R U  or  ?

Sentiment Analysis of Facebook Photo Comments

Ben Blamey
PhD Candidate
b.blamey@outlook.uwic.ac.uk
Cardiff School of Management, Cardiff Metropolitan University

Supervisory Team:
Dr Tom Crick, Director of Studies — tcrick@cardiffmet.ac.uk
Dr Giles Oatley, Supervisor — goatley@cardiffmet.ac.uk

Introduction

Sentiment Analysis is the classification of a document, such as a movie review [10], or a tweet [5,6], according to whether it contains generally positive (funny, complimentary, etc.) or generally negative (offensive or sad) opinions or statements. We call this the overall *sentiment* of the document. Particular words are more common in positive and negative documents respectively, and we can estimate the sentiment of a document through statistical analysis of the words within it.

Here, we describe the application of this technique to comments associated with photographs from the popular social networking site Facebook (see example in Figure 6), the first time this technique has been applied to this source of documents. As part of our wider work on developing methods for the selection of important content from a user's social media footprint, we required techniques for sentiment analysis that would be effective for text from online social networks (OSNs). The n-gram model of language used by sentiment analysis can be formulated in two ways: either as a sequence of words (or overlapping sequences of n consecutive words), or more rarely, as a set of all overlapping n-character strings, without consideration of individual words (see example in Figure 1).

Our hypothesis is that the character n-gram model will be intrinsically well-suited to the 'unnatural' language common to OSN corpora, and will achieve higher accuracy in the binary sentiment classification task. The aim was to see whether the character n-gram model offers improved accuracy on OSN corpora, with the movie review corpus serving as a non-OSN control.

```
w00t...bad ass shades!
```

Words as Features:

```
w00t
bad
ass
shades
```

Word-Bigrams:

```
w00t bad
bad ass
ass shades
```

4-Character-grams:

```
w00t.
00t..
0t...
t...b
...ba
..bad
.bad a
bad as
```

...and so on

Figure 1: Word vs. Character N-Grams -Extraction of Features

Background and Related Work

Text found in social media is rich in 'unnatural' language phenomena, defined as "informal expressions, variations, spelling errors ... irregular proper nouns, emoticons, unknown words" [1]. Existing NLP tools are known to struggle with such language, Ritter et al. have "demonstrated that existing tools for POS tagging, Chunking and Named Entity Recognition perform quite poorly when applied to Tweets"[2, pp. 1532]. We wondered whether the flexibility of the character n-gram language model would make it more appropriate than the word-based language model, for OSN text.

There are just a few examples of sentiment analysis employing the character n-gram model: Rybina [4] did binary sentiment classification on a selection of German web corpora, mostly product reviews, and finds that character n-grams consistently outperforms word n-grams by 4% on F1 score. This is an extremely interesting result, and our desire to repeat her findings were a key motivation for this work, but some details are unclear: the classifier that was used is closed-source, and it isn't obvious what method was used to label the data. Other studies have more mixed findings; Ye et al. [3] classified travel reviews using both character and word n-gram models with LingPipe, and found that neither was consistently better. Much work has studied sentiment analysis of OSN corpora, especially Twitter, using the word n-gram model, Go et. al [5], Pak and Paroubek [6] were amongst the first, with state of the art (Bespalov et al. [7]) approaches achieving accuracy of more than 90%.

Sentiment analysis studies of Facebook are comparatively rare, in one such study Kramer computed the positivity of status messages to create an index of "Gross National Happiness" [8]. To our knowledge, there have been no documented studies of sentiment analysis applying the character n-gram model to online social network text, and none looking at Facebook photo comments using either language model.

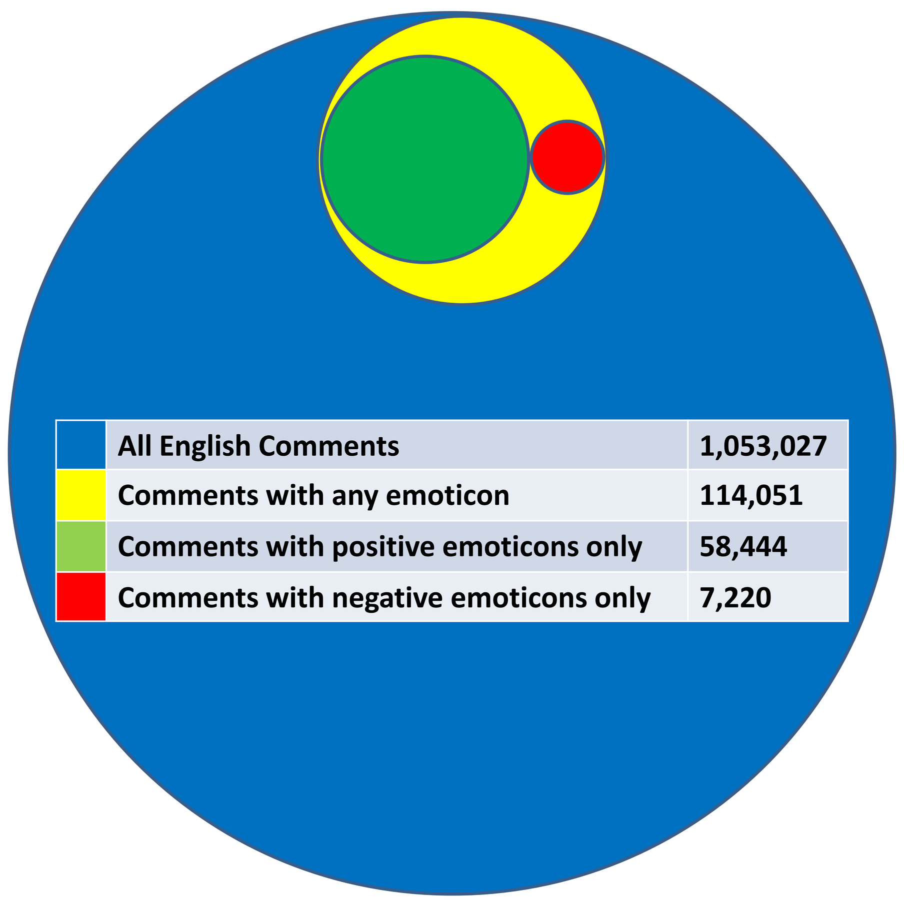

Figure 2: Number of Comments, with key.

Methods

An emoticon is a sequence of characters with the appearance of a facial expression, see Figure 3 for some examples! For Tweets and Facebook photo comments, we follow the approach of Read [9] of using the presence of emoticons as labels of sentiment (thereby allowing unsupervised machine learning), and them removing them from the text for classification. Around 20,000 Tweets were gathered from the Twitter Search API, using positive and negative emoticons as search terms. With Facebook photo comments, the percentage of comments that contained emoticons was low, (negative emoticons were found in less than 1% of the corpus), so it was necessary to collect a large amount of data (see Figure 2, and key).

URLs and Twitter 'mentions' and hash-tags were replaced with respective single characters, so their meaning is captured in both word and character n-gram models. Only the documents with exclusively happy { :) :D :-) =) :) :-D :o) =D } or sad { :(:-(:'(: (} emoticons only were selected, and emoticons were chosen that accurately mean 'happy' and 'sad' - documents with :P ('sticking out tongue') and similar emoticons were excluded, because they tend to be used for jokes and insults - which might confuse the classifiers. Also excluded were 'winks', e.g. ';-)', in case they were used to indicate flirtatious remarks, which again may disrupt classification. This yielded labelled corpora of 7000 Tweets and 7000 Facebook photo comments, alongside 1386 movie reviews for control.

Can You Guess the Emoticons?

(Answers at the bottom...)

ಠ_ಠ

:)

~(_8^(I)

(^▼^)

=:o]

:-P

:-###..

;-)

Figure 3: Examples of Emoticons

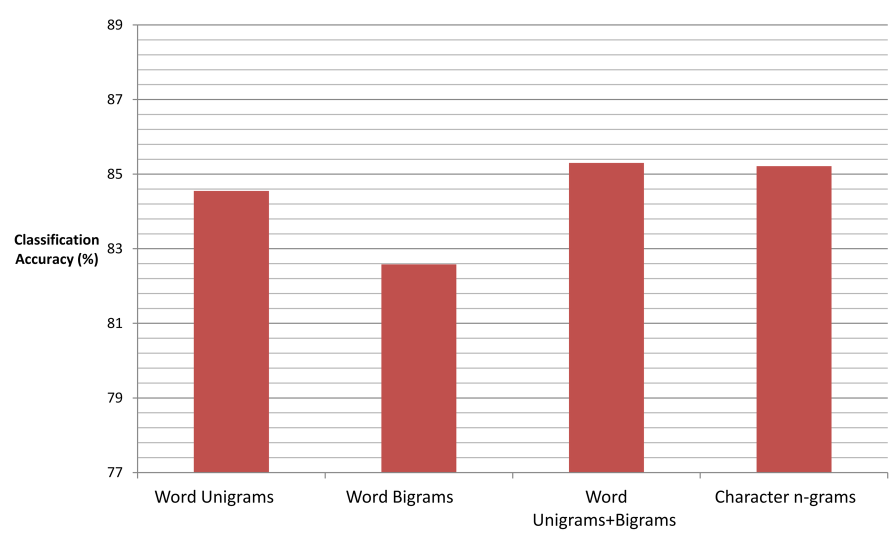

Figure 4: Online Social Network Corpora – Mean Classification Accuracies

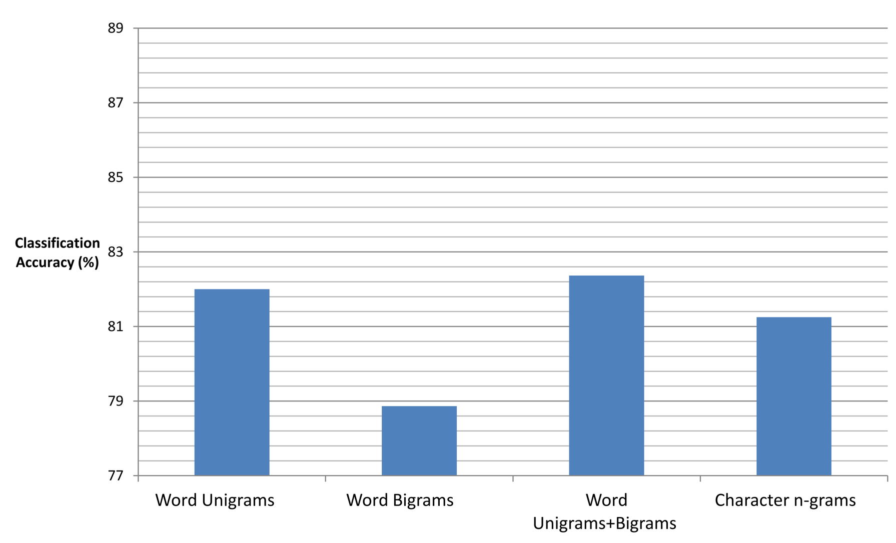

Figure 5: IMDB - Mean Classification Accuracies

Methods (Continued)

For word-based classification, further processing is necessary. Elongated words are squashed to a maximum of 3 repetitions, e.g. '<3<3<3<3<3' becomes '<3<3<3'. We followed the approach of Das and Chen [11] to handle negation (as in "I am not happy") by labelling words in a negative context, yielding a small improvement to classification, consistent with other studies.

For the text classification itself, we used 4 feature-sets: word unigrams, bi-grams, and the union of both, and the union of the sets of character n-grams where n=2,..,8. We do not trim low-frequency features, as it is generally discouraged except where necessary for performance. Three standard classifiers were used in our experiment:

- Naive Bayes (based on feature frequency), with 'plus-one' smoothing.
- Maximum Entropy (i.e. 'log-linear discrimitive') of the Stanford Classifier, with default settings (a quadratic prior with $\sigma = 1$).
- SVM light classifier [12], with default settings.
- For character n-grams only, the LingPipe [13] DynamicLM Classifier.

The classification accuracies are shown in Table 1, and summarized in Figures 4 and 5.

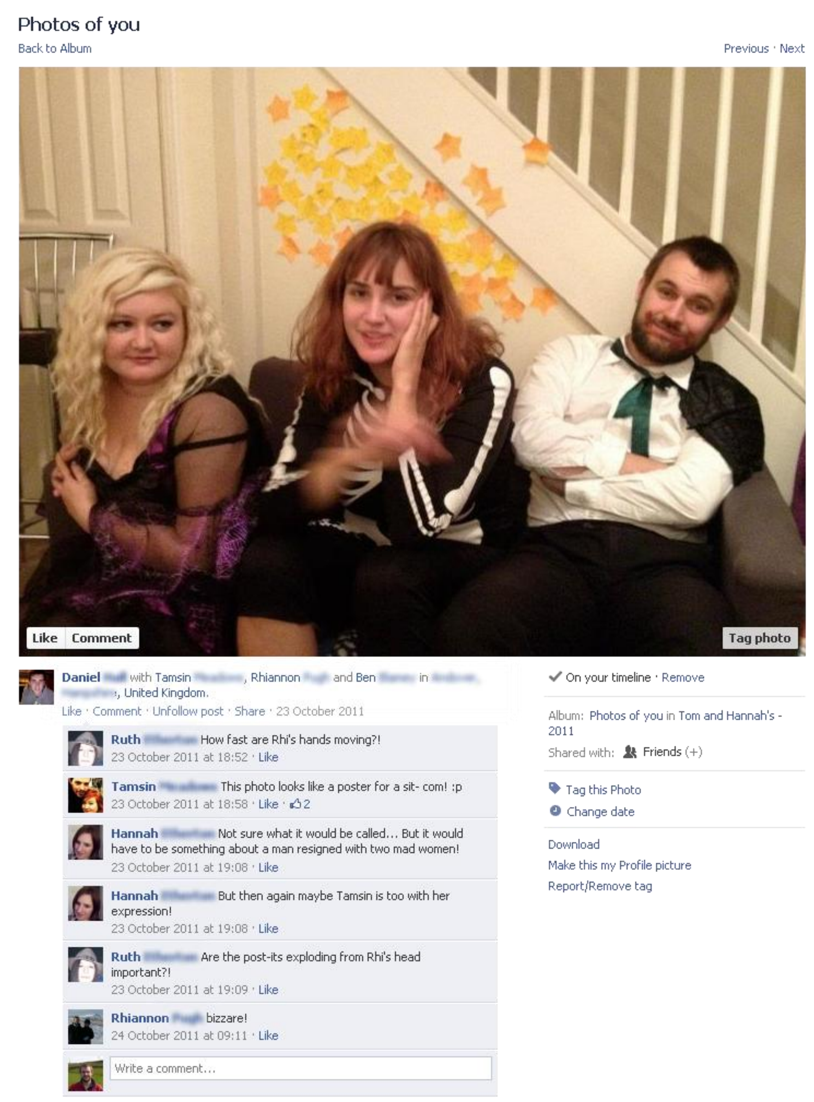

Figure 6: Typical Facebook Photo, with Comments

Conclusions

Neither word- nor character-grams yielding consistently higher accuracy (the findings contradict some existing studies) Looking more closely at our data, we can see that character n-grams consistently beat word unigrams, which is understandable, as 8 characters will often be enough to contain more than one word, and including word bigrams has often given better accuracy than unigrams alone.

Unfortunately, our hypothesis that character n-grams will be intrinsically well-suited to the 'un-natural' language common to OSN corpora was false: there doesn't seem to be a significant performance difference between the OSN and non-OSN corpora, for social network text and unnatural language – but the language of the social web is a pressing challenge for NLP; and as discussed, many of the existing tools struggle with it. The size of our corpus may have been an issue, to reap the full benefits of the character n-gram model more training data might be needed - LingPipe is designed to scale character n-gram data to the order of gigabytes [14].

Putting the issue of unnatural language aside, proponents of character n-gram models have a point: studies (including this one) have repeatedly shown that the character n-gram can perform as well as simple word n-gram models – whilst being considerably simpler to implement, especially when tokenization is hard, such as in Asian languages. There is no one 'right' way to do tokenization, negation, word squashing, stemming, and precise details are often thought too tedious for publication. Our tendency to automatically adopt the word-based model may suggest some degree of human-centric bias in our research thinking, or perhaps too strong a focus on English and other Western languages, within sentiment analysis research.

		Bayes	MaxEnt	SVM	LingPipe
IMDB Movie Reviews	Word Unigrams	80.5%	82.7%	82.8%	
	Word Bigrams	80.5%	79.4%	76.7%	
	Word Unigrams+Bigrams	81.4%	83.5%	82.2%	
	Character n-grams	81.9%	**84.6%**	82.6%	75.9%
Tweets	Word Unigrams	88.7%	91.2%	88.8%	
	Word Bigrams	89.5%	91.4%	90.5%	
	Word Unigrams+Bigrams	90.6%	91.6%	90.8%	
	Character n-grams	90.8%	91.9%	90.6%	**92.0%**
Facebook Photo Comments	Word Unigrams	80.2%	79.8%	78.6%	
	Word Bigrams	75.8%	75.8%	72.5%	
	Word Unigrams+Bigrams	**80.5%**	80.0%	78.3%	
	Character n-grams	80.4%	80.1%	80.0%	75.9%

Table 1: 3-fold cross-validated accuracies. The best performing configuration of feature-set and classifier for each corpus is shown in bold.

References

1. M. Hagiwara. Unnatural Language Processing Contest 2nd will be held at NLP2011 (2010). URL http://bit.ly/dGvUnR
2. A. Ritter, S. Clark, Mausam, O. Etzioni, in Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (ACL, Edinburgh, Scotland, UK., 2011), pp. 1524–1534
3. Q. Ye, Z. Zhang, R. Law, Expert Syst. Appl. 36(3), 6527 (2009). DOI 10.1016/j.eswa.2008.07.035
4. K. Rybina, Sentiment analysis of contexts around query terms in documents. Master's thesis (2012)
5. A. Go, R. Bhayani, L. Huang, Processing 150(12), 1 (2009)
6. A. Pak, P. Paroubek, in Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10) (ELRA, Valletta, Malta, 2010)
7. D. Bespalov, B. Bai, Y. Qi, A. Shokoufandeh, in Proceedings of the 20th ACM international conference on Information and Knowledge management (ACM, New York, NY, USA, 2011), CIKM '11, pp. 375–382
8. A.D.I. Kramer. Facebook gross national happiness application (2010). URL http://www.facebook.com/gnh/
9. J. Read, Proceedings of the ACL Student Research Workshop on ACL 05 43(June), 43 (2005)
10. B. Pang, L. Lee, S. Vaithyanathan, in Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing (EMNLP) (Association for Computational Linguistics,2002), pp. 79–86
11. S. Das, M. Chen, in Asia Pacific Finance Association Annual Conference. (APFA) (2001)
12. T. Joachims, Making large-scale SVM learning practical (MIT press, 1999)
13. Alias-i. Lingpipe 4.1.0 (2008). URL http://alias-i.com/lingpipe
14. B. Carpenter, in Proceedings of the Workshop on Software (Association for Computational Linguistics, Stroudsburg, PA, USA, 2005), Software '05, pp. 86–99

ANSWERS: Look of disapproval, Happy/Funny, Homer Simpson, Happy (Eastern), Bill Clinton, Tongue, Being Sick, Wink