Making Computers Laugh: Investigations in Automatic Humor Recognition

Rada Mihalcea
Department of Computer Science
University of North Texas
Denton, TX, 76203, USA
rada@cs.unt.edu

Carlo Strapparava
Istituto per la Ricerca Scientifica e Tecnologica
ITC – irst
I-38050, Povo, Trento, Italy
strappa@itc.it

Abstract

Humor is one of the most interesting and puzzling aspects of human behavior. Despite the attention it has received in fields such as philosophy, linguistics, and psychology, there have been only few attempts to create computational models for humor recognition or generation. In this paper, we bring empirical evidence that computational approaches can be successfully applied to the task of humor recognition. Through experiments performed on very large data sets, we show that automatic classification techniques can be effectively used to distinguish between humorous and non-humorous texts, with significant improvements observed over apriori known baselines.

1 Introduction

> *... pleasure has probably been the main goal all along. But I hesitate to admit it, because computer scientists want to maintain their image as hard-working individuals who deserve high salaries. Sooner or later society will realize that certain kinds of hard work are in fact admirable even though they are more fun than just about anything else. (Knuth, 1993)*

Humor is an essential element in personal communication. While it is merely considered a way to induce amusement, humor also has a positive effect on the mental state of those using it and has the ability to improve their activity. Therefore computational humor deserves particular attention, as it has the potential of changing computers into a creative and motivational tool for human activity (Stock et al., 2002; Nijholt et al., 2003).

Previous work in computational humor has focused mainly on the task of humor generation (Stock and Strapparava, 2003; Binsted and Ritchie, 1997), and very few attempts have been made to develop systems for automatic humor recognition (Taylor and Mazlack, 2004). This is not surprising, since, from a computational perspective, humor recognition appears to be significantly more subtle and difficult than humor generation.

In this paper, we explore the applicability of computational approaches to the recognition of verbally expressed humor. In particular, we investigate whether automatic classification techniques are a viable approach to distinguish between humorous and non-humorous text, and we bring empirical evidence in support of this hypothesis through experiments performed on very large data sets.

Since a deep comprehension of humor in all of its aspects is probably too ambitious and beyond the existing computational capabilities, we chose to restrict our investigation to the type of humor found in *one-liners*. A one-liner is a short sentence with comic effects and an interesting linguistic structure: simple syntax, deliberate use of rhetoric devices (e.g. alliteration, rhyme), and frequent use of creative language constructions meant to attract the readers attention. While longer jokes can have a relatively complex narrative structure, a one-liner must produce the humorous effect “in one shot”, with very few words. These characteristics make this type of humor particularly suitable for use in an automatic learning setting, as the humor-producing features are guaranteed to be present in the first (and only) sentence.

We attempt to formulate the humor-recognition

problem as a traditional classification task, and feed positive (humorous) and negative (non-humorous) examples to an automatic classifier. The humorous data set consists of one-liners collected from the Web using an automatic bootstrapping process. The non-humorous data is selected such that it is structurally and stylistically similar to the one-liners. Specifically, we use three different negative data sets: (1) Reuters news titles; (2) proverbs; and (3) sentences from the British National Corpus (BNC). The classification results are encouraging, with accuracy figures ranging from 79.15% (One-liners/BNC) to 96.95% (One-liners/Reuters). Regardless of the non-humorous data set playing the role of negative examples, the performance of the automatically learned humor-recognizer is always significantly better than apriori known baselines.

The remainder of the paper is organized as follows. We first describe the humorous and non-humorous data sets, and provide details on the Web-based bootstrapping process used to build a very large collection of one-liners. We then show experimental results obtained on these data sets using several heuristics and two different text classifiers. Finally, we conclude with a discussion and directions for future work.

2 Humorous and Non-humorous Data Sets

To test our hypothesis that automatic classification techniques represent a viable approach to humor recognition, we needed in the first place a data set consisting of both humorous (positive) and non-humorous (negative) examples. Such data sets can be used to automatically *learn* computational models for humor recognition, and at the same time *evaluate* the performance of such models.

2.1 Humorous Data

For reasons outlined earlier, we restrict our attention to one-liners, short humorous sentences that have the characteristic of producing a comic effect in very few words (usually 15 or less). The one-liners humor style is illustrated in Table 1, which shows three examples of such one-sentence jokes.

It is well-known that large amounts of training data have the potential of improving the accuracy of the learning process, and at the same time provide insights into how increasingly larger data sets can affect the classification precision. The manual construction of a very large one-liner data set may be however problematic, since most Web sites or mailing lists that make available such jokes do not usually list more than 50–100 one-liners. To tackle this problem, we implemented a Web-based bootstrapping algorithm able to automatically collect a large number of one-liners starting with a short *seed* list, consisting of a few one-liners manually identified.

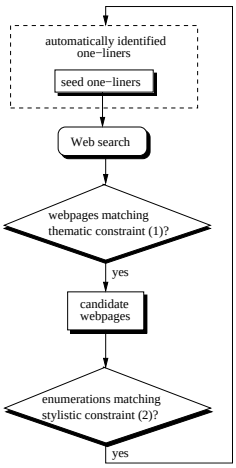


Figure 1: Web-based bootstrapping of one-liners.

The bootstrapping process is illustrated in Figure 1. Starting with the seed set, the algorithm automatically identifies a list of webpages that include at least one of the seed one-liners, via a simple search performed with a Web search engine. Next, the webpages found in this way are HTML parsed, and additional one-liners are automatically identified and added to the seed set. The process is repeated several times, until enough one-liners are collected.

An important aspect of any bootstrapping algorithm is the set of constraints used to steer the process and prevent as much as possible the addition of noisy entries. Our algorithm uses: (1) a *thematic* constraint applied to the theme of each webpage; and (2) a *structural* constraint, exploiting HTML annotations indicating text of similar genre.

The first constraint is implemented using a set of keywords of which at least one has to appear in the URL of a retrieved webpage, thus potentially limiting the content of the webpage to a theme related to that keyword. The set of keywords used in the current implementation consists of six words that explicitly indicate humor-related content: *oneliner*, *one-liner*, *humor*, *humour*, *joke*,

One-liners
Take my advice; I don't use it anyway.
I get enough exercise just pushing my luck.
Beauty is in the eye of the beer holder.
Reuters titles
Trocadero expects tripling of revenues.
Silver fixes at two-month high, but gold lags.
Oil prices slip as refiners shop for bargains.
BNC sentences
They were like spirits, and I loved them.
I wonder if there is some contradiction here.
The train arrives three minutes early.
Proverbs
Creativity is more important than knowledge.
Beauty is in the eye of the beholder.
I believe no tales from an enemy's tongue.

Table 1: Sample examples of one-liners, Reuters titles, BNC sentences, and proverbs.

funny. For example, *http://www.berro.com/Jokes* or *http://www.mutedfaith.com/funny/life.htm* are the URLs of two webpages that satisfy this constraint.

The second constraint is designed to exploit the HTML structure of webpages, in an attempt to identify enumerations of texts that include the seed one-liner. This is based on the hypothesis that enumerations typically include texts of similar genre, and thus a list including the seed one-liner is likely to include additional one-line jokes. For instance, if a seed one-liner is found in a webpage preceded by the HTML tag `<li>` (i.e. "list item"), other lines found in the same enumeration preceded by the same tag are also likely to be one-liners.

Two iterations of the bootstrapping process, started with a small seed set of ten one-liners, resulted in a large set of about 24,000 one-liners. After removing the duplicates using a measure of string similarity based on the longest common subsequence metric, we were left with a final set of approximately 16,000 one-liners, which are used in the humor-recognition experiments. Note that since the collection process is automatic, noisy entries are also possible. Manual verification of a randomly selected sample of 200 one-liners indicates an average of 9% potential noise in the data set, which is within reasonable limits, as it does not appear to significantly impact the quality of the learning.

2.2 Non-humorous Data

To construct the set of negative examples required by the humor-recognition models, we tried to identify collections of sentences that were non-humorous, but similar in structure and composition to the one-liners. We do not want the automatic classifiers to learn to distinguish between humorous and non-humorous examples based simply on text length or obvious vocabulary differences. Instead, we seek to enforce the classifiers to identify humor-specific features, by supplying them with negative examples similar in most of their aspects to the positive examples, but different in their comic effect.

We tested three different sets of negative examples, with three examples from each data set illustrated in Table 1. All non-humorous examples are enforced to follow the same length restriction as the one-liners, i.e. one sentence with an average length of 10–15 words.

1. *Reuters* titles, extracted from news articles published in the Reuters newswire over a period of one year (8/20/1996 – 8/19/1997) (Lewis et al., 2004). The titles consist of short sentences with simple syntax, and are often phrased to catch the readers attention (an effect similar to the one rendered by one-liners).
2. *Proverbs* extracted from an online proverb collection. Proverbs are sayings that transmit, usually in one short sentence, important facts or experiences that are considered true by many people. Their property of being condensed, but memorable sayings make them very similar to the one-liners. In fact, some one-liners attempt to reproduce proverbs, with a comic effect, as in e.g. *"Beauty is in the eye of the beer holder"*, derived from *"Beauty is in the eye of the beholder"*.
3. *British National Corpus (BNC)* sentences, extracted from BNC – a balanced corpus covering different styles, genres and domains. The sentences were selected such that they were similar in content with the one-liners: we used an information retrieval system implementing a vectorial model to identify the BNC sentence most similar to each of the 16,000 one-liners[1]. Unlike the Reuters titles or the proverbs, the BNC sentences have typically no added creativity. However, we decided to add this set of negative examples to our experimental setting, in order

[1]The sentence most similar to a one-liner is identified by running the one-liner against an index built for all BNC sentences with a length of 10–15 words. We use a *tf.idf* weighting scheme and a cosine similarity measure, as implemented in the Smart system (*ftp.cs.cornell.edu/pub/smart*)

to observe the level of difficulty of a humor-recognition task when performed with respect to simple text.

To summarize, the humor recognition experiments rely on data sets consisting of humorous (positive) and non-humorous (negative) examples. The positive examples consist of 16,000 one-liners automatically collected using a Web-based bootstrapping process. The negative examples are drawn from: (1) Reuters titles; (2) Proverbs; and (3) BNC sentences.

3 Automatic Humor Recognition

We experiment with automatic classification techniques using: (a) heuristics based on humor-specific stylistic features (alliteration, antonymy, slang); (b) content-based features, within a learning framework formulated as a typical text classification task; and (c) combined stylistic and content-based features, integrated in a stacked machine learning framework.

3.1 Humor-Specific Stylistic Features

Linguistic theories of humor (Attardo, 1994) have suggested many *stylistic features* that characterize humorous texts. We tried to identify a set of features that were both significant and feasible to implement using existing machine readable resources. Specifically, we focus on alliteration, antonymy, and adult slang, which were previously suggested as potentially good indicators of humor (Ruch, 2002; Bucaria, 2004).

Alliteration. Some studies on humor appreciation (Ruch, 2002) show that structural and phonetic properties of jokes are at least as important as their content. In fact one-liners often rely on the reader's awareness of attention-catching sounds, through linguistic phenomena such as alliteration, word repetition and rhyme, which produce a comic effect even if the jokes are not necessarily meant to be read aloud. Note that similar rhetorical devices play an important role in wordplay jokes, and are often used in newspaper headlines and in advertisement. The following one-liners are examples of jokes that include one or more alliteration chains:

Veni, Vidi, Visa: I came, I saw, I did a little shopping.
Infants don't enjoy infancy like adults do adultery.

To extract this feature, we identify and count the number of alliteration/rhyme chains in each example in our data set. The chains are automatically extracted using an index created on top of the CMU pronunciation dictionary[2].

Antonymy. Humor often relies on some type of incongruity, opposition or other forms of apparent contradiction. While an accurate identification of all these properties is probably difficult to accomplish, it is relatively easy to identify the presence of *antonyms* in a sentence. For instance, the comic effect produced by the following one-liners is partly due to the presence of antonyms:

A clean desk is a sign of a cluttered desk drawer.
Always try to be modest and be proud of it!

The lexical resource we use to identify antonyms is WORDNET (Miller, 1995), and in particular the *antonymy* relation among nouns, verbs, adjectives and adverbs. For adjectives we also consider an indirect antonymy via the *similar-to* relation among adjective synsets. Despite the relatively large number of *antonymy* relations defined in WORDNET, its coverage is far from complete, and thus the *antonymy* feature cannot always be identified. A deeper semantic analysis of the text, such as word sense disambiguation or domain disambiguation, could probably help detecting other types of semantic opposition, and we plan to exploit these techniques in future work.

Adult slang. Humor based on adult slang is very popular. Therefore, a possible feature for humor-recognition is the detection of sexual-oriented lexicon in the sentence. The following represent examples of one-liners that include such slang:

The sex was so good that even the neighbors had a cigarette.
Artificial Insemination: procreation without recreation.

To form a lexicon required for the identification of this feature, we extract from WORDNET DOMAINS[3] all the synsets labeled with the domain SEXUALITY. The list is further processed by removing all words with high polysemy (≥ 4). Next, we check for the presence of the words in this lexicon in each sentence in the corpus, and annotate them accordingly. Note that, as in the case of antonymy, WORDNET coverage is not complete, and the *adult slang* feature cannot always be identified.

Finally, in some cases, all three features (alliteration,

[2] Available at *http://www.speech.cs.cmu.edu/cgi-bin/cmudict*

[3] WORDNET DOMAINS assigns each synset in WORDNET with one or more "domain" labels, such as SPORT, MEDICINE, ECONOMY. See *http://wndomains.itc.it*.

antonymy, adult slang) are present in the same sentence, as for instance the following one-liner:

Behind every great$_{al}$ man$_{ant}$ is a great$_{al}$ woman$_{ant}$, and behind every great$_{al}$ woman$_{ant}$ is some guy staring at her behind$_{sl}$!

3.2 Content-based Learning

In addition to stylistic features, we also experimented with *content-based features*, through experiments where the humor-recognition task is formulated as a traditional text classification problem. Specifically, we compare results obtained with two frequently used text classifiers, Naïve Bayes and Support Vector Machines, selected based on their performance in previously reported work, and for their diversity of learning methodologies.

Naïve Bayes. The main idea in a Naïve Bayes text classifier is to estimate the probability of a category given a document using joint probabilities of words and documents. Naïve Bayes classifiers assume word independence, but despite this simplification, they perform well on text classification. While there are several versions of Naïve Bayes classifiers (variations of multinomial and multivariate Bernoulli), we use the multinomial model, previously shown to be more effective (McCallum and Nigam, 1998).

Support Vector Machines. Support Vector Machines (SVM) are binary classifiers that seek to find the hyperplane that best separates a set of positive examples from a set of negative examples, with maximum margin. Applications of SVM classifiers to text categorization led to some of the best results reported in the literature (Joachims, 1998).

4 Experimental Results

Several experiments were conducted to gain insights into various aspects related to an automatic humor recognition task: classification accuracy using stylistic and content-based features, learning rates, impact of the type of negative data, impact of the classification methodology.

All evaluations are performed using stratified ten-fold cross validations, for accurate estimates. The baseline for all the experiments is 50%, which represents the classification accuracy obtained if a label of "humorous" (or "non-humorous") would be assigned by default to all the examples in the data set. Experiments with uneven class distributions were also performed, and are reported in section 4.4.

4.1 Heuristics using Humor-specific Features

In a first set of experiments, we evaluated the classification accuracy using stylistic humor-specific features: alliteration, antonymy, and adult slang. These are numerical features that act as heuristics, and the only parameter required for their application is a threshold indicating the minimum value admitted for a statement to be classified as humorous (or non-humorous). These thresholds are learned automatically using a decision tree applied on a small subset of humorous/non-humorous examples (1000 examples). The evaluation is performed on the remaining 15,000 examples, with results shown in Table 2[4].

Heuristic	One-liners Reuters	One-liners BNC	One-liners Proverbs
Alliteration	74.31%	59.34%	53.30%
Antonymy	55.65%	51.40%	50.51%
Adult slang	52.74%	52.39%	50.74%
ALL	76.73%	60.63%	53.71%

Table 2: Humor-recognition accuracy using alliteration, antonymy, and adult slang.

Considering the fact that these features represent *stylistic* indicators, the style of Reuters titles turns out to be the most different with respect to one-liners, while the style of proverbs is the most similar. Note that for all data sets the alliteration feature appears to be the most useful indicator of humor, which is in agreement with previous linguistic findings (Ruch, 2002).

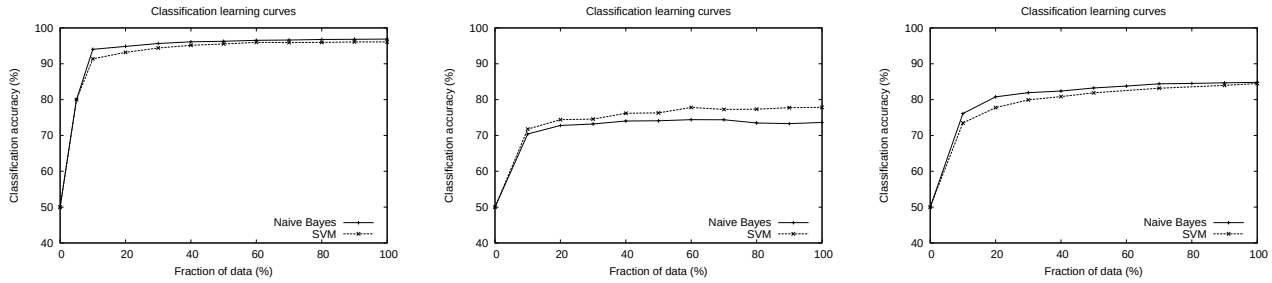
4.2 Text Classification with Content Features

The second set of experiments was concerned with the evaluation of content-based features for humor recognition. Table 3 shows results obtained using the three different sets of negative examples, with the Naïve Bayes and SVM text classifiers. Learning curves are plotted in Figure 2.

Classifier	One-liners Reuters	One-liners BNC	One-liners Proverbs
Naïve Bayes	96.67%	73.22%	84.81%
SVM	96.09%	77.51%	84.48%

Table 3: Humor-recognition accuracy using Naïve Bayes and SVM text classifiers.

[4]We also experimented with decision trees learned from a larger number of examples, but the results were similar, which confirms our hypothesis that these features are heuristics, rather than learnable properties that improve their accuracy with additional training data.



(a) (b) (c)

Figure 2: Learning curves for humor-recognition using text classification techniques, with respect to three different sets of negative examples: (a) Reuters; (b) BNC; (c) Proverbs.

Once again, the content of Reuters titles appears to be the most different with respect to one-liners, while the BNC sentences represent the most similar data set. This suggests that joke content tends to be very similar to regular text, although a reasonably accurate distinction can still be made using text classification techniques. Interestingly, proverbs can be distinguished from one-liners using content-based features, which indicates that despite their stylistic similarity (see Table 2), proverbs and one-liners deal with different topics.

4.3 Combining Stylistic and Content Features

Encouraged by the results obtained in the first two experiments, we designed a third experiment that attempts to jointly exploit stylistic and content features for humor recognition. The feature combination is performed using a stacked learner, which takes the output of the text classifier, joins it with the three humor-specific features (alliteration, antonymy, adult slang), and feeds the newly created feature vectors to a machine learning tool. Given the relatively large gap between the performance achieved with content-based features (text classification) and stylistic features (humor-specific heuristics), we decided to implement the second learning stage in the stacked learner using a memory based learning system, so that low-performance features are not eliminated in the favor of the more accurate ones[5]. We use the Timbl memory based learner (Daelemans et al., 2001), and evaluate the classification using a stratified ten-fold cross validation. Table 4 shows the results obtained in this experiment, for the three different data sets.

One-liners Reuters	One-liners BNC	One-liners Proverbs
96.95%	79.15%	84.82%

Table 4: Humor-recognition accuracy for combined learning based on stylistic and content features.

Combining classifiers results in a statistically significant improvement ($p < 0.0005$, paired t-test) with respect to the best individual classifier for the One-liners/Reuters and One-liners/BNC data sets, with relative error rate reductions of 8.9% and 7.3% respectively. No improvement is observed for the One-liners/Proverbs data set, which is not surprising since, as shown in Table 2, proverbs and one-liners cannot be clearly differentiated using stylistic features, and thus the addition of these features to content-based features is not likely to result in an improvement.

4.4 Discussion

The results obtained in the automatic classification experiments reveal the fact that computational approaches represent a viable solution for the task of humor-recognition, and good performance can be achieved using classification techniques based on stylistic and content features.

Despite our initial intuition that one-liners are most similar to other creative texts (e.g. Reuters titles, or the sometimes almost identical proverbs), and thus the learning task would be more difficult in relation to these data sets, comparative experimental results show that in fact it is more difficult to distinguish humor with respect to regular text (e.g. BNC

[5]Using a decision tree learner in a similar stacked learning experiment resulted into a flat tree that takes a classification decision based exclusively on the content feature, ignoring completely the remaining stylistic features.

sentences). Note however that even in this case the combined classifier leads to a classification accuracy that improves significantly over the apriori known baseline.

An examination of the content-based features learned during the classification process reveals interesting aspects of the humorous texts. For instance, one-liners seem to constantly make reference to human-related scenarios, through the frequent use of words such as *man*, *woman*, *person*, *you*, *I*. Similarly, humorous texts seem to often include negative word forms, such as the negative verb forms *doesn't*, *isn't*, *don't*, or negative adjectives like *wrong* or *bad*. A more extensive analysis of content-based humor-specific features is likely to reveal additional humor-specific content features, which could also be used in studies of humor generation.

In addition to the three negative data sets, we also performed an experiment using a corpus of arbitrary sentences randomly drawn from the three negative sets. The humor recognition with respect to this negative mixed data set resulted in 63.76% accuracy for stylistic features, 77.82% for content-based features using Naïve Bayes and 79.23% using SVM. These figures are comparable to those reported in Tables 2 and 3 for One-liners/BNC, which suggests that the experimental results reported in the previous sections do not reflect a bias introduced by the negative data sets, since similar results are obtained when the humor recognition is performed with respect to arbitrary negative examples.

As indicated in section 2.2, the negative examples were selected structurally and stylistically similar to the one-liners, making the humor recognition task more difficult than in a real setting. Nonetheless, we also performed a set of experiments where we made the task even harder, using uneven class distributions. For each of the three types of negative examples, we constructed a data set using 75% non-humorous examples and 25% humorous examples. Although the baseline in this case is higher (75%), the automatic classification techniques for humor-recognition still improve over this baseline. The stylistic features lead to a classification accuracy of 87.49% (One-liners/Reuters), 77.62% (One-liners/BNC), and 76.20% (One-liners/Proverbs), and the content-based features used in a Naïve Bayes classifier result in accuracy figures of 96.19% (One-liners/Reuters), 81.56% (One-liners/BNC), and 87.86% (One-liners/Proverbs).

Finally, in addition to classification accuracy, we were also interested in the variation of classification performance with respect to data size, which is an aspect particularly relevant for directing future research. Depending on the shape of the learning curves, one could decide to concentrate future work either on the acquisition of larger data sets, or toward the identification of more sophisticated features. Figure 2 shows that regardless of the type of negative data, there is significant learning only until about 60% of the data (i.e. about 10,000 positive examples, and the same number of negative examples). The rather steep ascent of the curve, especially in the first part of the learning, suggests that humorous and non-humorous texts represent well distinguishable types of data. An interesting effect can be noticed toward the end of the learning, where for both classifiers the curve becomes completely flat (One-liners/Reuters, One-liners/Proverbs), or it even has a slight drop (One-liners/BNC). This is probably due to the presence of noise in the data set, which starts to become visible for very large data sets[6]. This plateau is also suggesting that more data is not likely to help improve the quality of an automatic humor-recognizer, and more sophisticated features are probably required.

5 Related Work

While humor is relatively well studied in scientific fields such as linguistics (Attardo, 1994) and psychology (Freud, 1905; Ruch, 2002), to date there is only a limited number of research contributions made toward the construction of computational humour prototypes.

One of the first attempts is perhaps the work described in (Binsted and Ritchie, 1997), where a formal model of semantic and syntactic regularities was devised, underlying some of the simplest types of puns (*punning riddles*). The model was then exploited in a system called JAPE that was able to automatically generate amusing puns.

Another humor-generation project was the HAHAcronym project (Stock and Strapparava, 2003), whose goal was to develop a system able to automatically generate humorous versions of existing

[6]We also like to think of this behavior as if the computer is losing its sense of humor after an overwhelming number of jokes, in a way similar to humans when they get bored and stop appreciating humor after hearing too many jokes.

acronyms, or to produce a new amusing acronym constrained to be a valid vocabulary word, starting with concepts provided by the user. The comic effect was achieved mainly by exploiting incongruity theories (e.g. finding a religious variation for a technical acronym).

Another related work, devoted this time to the problem of humor comprehension, is the study reported in (Taylor and Mazlack, 2004), focused on a very restricted type of wordplays, namely the "Knock-Knock" jokes. The goal of the study was to evaluate to what extent wordplay can be automatically identified in "Knock-Knock" jokes, and if such jokes can be reliably recognized from other non-humorous text. The algorithm was based on automatically extracted structural patterns and on heuristics heavily based on the peculiar structure of this particular type of jokes. While the wordplay recognition gave satisfactory results, the identification of jokes containing such wordplays turned out to be significantly more difficult.

6 Conclusion

A conclusion is simply the place where you got tired of thinking. (anonymous one-liner)

The creative genres of natural language have been traditionally considered outside the scope of any computational modeling. In particular humor, because of its puzzling nature, has received little attention from computational linguists. However, given the importance of humor in our everyday life, and the increasing importance of computers in our work and entertainment, we believe that studies related to computational humor will become increasingly important.

In this paper, we showed that automatic classification techniques can be successfully applied to the task of humor-recognition. Experimental results obtained on very large data sets showed that computational approaches can be efficiently used to distinguish between humorous and non-humorous texts, with significant improvements observed over apriori known baselines. To our knowledge, this is the first result of this kind reported in the literature, as we are not aware of any previous work investigating the interaction between humor and techniques for automatic classification.

Finally, through the analysis of learning curves plotting the classification performance with respect to data size, we showed that the accuracy of the automatic humor-recognizer stops improving after a certain number of examples. Given that automatic humor-recognition is a rather understudied problem, we believe that this is an important result, as it provides insights into potentially productive directions for future work. The flattened shape of the curves toward the end of the learning process suggests that rather than focusing on gathering more data, future work should concentrate on identifying more sophisticated humor-specific features, e.g. semantic oppositions, ambiguity, and others. We plan to address these aspects in future work.

References

S. Attardo. 1994. *Linguistic Theory of Humor*. Mouton de Gruyter, Berlin.

K. Binsted and G. Ritchie. 1997. Computational rules for punning riddles. *Humor*, 10(1).

C. Bucaria. 2004. Lexical and syntactic ambiguity as a source of humor. *Humor*, 17(3).

W. Daelemans, J. Zavrel, K. van der Sloot, and A. van den Bosch. 2001. Timbl: Tilburg memory based learner, version 4.0, reference guide. Technical report, University of Antwerp.

S. Freud. 1905. *Der Witz und Seine Beziehung zum Unbewussten*. Deutike, Vienna.

T. Joachims. 1998. Text categorization with Support Vector Machines: learning with many relevant features. In *Proceedings of the European Conference on Machine Learning*.

D.E. Knuth. 1993. *The Stanford Graph Base: A Platform for combinatorial computing*. ACM Press.

D. Lewis, Y. Yang, T. Rose, and F. Li. 2004. RCV1: A new benchmark collection for text categorization research. *The Journal of Machine Learning Research*, 5:361–397.

A. McCallum and K. Nigam. 1998. A comparison of event models for Naive Bayes text classification. In *Proceedings of AAAI-98 Workshop on Learning for Text Categorization*.

G. Miller. 1995. Wordnet: A lexical database. *Communication of the ACM*, 38(11):39–41.

A. Nijholt, O. Stock, A. Dix, and J. Morkes, editors. 2003. *Proceedings of CHI-2003 workshop: Humor Modeling in the Interface*, Fort Lauderdale, Florida.

W. Ruch. 2002. Computers with a personality? lessons to be learned from studies of the psychology of humor. In *Proceedings of the The April Fools Day Workshop on Computational Humour*.

O. Stock and C. Strapparava. 2003. Getting serious about the development of computational humour. In *Proceedings of the 8^{th} International Joint Conference on Artificial Intelligence (IJCAI-03)*, Acapulco, Mexico.

O. Stock, C. Strapparava, and A. Nijholt, editors. 2002. *Proceedings of the The April Fools Day Workshop on Computational Humour*, Trento.

J. Taylor and L. Mazlack. 2004. Computationally recognizing wordplay in jokes. In *Proceedings of CogSci 2004*, Chicago.