Mining wisdom

Anders Søgaard
Center for Language Technology
University of Copenhagen
DK-2300 Copenhagen S
soegaard@hum.ku.dk

Abstract

Simple text classification algorithms perform remarkably well when used for detecting famous quotes in literary or philosophical text, with f-scores approaching 95%. We compare the task to topic classification, polarity classification and authorship attribution.

1 Introduction

Mark Twain famously said that 'the difference between the right word and the almost-right word is the difference between lightning and a lightning bug.' Twain's quote is also about the importance of quotes. A great quote can come in handy when you are looking to inspire people, make them laugh or persuade people to believe in a particular point of view. Quotes are emblems that serve to remind us of philosophical or political stand-points, world views, perspectives that comfort or entertain us. Famous quotes such as 'Cogito ergo sum' (Descartes) and 'God is dead' (Nietzsche) occur millions of times on the Internet.

The importance of quotes has motivated publishing houses to create and publish large collections of quotes. In this process, the editor typically spends years reading philosophy books, literature, and interviews to find good quotes, but this process is both expensive and cumbersome. In this paper, we consider the possibility of automatically learning what is a good quote, and what is not.

1.1 Related work

While there seems to have been no previous work on identifying quotes, the task is very similar to widely studied tasks such as topic classification, polarity classification, (lexical sample) word sense disambiguation (WSD) and authorship attribution. In most of these applications, texts are represented as bags-of-words, i.e. a text is represented as a vector $\mathbf{x} = \langle x_1, \ldots, x_N \rangle$ where each x_i encodes the presence and possibly the frequency of an n-gram. It is common to exclude stop words or closed class items such as pronouns and adpositions from the set of n-grams when constructing the bags-of-words. Sometimes lemmatization or word clustering is also used to avoid data sparsity.

Topic classification is the classic problem in text classification of distinguishing articles on a particular topic from other articles on other topics, say sports from international politics and letters to the editor. Several resources exist for evaluating topic classifiers such as Reuters 20 Newsgroups. Common baselines are Naive Bayes, logistic regression, or SVM classifiers trained on bag-of-words representations of n-grams with stop words removed.

While newspaper articles typically consist of tens or hundreds of sentences, famous quotes typically consist of one or two sentences, and it is interesting to compare quotation mining to work on applying topic classification techniques to short texts or sentences (Cohen et al., 2003; Wang et al., 2005; Khoo et al., 2006). Cohen et al. (2003) and Khoo et al. (2006) classify sentences in email wrt. their role in discourse. Khoo et al. (2006) argue that extending a bag-of-words representation with frequency counts is meaningless in small text and restrict themselves to binary representations. They show empirically that excluding stop words and lemmatization

Workshop on Computational Linguistics for Literature, pages 54–58,
Montréal, Canada, June 8, 2012.

both lead to impoverished results. We also observe that stop words are extremely useful for quotation mining.

Polarity classification is the task of determining whether an opinionated text about a particular topic, say a user review of a product, is positive or negative. Polarity classification is different from quotation mining in that there is a small set of strong predictors of polarity (pivot features) (Wang et al., 2005; Blitzer et al., 2007), e.g. the polarity words listed in subjectivity lexica, including opinionated adjectives such as *good* or *awful*. The meaning of polarity words is context-sensitive, however, so context is extremely important when modeling polarity.

Some quotes are expressions of opinion, and there has been some previous research on polarity classification in direct quotations (not famous quotes). Balahur et al. (2009) present work on polarity classification of newspaper quotations, for example. They use an SVM classifier on a bag-of-words representation of direct quotes in the news, but using only words taken from subjectivity lexica as features. Drury et al. (2011) present a strategy for polarity classification of direct quotations from financial news. They use a Naive Bayes classifier on a bag-of-words models of unigrams, but learn group-specific models for analysts and CEOs.

WSD. The lexical sample task in WSD is the task of determining the meaning of a specific target word in context. Mooney (1996) argues that Naive Bayes classification and perceptron classifiers are particularly fit for lexical sample word sense disambiguation problems, because they combine weighted evidence from *all* features rather than select a subset of features for early discrimination. This of course also holds for logistic regression and SVMs. Whether a sentence is a good quotation or not also depends on many aspects of the sentence, and experiments on held-out data comparing Naive Bayes with decision tree-based learning algorithms, also mentioned in Sect. 5, clearly demonstrated that early discrimination based on single features is a bad idea. In this respect, quotation mining is more similar to lexical sample WSD than to topic and polarity classification where there is a small set of pivot features.

Authorship attribution is the task of determining which of a given set of authors wrote a particular text. One of the insights from authorship attribution is that stop words are important when you want to learn stylistic differences. Stylistic differences can be identified from the distribution of closed class words (Arun et al., 2009). As already mentioned, we observe the same holds for quotation mining.

Positives
Two lives that once part are as ships that divide.
My appointed work is to awaken the divine nature that is within.
Discussion in America means dissent.
Negatives
The business was finished, and Harriet safe.
But how shall I do? What shall I say?
I am quite determined to refuse him.

Figure 1: Examples.

In conclusion, early-discrimination learning algorithms do not seem motivated for applications such as mining quotes where pivot features are hard to choose *a priori*. Furthermore, we hypothesize that it is better *not* to exclude stop words. Quotation mining can thus in our view be thought of as an application that is similar to sentence classification in that famous quotes are relatively small, and similar to authorship attribution in that style is an important predictor of whether a sentence is a famous quote.

2 Data

We obtain the database of famous quotes from a popular on-line collection of quotes[1] and use philosophical and literary text sampled from the Gutenberg corpus as negative data. In particular we use the portion of Gutenberg documents that is distributed in the corpora collection at NLTK.[2] This gives us a total of 44,385 positive data points (famous quotes) and 247,115 negative data points (ordinary sentences). In our experiments we use the top 4,000 data points in each sample, i.e. a total of 8,000 data points, except for when we derive a learning curve later on, which uses up to $2 \times 20,000$ data points. Some sample data points are presented in Figure 1.

3 Experiment

Each data point is represented as a binary bag-of-words - or bag-of-n-grams, really. Our initial hypothesis was to include stop words and keep infor-

[1] http://quotationsbook.com

[2] http://nltk.org

mation about case (capital letters). Stop words are extremely important to distinguish between literary styles, and we speculated that quotes can be distinguished from ordinary text in part by their style. We also speculated that there would be a tendency to capitalize some words in quotes, e.g. 'God', 'the Other', or 'the World'. Finally, we hypothesized that including more context would be beneficial. Our intuition was that sometimes larger chunks such as 'He who' may indicate that a sentence is a quote without the component words being indicative of that in any way.

To evaluate these hypotheses we considered a logistic regression classifier over bag-of-word representations of the quotes and our neutral sentences. We used a publicly available implementation[3] of limited memory L-BFGS to find the weights that maximize the log-likelihood of the training data:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} \sum_i y(i) \log \frac{1}{1+e^{-\mathbf{w}\cdot\mathbf{x}}} + (1-y^{(i)})$$

$$\log \frac{e^{-\mathbf{w}\cdot\mathbf{x}}}{1+e^{-\mathbf{w}\cdot\mathbf{x}}}$$

where $\mathbf{w}\cdot\mathbf{x}$ is the dot product of weights and binary features in the usual way. We prefer logistic regression over Naive Bayes, since logistic regression is more resistant to possible dependencies between variables. The conditional likelihood maximization in logistic regression will adjust its parameters to maximize the fit even when the resulting parameters are inconsistent with the Naive Bayes assumption. Finally, logistic regression is less sensitive to parameter tuning than SVMs, so to avoid expensive parameter optimization we settled for logistic regression.

To test the importance of case, we did experiments with and without lowercasing of all words. To test the importance of stop words, we did experiments where stop words had been removed from the texts in advance. We also considered models with bigrams and trigrams to test the impact of bigger units of text (context). Finally, we varied the size of the dataset to obtain a learning curve suggesting how our model would perform in the limit.

[3] http://mallet.cs.umass.edu/

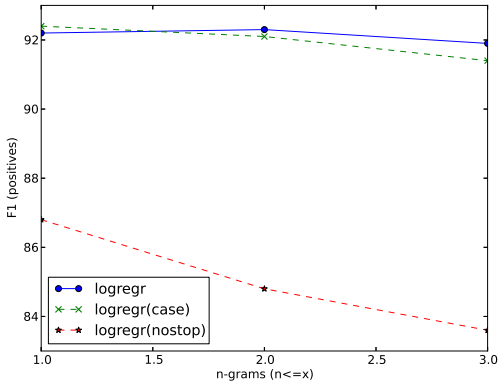

Figure 2: Results with n-grams of different sizes w/o lower-casing and w/o stop words.

4 Results

We report f-scores obtained by 10-fold cross-validation over a balanced 8,000 data points in Figure 2. The green line is our hypothesis model using n-grams of up to different lengths (1, 2 and 3). In this model features are *not* lower-cased (case is preserved), and stop words are *included*. This corresponds to our hypotheses about what would work best for quotation mining. The green line tells us that our unigram model is considerably better than our bigram and trigram models. This is probably because the bigrams and trigrams are too sparsely distributed in our data selection.

The blue line represents results with lowercased features. This means that features will be less sparse, and we now see that the bigram model is slightly better than the unigram model.

The red line represents results where stop words have been removed. This would be a typical model for topic classification. We see that this performs radically worse than the other two models, suggesting that our hypothesis about the usefulness of stop words for quotation mining was correct. The observation that the bigram and trigram models without stop words are much worse than the unigram model without stop words is most likely due to the extra sparsity introduced by open class trigrams.

Our main result is that with sufficient training data the f-score for detecting famous quotes in philosophical and literary text approaches 95%. The learning curves in Figure 3 are the results of our hypothesis

Source	Quote
Bill Clinton's Inaugural 1992	Powerful people maneuver for position and worry endlessly about who is in and who is out, who is up and who is down, forgetting those people whose toil and sweat sends us here and paves our way.
Bill Clinton's Inaugural 1997	But let us never forget : The greatest progress we have made, and the greatest progress we have yet to make, is in the human heart.
PTB CoNLL 2007 test	When the dollar is in a free-fall , even central banks can't stop it .
Europarl 01-17-00	Our citizens can not accept that the European Union takes decisions in a way that is, at least on the face of it, bureaucratic .
Europarl 01-18-00	If competition policy is to be made subordinate to the aims of social and environmental policy , real efficiency and economic growth will remain just a dream .
Europarl 01-19-00	For Europe to become the symbol of peace and fraternity , we need a bold and generous policy to come to the aid of the most disadvantaged .

Figure 4: The sentence with highest probability of being a quote in each corpus according to our 20K logistic regression unigram model).

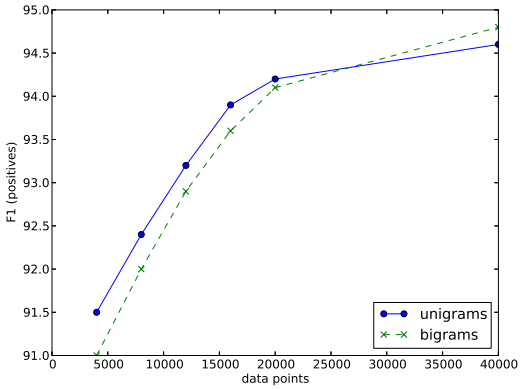

Figure 3: Learning curves for unigram and bigram models without lower-casing and with stop words.

model (green line in Figure 2) obtained with varying amounts of training data, from 4,000 to 40,000 data points. The learning curves also confirm that the bigram model was suffering from sparsity with smaller data selections, and we observe that the bigram model becomes superior to the unigram model with about 30,000 data points. The learning curves show that F-scores for positive class approach 95% as we add more training data.

5 Discussion

To confirm Mooney's hypothesis that it is better to combine weighted evidence from *all* features rather than select a subset of features for early discrimination, also in the case of mining quotes, we ran a decision tree algorithm on the same data sets used above. The f-score for detecting quotes was consistently below 65%.

The decision tree algorithm tries to find good features for early discrimination. Interestingly, one of the most discriminative features picked up by the decision tree from trigram data with case preserved was the bigram 'He who'. This feature was used to split 500 sentences, leaving only 11 in the minority class. Other discriminative features include 'People', 'we are', 'if you have', and 'Nothing is more'.

Similarly, we can observe remarkable differences in marginal distributions by considering the most frequent words in positive and negative texts. Words such as "who", "all", "word", and "things" occur much more frequently in quotes than in more balanced literary philosophical text. Interestingly '–' is also a very good predictor of a sentence being a potential quote.

Finally, we ran a model on other corpora to identify novel candidates of famous quotes (Figure 4). We ran it on texts where you would expect to find potential famous quotes (e.g. inaugurals), as well as on texts where you would not expect that.

6 Conclusion

Simple text classification algorithms perform remarkably well when used for detecting famous quotes in literary or philosophical text, with f-scores approaching 95%. We compare the task to topic classification, polarity classification and authorship attribution and observe that unlike in topic classification, stop words are extremely useful for quotation mining.

References

R Arun, R Saradha, V Suresh, M Murty, and C Madhavan. 2009. Stopwords and stylometry: a latent Dirichlet allocation approach. In *NIPS workshop on Applications for Topic Models*.

Alexandra Balahor, Ralf Steinberger, Erik van der Goot, Bruno Pouliquen, and Mijail Kabadjov. 2009. Opinion mining on newspaper quotations. In *IEEE/WIC/ACM Web Intelligence*.

John Blitzer, Mark Dredze, and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*.

William Cohen, Vitor Carvalho, and Tom Mitchell. 2003. Learning to classify email into "speech acts". In *EMNLP*.

Brett Drury, Gaël Dias, and Luis Torgo. 2011. A contextual classification strategy for polarity analysis of direct quotations from financial news. In *RANLP*.

Anthony Khoo, Yuval Marom, and David Albrecht. 2006. Experiments with sentence classification. In *ALTW*.

Raymond Mooney. 1996. Comparative experiments on disambiguating word senses. In *EMNLP*.

Chao Wang, Jie Lu, and Guangquan Zhang. 2005. A semantic classification approach for online product reviews. In *IEEE/WIC/ACM Web Intelligence*.