# Automatic Satire Detection: Are You Having a Laugh?

**Clint Burfoot**
CSSE
University of Melbourne
VIC 3010 Australia
cburfoot@csse.unimelb.edu.au

**Timothy Baldwin**
CSSE
University of Melbourne
VIC 3010 Australia
tim@csse.unimelb.edu.au

## Abstract

We introduce the novel task of determining whether a newswire article is "true" or satirical. We experiment with SVMs, feature scaling, and a number of lexical and semantic feature types, and achieve promising results over the task.

## 1 Introduction

This paper describes a method for filtering satirical news articles from true newswire documents. We define a satirical article as one which deliberately exposes real-world individuals, organisations and events to ridicule.

Satirical news articles tend to mimic true newswire articles, incorporating irony and non sequitur in an attempt to provide humorous insight. An example excerpt is:

> Bank Of England Governor Mervyn King is a Queen, Says Fed Chairman Ben Bernanke
>
> During last night's appearance on the American David Letterman Show, Fed Chairman Ben Bernanke let slip that Bank of England (BOE) Governor, Mervyn King, enjoys wearing women's clothing.

Contrast this with a snippet of a true newswire article:

> Delegates prepare for Cairo conference amid tight security
>
> Delegates from 156 countries began preparatory talks here Saturday ahead of the official opening of the UN World Population Conference amid tight security.

The basis for our claim that the first document is satirical is surprisingly subtle in nature, and relates to the absurdity of the suggestion that a prominent figure would expose another prominent figure as a cross dresser, the implausibility of this story appearing in a reputable news source, and the pun on the name (*King* being a *Queen*).

Satire classification is a novel task to computational linguistics. It is somewhat similar to the more widely-researched text classification tasks of spam filtering (Androutsopoulos et al., 2000) and sentiment classification (Pang and Lee, 2008), in that: (a) it is a binary classification task, and (b) it is an intrinsically semantic task, i.e. satire news articles are recognisable as such through interpretation and cross-comparison to world knowledge about the entities involved. Similarly to spam filtering and sentiment classification, a key question asked in this research is whether it is possible to perform the task on the basis of simple lexical features of various types. That is, is it possible to automatically detect satire without access to the complex inferencing and real-world knowledge that humans make use of.

The primary contributions of this research are as follows: (1) we introduce a novel task to the arena of computational linguistics and machine learning, and make available a standardised dataset for research on satire detection; and (2) we develop a method which is adept at identifying satire based on simple bag-of-words features, and further extend it to include richer features.

## 2 Corpus

Our satire corpus consists of a total of 4000 newswire documents and 233 satire news articles, split into fixed training and test sets as detailed in Table 1. The newswire documents were randomly sampled from the English Gigaword Corpus. The satire documents were selected to relate closely to at least one of the newswire documents by: (1) randomly selecting a newswire document; (2) hand-picking a key individual, institution or event from the selected document, and using it to formulate a phrasal query (e.g. *Bill Clinton*); (3) using the query to issue a site-restricted query to the



*Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*, pages 161–164,
Suntec, Singapore, 4 August 2009. 

| | Training | Test | Total |
|---|---|---|---|
| TRUE | 2505 | 1495 | 4000 |
| SATIRE | 133 | 100 | 233 |

Table 1: Corpus statistics

Google search engine;[1] and (4) manually filtering out "non-newsy", irrelevant and overly-offensive documents from the top-10 returned documents (i.e. documents not containing satire news articles, or containing satire articles which were not relevant to the original query). All newswire and satire documents were then converted to plain text of consistent format using `lynx`, and all content other than the title and body of the article was manually removed (including web page menus, and header and footer data). Finally, all documents were manually post-edited to remove references to the source (e.g. *AP* or *Onion*), formatting quirks specific to a particular source (e.g. all caps in the title), and any textual metadata which was indicative of the document source (e.g. editorial notes, dates and locations). This was all in an effort to prevent classifiers from accessing superficial features which are reliable indicators of the document source and hence trivialise the satire detection process.

It is important to note that the number of satirical news articles in the corpus is significantly less than the number of true newswire articles. This reflects an impressionistic view of the web: there is far more true news content than satirical news content.

The corpus is novel to this research, and is publicly available for download at `http://www.csse.unimelb.edu.au/research/lt/resources/satire/`.

## 3 Method

### 3.1 Standard text classification approach

We take our starting point from topic-based text classification (Dumais et al., 1998; Joachims, 1998) and sentiment classification (Turney, 2002; Pang and Lee, 2008). State-of-the-art results in both fields have been achieved using support vector machines (SVMs) and bag-of-words features. We supplement the bag-of-words model with feature weighting, using the two methods described below.

**Binary feature weights:** Under this scheme all features are given the same weight, regardless of how many times they appear in each article. The topic and sentiment classification examples cited found binary features gave better performance than other alternatives.

**Bi-normal separation feature scaling:** BNS (Forman, 2008) has been shown to outperform other established feature representation schemes on a wide range of text classification tasks. This superiority is especially pronounced for collections with a low proportion of positive class instances. Under BNS, features are allocated a weight according to the formula:

$$|\mathrm{F}^{-1}(tpr) - \mathrm{F}^{-1}(fpr)|$$

where $\mathrm{F}^{-1}$ is the inverse normal cumulative distribution function, $tpr$ is the true positive rate (P(feature|positive class)) and $fpr$ is the false positive rate (P(feature|negative class)).

BNS produces the highest weights for features that are strongly correlated with either the negative or positive class. Features that occur evenly across the training instances are given the lowest weight. This behaviour is particularly helpful for features that correlate with the negative class in a negatively-skewed classification task, so in our case BNS should assist the classifier in making use of features that identify true articles.

SVM classification is performed with SVM$^{light}$ (Joachims, 1999) using a linear kernel and the default parameter settings. Tokens are case folded; currency amounts (e.g. $2.50), abbreviations (e.g. U.S.A.), and punctuation sequences (e.g. a comma, or a closing quote mark followed by a period) are treated as separate features.

### 3.2 Targeted lexical features

This section describe three types of features intended to embody characteristics of satire news documents.

**Headline features:** Most of the articles in the corpus have a headline as their first line. To a human reader, the vast majority of the satire documents in our corpus are immediately recognisable as such from the headline alone, suggesting that our classifiers may get something out of having the

[1] The sites queried were `satirewire.com`, `theonion.com`, `newsgroper.com`, `thespoof.com`, `brokennewz.com`, `thetoque.com`, `bbspot.com`, `neowhig.org`, `humorfeed.com`, `satiricalmuslim.com`, `yunews.com`, `newsbiscuit.com`.

headline contents explicitly identified in the feature vector. To this end, we add an additional feature for each unigram appearing on the first line of an article. In this way the heading tokens are represented twice: once in the overall set of unigrams in the article, and once in the set of heading unigrams.

**Profanity:** true news articles very occasionally include a verbal quote which contains offensive language, but in practically all other cases it is incumbent on journalists and editors to keep their language "clean". A review of the corpus shows that this is not the case with satirical news, which occasionally uses profanity as a humorous device.

Let *P* be a binary feature indicating whether or not an article contains profanity, as determined by the `Regexp::Common::profanity` Perl module.[2]

**Slang:** As with profanity, it is intuitively true that true news articles tend to avoid slang. An impressionistic review of the corpus suggests that informal language is much more common to satirical articles. We measure the informality of an article as:

$$i \stackrel{\text{def}}{=} \frac{1}{|T|} \sum_{t \in T} s(t)$$

where $T$ is the set of unigram tokens in the article and $s$ is a function taking the value 1 if the token has a dictionary definition marked as slang and 0 if it does not.

It is important to note that this measure of "informality" is approximate at best. We do not attempt, e.g., to disambiguate the sense of individual word terms to tell whether the slang sense of a word is the one intended. Rather, we simply check to see if each word has a slang usage in Wiktionary.[3]

A continuous feature is set to the value of $i$ for each article. Discrete features *highi* and *lowi* are set as:

$$\textit{highi} \stackrel{\text{def}}{=} \begin{cases} 1 & v > \bar{i} + 2\sigma; \\ 0 & \end{cases}$$

$$\textit{lowi} \stackrel{\text{def}}{=} \begin{cases} 1 & v < \bar{i} - 2\sigma; \\ 0 & \end{cases}$$

where $\bar{i}$ and $\sigma$ are, respectively, the mean and standard deviation of $i$ across all articles.

[2] `http://search.cpan.org/perldoc?Regexp::Common::profanity`

[3] `http://www.wiktionary.org`

### 3.3 Semantic validity

Lexical approaches are clearly inadequate if we assume that good satirical news articles tend to emulate real news in tone, style, and content. What is needed is an approach that captures the document semantics.

One common device in satire news articles is absurdity, in terms of describing well-known individuals in unfamiliar settings which parody their viewpoints or public profile. We attempt to capture this via *validity*, in the form of the relative frequency of the particular combination of key participants reported in the story. Our method identifies the named entities in a given document and queries the web for the conjunction of those entities. Our expectation is that true news stories will have been reported in various forums, and hence the number of web documents which include the same combination of entities will be higher than with satire documents.

To implement this method, we first use the Stanford Named Entity Recognizer[4] (Finkel et al., 2005) to identify the set of person and organisation entities, $E$, from each article in the corpus.

From this, we estimate the validity of the combination of entities in the article as:

$$v(E) \stackrel{\text{def}}{=} |g(E)|$$

where $g$ is the set of matching documents returned by Google using a conjunctive query. We anticipate that $v$ will have two potentially useful properties: (1) it will be relatively lower when $E$ includes made-up entity names such as *Hitler Commemoration Institute*, found in one satirical corpus article; and (2) it will be relatively lower when $E$ contains unusual combinations of entities such as, for example, those in the satirical article beginning *Missing Brazilian balloonist Padre spotted straddling Pink Floyd flying pig*.

We include both a continuous representation of $v$ for each article, in the form of $\log(v(E))$, and discrete variants of the feature, based on the same methodology as for *highi* and *lowi*.

## 4 Results

The results for our classifiers over the satire corpus are shown in Table 2. The baseline is a naive classifier that assigns all instances to the positive

[4] `http://nlp.stanford.edu/software/CRF-NER.shtml`

| ("article⇒SATIRE?") | P | R | F |
|---|---|---|---|
| all-positive baseline | 0.063 | **1.000** | 0.118 |
| BIN | 0.943 | 0.500 | 0.654 |
| BIN+lex | 0.945 | 0.520 | 0.671 |
| BIN+val | 0.943 | 0.500 | 0.654 |
| BIN+all | 0.945 | 0.520 | 0.671 |
| BNS | 0.944 | 0.670 | 0.784 |
| BNS+lex | 0.957 | 0.660 | 0.781 |
| BNS+val | 0.945 | 0.690 | **0.798** |
| BNS+all | **0.958** | 0.680 | 0.795 |

Table 2: Results for satire detection (P = precision, R = recall, and F = F-score) for binary unigram features (BIN) and BNS unigram features (BNS), optionally using lexical (lex), validity (val) or combined lexical and validity (all) features

class (i.e. SATIRE). An SVM classifier with simple binary unigram word features provides a standard text classification benchmark.

All of the classifiers easily outperform the baseline. This is to be expected given the low proportion of positive instances in the corpus. The benchmark classifier has very good precision, but recall of only 0.500. Adding the heading, slang, and profanity features provides a small improvement in both precision and recall.

Moving to BNS feature scaling keeps the very high precision and increases the recall to 0.670. Adding in the heading, slang and profanity lexical features ("+lex") actually decreases the F-score slightly, but adding the validity features ("+val") provides a near 2 point F-score increase, resulting in the best overall F-score of 0.798.

All of the BNS scores achieve statistically significant improvements over the benchmark in terms of F-score (using approximate randomisation, $p < 0.05$). The 1-2% gains given by adding in the various feature types are not statistically significant due to the small number of satire instances concerned.

All of the classifiers achieve very high precision and considerably lower recall. Error analysis suggests that the reason for the lower recall is subtler satire articles, which require detailed knowledge of the individuals to be fully appreciated as satire. While they are not perfect, however, the classifiers achieve remarkably high performance given the superficiality of the features used.

## 5 Conclusions and future work

This paper has introduced a novel task to computational linguistics and machine learning: determining whether a newswire article is "true" or satirical. We found that the combination of SVMs with BNS feature scaling achieves high precision and lower recall, and that the inclusion of the notion of "validity" achieves the best overall F-score.

## References


Ion Androutsopoulos, John Koutsias, Konstantinos V. Chandrinos, George Paliouras, and Constantine D. Spyropoulos. 2000. An evaluation of Naive Bayesian anti-spam filtering. In *Proceedings of the 11th European Conference on Machine Learning*, pages 9–17, Barcelona, Spain.

Susan Dumais, John Platt, David Heckerman, and Mehran Sahami. 1998. Inductive learning algorithms and representations for text categorization. In *Proceedings of the Seventh International Conference on Information and Knowledge Management*, pages 148–155, New York, USA.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 363–370, Ann Arbor, USA.

George Forman. 2008. BNS scaling: An improved representation over TF-IDF for SVM text classification. In *Proceedings of the 17th International Conference on Information and Knowledge Management*, pages 263–270, Napa Valley, USA.

Thorsten Joachims. 1998. Text categorization with support vector machines: learning with many relevant features. In *Proceedings of the 10th European Conference on Machine Learning*, pages 137–142, Chemnitz, Germany.

Thorsten Joachims. 1999. Making large-scale support vector machine learning practical. In Bernhard Schölkopf, Christopher J. C. Burges, and Alexander J. Smola, editors, *Advances in Kernel Methods: Support Vector Learning*, pages 169–184. MIT Press, Cambridge, USA.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2):1–135.

Peter Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424, Philadelphia, USA.