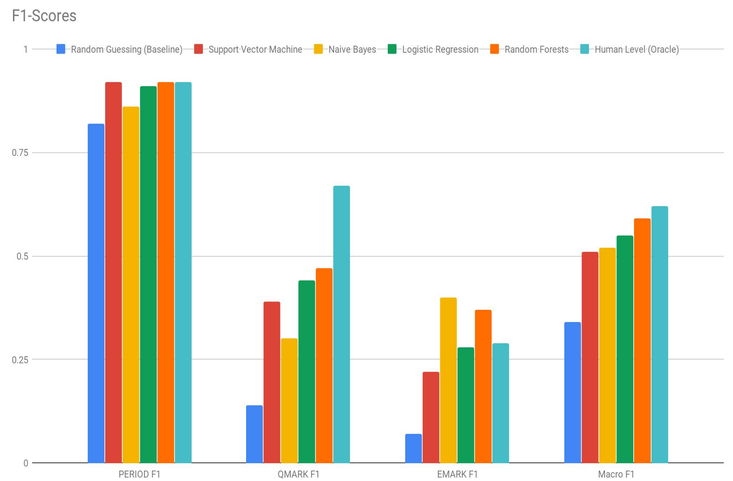
	P	Q	E
P	1903	17	12
Q	128	69	1
E	182	16	42

	Prec	Rec	F1	Supp
P	0.86	0.98	0.92	1932
Q	0.68	0.35	0.46	198
E	0.76	0.17	0.28	240
Macro	0.77	0.50	0.55	2370

hinge loss, linear kernel

Neural Network

Using the remaining time for this project we plan on developing a fully connected deep neural network with an extended feature vector that contains a one-hot for the first word, one-hot for the last word, a binary vector for words that start clauses, and a binary vector for the length of the sentence.



Data

We drew data from 10 of the top English books available at project Gutenberg, available for free use.

1. A Christmas Carol by Charles Dickens
2. Pride and Prejudice by Jane Austen
3. Frankenstein by Mary Wollstonecraft Shelley
4. A Tale of Two Cities by Charles Dickens
5. Heart of Darkness by Joseph Conrad
6. A Modest Proposal by Jonathan Swift
7. Moby Dick by Herman Melville
8. Dracula by Bram Stoker
9. Alice's Adventures in Wonderland by Lewis Carol
10. The Adventures of Sherlock Holmes by Sir Arthur Conan Doyle

We wanted to ensure that we could extract usable examples even from complex grammar structures such as dialogue and clauses . Additionally to maximize the number of QMARK and EXPOINT samples we needed to ensure that we could extract standalone sentences within quotations (ex: "'How are you?' said Frankenstein."). Our definition for a sentence was a sequence of space-separated words beginning with a capital word, ending with an end mark, and unbroken by any single or double quotation marks.

Additionally we added five special-use tokens: <NUMBER>, <COMMA>, <SEMICOLON>, <PROPER>, <UNKNOWN> to handle important cases not counted in our dictionary.

Analysis

We found that all of our best models in class were able to outperform our baseline but none were able to beat our oracle (as measured by Macro-averaged F1 score). In order from best to worst model we have random forests, logistic regression, naive bayes, and SVMs. Still we found that their differences in macro-averaged F1 score were minimal and could easily have been the result of poorly tuned hyperparameters.

Additionally we found that all of the models did a better job with precision on QMARKs than on EXPOINTs. This result is in-line with the intuitive understanding that questions can be found by looking for specific questions words (ex. "who", "will", "when") while exclamatory sentences are often structurally similar and indistinguishable to sentences with periods.

Future Work

We were quite impressed with our models and would like to experiment with deep learning techniques, particularly sequence based models. Some additional models we would like to try:

- Fully-connected neural network with expanded feature vector
- RNN with one-hot vector
- RNN with word vector from existing embeddings

We would also like to try a model that uses surrounding sentences as context to assist in end mark prediction. For this we would need a bidirectional RNN.

References

- Baron, D., Shriberg, E., & Stolcke, A. (2002). Automatic punctuation and disfluency detection in multi-party meetings using prosodic and lexical cues. In *Seventh International Conference on Spoken Language Processing*.
- Christensen, H., Gotoh, Y., & Renals, S. (2001). Punctuation annotation using statistical prosody models. In *ISCA tutorial and research workshop (ITRW) on prosody in speech recognition and understanding*.
- Maas, A. L., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011, June). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies-volume 1* (pp. 142-150). Association for Computational Linguistics.
- Makhoul, J., Baron, A., Bulyko, I., Nguyen, L., Ramshaw, L., Stallard, D., ... & Xiang, B. (2005). The effects of speech recognition and punctuation on information extraction performance. In *Ninth European Conference on Speech Communication and Technology*.
- Ueffing, N., Bisani, M., & Vozila, P. (2013). Improved models for automatic punctuation prediction for spoken and written text. In *Interspeech* (pp. 3097-3101).