Haiyang Sui
Christopher Khoo

School of Communication & Information
Nanyang Technological University
Singapore

Syin Chan

School of Computer Engineering
Nanyang Technological University
Singapore

Sentiment Classification of Product Reviews Using SVM and Decision Tree Induction

ABSTRACT

This paper reports a study in automatic sentiment classification, i.e. automatically classifying documents as expressing positive or negative sentiments/opinions. The study investigates the effectiveness of using SVM (Support Vector Machine) and Decision Tree induction on various text features to classify product reviews into *recommended (positive sentiment)* and *not recommended (negative sentiment)*. Compared with traditional topical classification, it was hypothesized that syntactic and semantic processing of text would be more important for sentiment classification. In this study, five different approaches, *unigrams* (individual words), *part-of-speech tagging, association rules, use of negation*, and *use of WordNet synsets (identifying a set of synonyms)* were investigated. A sample of 1,800 miscellaneous product reviews was retrieved from Review Centre (www.reviewcentre.com, 2003) for the study. 1,200 reviews were used for training, and 600 for validation. Using SVM, the baseline unigrams approach obtained an accuracy rate of 81.3%. The use of WordNet synsets obtained marginally better result of 81.7%. The other text features did not yield better results. Error analysis suggests 3 approaches for improving classification accuracy: making inference from superficial words, solving the problems of "comments on parts" and "negation". Finally, Decision Tree induction was used to generate a list of indicative words that can identify the polarity of articles.

1 INTRODUCTION

Research in *Automatic Text Classification* seeks to develop models for assigning category labels to new documents or document segments based on a training set of documents that have been pre-classified by domain experts. Most studies of automatic text classification have focused on "topical classification", i.e. classifying documents according to various subjects (e.g. education vs. entertainment). This study is in the area of "Sentiment Classification" – automatically classifying documents according to the overall sentiment expressed in them. In particular, this study investigated the application of machine-learning methods for classifying product reviews into two categories: *recommended (positive sentiment)* and *not recommended (negative sentiment)*.

Automatic sentiment classification is useful in many areas. It can be used to classify product reviews into positive and negative, so that potential customers can have an overall idea of how a product is perceived by other users (Turney, 2002; Pang, Lee & Vaithyanathan, 2002; Dave, Lawrence & Pennock, 2003). It can also be used to classify Web articles into positive or negative comments, enabling users to browse Web pages more efficiently. Moreover, the technique can be used for filtering out email messages with impolite or abusive words (Spertus, 1997). In the area of social science research, it can be used to categorize news articles into positive and negative views, according to various research purposes (Semetko & Valkenburg, 2000; Lind & Salo, 2002).

Though machine-learning techniques have long been used in topical text classification with good results, they are less effective when applied to sentiment classification (Pang et al. 2002). Sentiment classification is a more difficult task compared to traditional topical classification, which classifies articles by comparing individual words (unigrams) in various subject areas. In sentiment classification, unigrams may not be enough for accurate classification. For instance, the two phrases "you will be disappointed" and "it is unsatisfactory" do not share the same words, but both express negative sentiments.

In this study, the authors investigated five different approaches to perform the classification using different text features: *unigrams(individual words), part-of-speech tagging, association rules, use of negation,* and *use of WordNet synsets.*

2 PREVIOUS STUDIES

Relatively few studies have focused on non-topical text classification, which is a relatively new research area. A few studies have focused on genre classification, e.g. identifying whether a document is a research paper or a commercial report. Other studies have attempted to identify words with different semantic orientation or polarity. Such word lists are useful for sentiment classification of documents. There are also a small number of studies on classification of reviews.

Kessler et al. (1997) attempted to automatically detect whether the genre of a document is *reportage, editorial, legal document, scitech, non-fiction,* or *fiction.* They identified genres through 3 kinds of cues in documents: *lexical cues* (e.g. Mr or Ms), *character-level cues* (e.g. the presence of question marks, exclamations marks, capitalized and hyphennated words), and *derivative cues* (e.g. the standard deviation of the sentence length). Logistic Regression and Neural Network modelling were used to construct genre classifiers, achieving an accuracy of 79%.

Finn et al. (2002) simply considered genre classification as a special case of topical classification, and applied the machine-learning methods, such as Naïve Bayes, C4.5 and KNN, to "Bag-of-Words", "part-of-speech" tagging, and hand-crafted shallow linguistic features of the text. The hand-crafted features performed best with an average accuracy of 88%.

Hatzivassiloglou and Mckeown's (1997) work can be considered a basic study of semantic orientation. They hypothesized that the conjunction of two adjectives in a sentence can be used to identify whether the two are of the same or different semantic orientations. For instance, in the sentence "The tax proposal was simplistic **but** well received", "but" connects the two opposite words "simplistic" and "well received". Using a log-linear regression model, they identified 1,336 adjectives as having *positive* or *negative* orientation by analyzing a corpus of 12 million words, and reported accuracy ranging from 78.08% to 92.37%. Their study sought to identify positive and negative adjectives, not documents, and can be used later to identify the overall polarity of documents.

Based on the same idea of using the conjunction of words to identify semantic orientation, Turney and Littman (2002) used a simpler approach to identify the semantic orientations of four kinds of words: adjectives, adverbs, nouns, and verbs. In their study, the semantic orientation of a word is calculated as: PMI (word, {positive samples}) - PMI (word, {negative samples}), where PMI (word, {positive samples}) is the Point-wise Mutual Information formula, which measures how closely the word is associated with positive words like *good, excellent,* etc, PMI (word, {negative samples}) measures how closely the word is associated with negative words like *poor, nasty,* etc.

The above PMI-IR formula has been used in a study by Turney (2002), which focused on classifying various product reviews into *recommended* or *not recommended,* according to the semantic orientation of the adjective and adverb phrases. An average accuracy of 74% was reported in the experiment, ranging from 84% for automobile reviews to 66% for movie reviews. A byproduct of the study was a list of phrases assigned with semantic orientation.

Pang et al. (2002) focused on classifying movie reviews into positive or negative. They treated the problem as a special case of topical categorization, and used three traditional machine-learning classifiers: Naïve Bayes, maximum entropy classification, and SVM (support vector machine). In their study, several features of the documents, such as unigram (individual word), bigram (two-word phrase), and part-of-speech tag, were separately selected to feed into the three classifiers. Using unigrams as terms, and the Term Presence as weight, SVM outperformed the other two methods, yielding a highest accuracy of 82.9 %.

Both the studies above reported that sentiment classification is more difficult than topical classification, and the authors suggested: "fundamentally, it seems that some form of discourse analysis is necessary" (Pang et al., 2002, p. 85).

In this study, our overall approach is applying SVM and decision-tree induction on various document features: *Unigrams, part-of-speech tags, association rules, use of negation,* and *use of WordNet synsets.* By using existing machine-learning methods, we can focus on investigating the various features of documents, and how these features can be manipulated to improve the performance of machine-learning methods.

3 RESEARCH METHOD

Sampling

Using a computer program, product reviews were automatically downloaded from Review Centre (www.reviewcentre. com, 2003), which hosts millions of product reviews by consumers. After filtering out blank Web pages, a sample of 1,800 product reviews was systematically selected, comprising 900 positive reviews and 900 negative reviews.

The sample was divided into a training set of 1,200 reviews (600 positive and 600 negative) for developing the classification model, and a test set of 600 reviews (300 positive and 300 negative) for evaluating the accuracy of the model. Table 1 summarizes the distribution of the 1,800 product reviews by product category. The majority of reviews are of mobile phones and electronic equipments.

Review Centre rates product reviews using a 10-star rating system. In this study, reviews with 7 stars or above are coded as *recommended (positive)*, while reviews with 4 stars or below are *non-recommended (negative)*. The aim of the classification model is to predict from the natural language text of the review whether the review is coded as recommended or non-recommended.

Table 1. Distribution of product reviews by product category

Category	Number
Book	4
CD	6
Film	9
Web Site	11
Holiday	13
Computer Game	30
Electrical Appliances	31
Computer Software	34
Sports Equipment	36
Film Cameras	62
Online Shop	76
HiFi	88
Digital Camera	116
Computer Hardware	121
Home Cinema and DVD Player	140
Car	156
Mobile Phone	867

Pre-Processing

The texts of the reviews were tokenized and the words extracted were stemmed using the Conexor parser (Tapanainen & Järvinen, 1997). Each review was converted into a vector of term weights, indicating the importance of each term (i.e. word) in the review. Three weighting schemes were investigated: *TF(Term Frequency)*, *TFIDF*, and *Term Presence(binary weighting)*.

The *TFIDF* weight has been used in many studies on topical text classification, and is defined by the formula: $\mathrm{TF} \times \mathrm{Log}\left(\frac{\mathrm{N}}{\mathrm{DF}}\right)$

where *TF* is the number of times the term occurs in the review, *N* is the number of reviews in the training set, and *DF* is the document frequency – the number of reviews in the training set containing the term.

Term Presence (binary weighting) has the value 1 if the term exists in the review, 0 otherwise. *Term Frequency* (TF) uses the frequency of the term in the review as the weight.

Machine-learning Methods

Two machine-learning methods, Support Vector Machine (SVM) and Decision Tree induction, were used in this study. SVM has been applied to text classification in Joachims's study (1998), and later used in many studies (Joachims, 1999; Schohn & Cohn, 2000). The core idea is to find a hyperspace surface H, which separates positive and negative examples with the maximum distance. Yang (1999) claimed that SVM and k-NN methods were significantly better than other classifiers. Sebastiani (2002) reported that SVM delivered top-notch performance in some experiments. In this study, SVM^{light} (www.svmlight. joachims.org, 2003), a publicly available SVM program, was used for automatic review classification.

However, the SVM program is a black box, and it is difficult to inspect the nature of the classification model and determine what words are important in the model. Thus, we applied another machine-learning method, Decision Tree induction (Quinlan, 1983), to identify useful words for sentiment classification. Many studies have found that SVM performed better than Decision Tree on text classification (Joachims, 1998; Yang & Liu, 1999). However, the decision tree model is easy to interpret and can be converted to IF-THEN rules. We use it in this study to identify the useful linguistic features. C5.0 (www.rulequest.com/see5-info.html, 2003), a program implementing decision tree induction, was used in this study.

A decision tree is a tree structure that is used like a flow-chart. Each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf notes represent categories. To categorize a new case, the attribute values of the case are tested against the decision tree. A path is traced from the root to a leaf node that holds the category prediction for that document.

Approaches Investigated

Baseline (Unigram). The baseline approach is simply using all the individual stemmed words (unigrams) that appeared in at least 3 reviews. Other approaches will be compared to this baseline to see whether they improve on the effectiveness.

Selected Terms (Verbs, Adjectives, Adverbs). We noticed that in many product reviews, verbs, adjectives, and adverbs were used to express positive or negative sentiments, so we experimented with selected terms, i.e. only verbs, adjectives, and adverbs. Conexor parser was used to process the product reviews and filter out words that are verbs, adjectives and adverbs.

Terms labeled with part-of-speech (POS) tags. In the baseline approach, we used the original words as terms, without considering their part-of-speech tags. However, the diversity of word senses may result in ambiguity. Here we combined the individual words with their POS tags. In this approach, the words "better:adjective" and "better:verb" are considered different terms.

Use of negation. Use of negation in reviewers' comments need to be taken into account in a classification model. Negation words include: *can't, couldn't, didn't, doesn't, no, none, not, won't, isn't, wasn't.* In this approach, each negation and its adjacent word are combined to generate a new term. For example, "not good" is converted into "Neg_good".

Association Rules. So far, we only considered individual words as input terms, but sometimes, phrases and pairs may be more useful to infer sentiment. For instance, the concatenation of two words "better than" probably implies a positive sentiment. We used an association rule algorithm to extract bigram or trigram linear patterns that occurred at least in 50 reviews. These patterns replaced the individual words in the text. For example, the words "better than" in the text were replaced with the pattern *better&than.*

Use of WordNet Synsets. Though different words were used in the reviews, many of them are synonymous. For example, "benefit," "profit," and "gain" are close in meaning. To investigate the use of synonyms, individual words were mapped to synsets of WordNet (www.cogsci.princeton.edu/~wn/, 2003). In effect, words were replaced with WordNet synsets, and term weights were calculated for the synsets. No disambiguation was attempted. All candidate synsets were used to replace a word.

4 RESULTS

Using SVM

Table 2 lists the results of the various approaches attempted in this study.

Table 2. Various approaches and results

ID	Approach	Term Weighting	Accuracy
1	Unigram	TFIDF	***81.0%***
2	Unigram	TF	***81.3%***
3	Unigram	Presence	79.3%
4	Selected terms (Verbs, Adjectives, Adverbs)	TF	77.3%
5	Selected terms (Verbs, Adjectives, Adverbs)	Presence	77.0%

6	Selected terms, labeled with POS	TF	76.6%
7	Selected terms, labeled with POS	Presence	77.5%
8	All terms, labeled with POS	TF	78.4%
9	All terms, labeled with POS	Presence	***81.3%***
10	Association rules	TF	73.8%
11	Use of Negation	TF	79.0%
12	WordNet synsets (Nouns, Verbs, Adjectives, Adverbs)	TF	***81.3%***
13	WordNet synsets plus words	TF	***81.7%***
14	Decision Tree induction using WordNet synsets	TF	77.5%

Note: $\text{Accuracy} = \dfrac{\text{No. of positive reviews correctly classified}}{\text{300 (positive reviews in test set)}}$

Though the use of unigrams is the simplest approach, it obtained one of the best results: 81.33% accuracy. Though *TFIDF* weighting is effective in traditional topical text classification, it did not do better than *TF* when applied to sentiment classification in this study. This confirmed the results of the study by Pang et al. (2002).

Limiting the terms to just verbs, adjectives and adverbs did not degrade the effectiveness too much. The accuracy rate fell from 81.3% to 77.3%. This supports the hypothesis that positive and negative sentiments are expressed mostly through verbs, adjectives and adverbs.

The use of additional part-of-speech information did not improve results, possibly because it increased the number of dimensions (each word is subdivided into different part-of-speech) and reduced the TF for each term.

The use of *association rules (bigrams and trigrams)* and *negation* also failed to achieve better results. This suggests that the simple approach to handling the phrasal terms and negation is not good enough.

Finally, the use of WordNet synsets produced marginal improvement. Because WordNet includes only four kinds of words (Nouns, Verbs, Adjectives and Adverbs), we first used only the synsets of these four kinds of words. The resulting accuracy 81.3% was the same as for the baseline unigrams approach. By including the individual words in addition to synsets, the result improved marginally to 81.7%.

An advantage of using synsets is that, the number of words used in the classification model was decreased from 2,200 unique words to 1,883 synsets. This reduces the computational complexity. It also helped to simplify the decision tree constructed using decision tree induction described in the next section.

Using Decision Tree Induction

Now using synsets as terms, we applied Decision Tree induction on the training sample to identify indicative words that are potentially useful for identifying the polarity of reviews.

As expected, the Decision Tree model was not as effective as the SVM model when applied on the test data set. The Decision Tree obtained 77.5% accuracy, compared to 81.7% for SVM. However, it did generate some rules, which could be useful for identifying indicative words. An interesting finding is that some rules are product-related. For instance, in Figure 1, the node "lens" is normally used for camera, so its right branches are most probably applicable for classifying camera reviews. Figure 1 lists a part of decision tree generated.

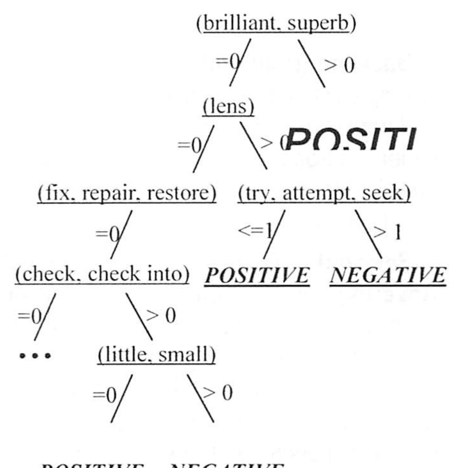

Figure 1. Some branches of Decision Tree

Going down the branches of the tree, it is clear that the synsets "try, attempt, seek" and "little, small" may be considered an indication of negative comment, while "brilliant, superb" is obviously a positive sign. The decision tree can be converted into a set of rules for easier reading and interpretation. The set of rules for sentiment classification is listed in the Appendix.

4 ERROR ANALYSIS

We analyzed the reviews in the test set that were wrongly classified by the SVM model to identify the sources of error and directions for improving the automatic classification. Normally, when applied to topical text classification, the accuracy of SVM is above 85% (Joachims, 1998; Yang & Liu, 1999).

The possible reasons for failure in automatic classification are summarized in Table 3.

Table 3. Error analysis

Reason	Quantity	Percent
Non-indicative comments	6	5.5%
Short review	8	7.3%
Negation	16	14.5%
Need inferencing	37	33.6%
Comments on parts	43	39.1%

Notes:

1. **Non-indicative comments.** In 6 cases, there was no apparent relation between the reviewer's comments and the number of stars given. For instance, the comment "The Weihrauch HW35 is a bit heavy and gets scratched too easily" is apparently negative, however the reviewer gave it 8 stars.
2. **Short review.** Some reviews are too short to be classified accurately. For example, the comment "This is an OK phone but slow" is difficult to classify without more context.
3. **Negation.** Although we attempted a simple bigram approach to handling negation, it did not improve the results. Some negations in the reviews still affected the effectiveness of the classifier. For instance, the sentence "I'd *never* regretted purchasing it" is actually a positive comment, however.
4. **Need inferencing.** Some comments are complex and need inferencing to identify the polarity. The sentence "if the price dropped, the company would be surprised how it would sell" contains no apparent positive or negative words.
5. **Comments on parts.** Sometimes, though the reviewer commented negatively on parts of the product, he/she is actually satisfied with the product as a whole, e.g. "The best phone I've had yet. The ONLY bad point is that...". This is the most common problem in sentiment classification.

As shown in Table 3, the errors attributed to "*negation*", "*need inferencing*" and "*comments on parts*" account for a large portion of the wrong classifications. In order to rectify these errors, further syntactic and semantic processing and inferencing appears to be needed.

5 CONCLUSION

This study has investigated the use of Support Vector Machine (SVM) for automatic sentiment classification of product reviews using various text features. This is a relatively new area in automatic text classification. It was found that using TF weighting scheme, the baseline unigrams approach gave one of the best results with 81.3% accuracy. Limiting the analysis to just verbs, adjectives and adverbs degraded the results by about 4%, suggesting that sentiments are expressed mostly through verbs, adjectives and adverbs. Replacing words with WordNet synsets marginally improved the accuracy to 81.7%. Since WordNet is capable of revealing deeper relations between synsets (e.g. "is a kind of" or "is one way to"), the use of synsets is worth further investigation. We plan to explore the use of higher-level semantic categories as well as relations between word categories in the automatic classification model. The error analysis on wrong classifications identified five factors affecting the performance of machine-learning methods: "Non-indicative comments," "Short review," "Negation," "Need inferencing," and "Comments on parts", with the last three factors accounting for most of the errors.

Future work will focus on performing syntactic and semantic processing and inferencing to address the problems of "Negation," "inferencing," and "Comments on parts".

By applying Decision Tree on the data set, the authors obtained a list of indicative words, which may be useful for further analysis. A set of rules using synsets to classify reviews is listed in the Appendix.

REFERENCE

Dave, D., Lawrence, S., Pennock, D. M. (2003). Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th International World Wide Web Conference,* Budapest, Hungary, May 20-24, 2003, 519-528.

Data Mining Tools See5 and C5.0. (n.d.). Retrieved July 18, 2003 from http://www.rulequest.com/see5-info.html

Finn, A., Kushmerick, N., & Smyth, B. (2002). Genre Classification and Domain Transfer for Information Filtering. In *Proceedings of the 24th European Colloquium on Information Retrieval Research*, Glasgow, Scotland, UK, March 25-27, 2002, 353-362.

Hatzivassiloglou, V., & McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and the 8th Conference of the European Chapter of the Association for Computational Linguistics*, New Brunswick, NJ, U.S.A., July 7-10, 1997, 174-181.

Joachims, T. (1998). Text categorization with support vector machines: Learning with many relevant features. In *Proceedings of 10th European Conference on Machine-learning,* Chemnitz, Germany, April 21-24, 1998, 137-142.

Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *Proceedings of 16th International Conference on Machine-learning*, Bled, Slovenia, June 27-30, 1999, 200-209.

Kessler, B., Nunberg, G., & Schutze, H. (1997). Automatic detection of text genre. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, Madrid, Spain. July 7-11, 1997, 32-38.

Lind, R. A., & Salo, C. (2002). The framing of feminists and feminism in news and public affairs programs in U.S. Electronic Media. Journal of Communication, 52(1), 211-228.

Pang, B., Lee, L., & Vaithyanathan, S. (2002) Thumbs up? Sentiment classification using machine-learning techniques. In *Proceedings of the 2002 Conference on Empirical Methods in Natural Language Processing,* Philadelphia, PA, U.S.A., July 6-7, 2002, 79-86.

Quinlan, R. (1983). "Learning efficient classification procedures and their application to chess end games". In *Machine Learning: An Artificial Intelligence Approach*, R.S. Michalski, J.G. Carbonell, and T.M. Mitchell (eds.). Morgan Kaufmann. pp. 463-482.

Review Centre – Welcome to Review Centre. (2003). Retrieved June 15, 2003 from http://www.reviewcentre.com

Schohn, G., & Cohn, D. (2000). Less is more: Active learning with support vector machines. In *Proceedings of the 17th International Conference on Machine-learning*, Stanford, CA, U.S.A., June 29-July 2, 2000, 839-846.

Semetko, H. A., & Valkenburg, P. M. (2000). Framing European politics: A content analysis of press and television news. *Journal of Communication, 50*(2), 93-109.

Sebastiani, F. (2002). Machine-learning in automated text categorization. *ACM Computing Surveys, 34*(1), 1-47.

Spertus, E. (1997). Somkey: Automatic recognition of hostile messages. In *Proceedings of the 5th International Conference on Intelligent User Interfaces*, Providence, RI, U.S.A., July 27-31, 1997, 1058-1065.

SVM^{light} – Support Vector Machine. (2002). Retrieved March 25, 2003 from http://svmlight.joachims.org

Turney, P. D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, Philadelphia, PA, U.S.A., July 7-12, 2002, 417-424.

Turney, P. D., & Littman, M. L. (2002). *Unsupervised learning of semantic orientation from a hundred-billion-word corpus* (Report No. ERB-1094). National Research Council, Institute for Information Technology. (NRC #44929)

Yang, Y. (1999). An evaluation of statistical approaches to text categorization. *Information Retrieval. 1*(1-2), 69-90.

Yang, Y., & Liu, X. (1999). A re-examination of text categorization methods. In *Proceedings of 22nd ACM International Conference on Research and Development in Information Retrieval*, Berkeley, CA, U.S.A., August 16-19, 1999, 42-49.

Tapanainen, P., & Järvinen, T. (1997). A non-projective dependency parser. In *Proceedings of the 5th Conference on Applied Natural Language Processing,* Washington, WA, U.S.A., March 31-April 3, 1997, 64-71.

Wordnet. (nd). Retrieved June 18, 2003 from http://www.cogsci.princeton.edu/~wn/

APPENDIX Rule Set for Sentiment Classification of Product Reviews

Rules for negative sentiment

Rule #1 :
if (*buy, purchase*) > 1
and (*camera, photographic camera*)= 0
and (*good, well*) = 0
and (*brilliant, superb*) = 0
and (*additional, extra, other*) = 0
then -> **negative**

Rule #2 :
if (*begin, commence, start*) = 0
and (*good, well*) = 0
and (*bad*) > 0
and (*great*) = 0
then -> **negative**

Rule #3 :
if (*appear, look, seem*) = 0
and (*late, lately, latterly*) > 0
and (*great*) > 0
then -> **negative**

Rule #4 :
if (*network, web*) = 0
then -> **negative**

Rules for positive sentiment

Rule #1 :
if (*side*) = 0
and (*crisp, sharp*) > 0
then -> **positive**

Rule #2 :
if (*apply, use, utilize*) <= 1
and (*break, interrupt*) = 0
and (*desire, want*) = 0
and (*caller, company*) = 0
and (*human, person, someone*) = 0
and (*good, well*) > 0
and (*decent, nice*) > 0
and (*fast*) = 0
then -> **positive**

Rule #3 :
if (*begin, commence, start*) = 0
and (*break, interrupt*) = 0
and (*card*) > 0
and (*week, hebdomad*) = 0
and (*good, well*) > 0
and (*brilliant, superb*) = 0
then -> **positive**

Rule #4 :
if (*finger*) = 0
and (*brilliant, superb*) > 0
then -> **positive**

Rule #16 :
if (*become, get*) = 0
and (*begin, commence, start*) = 0
and (*break, interrupt*) = 0
and (*fix, repair, restore*) = 0
and (*desire, want*) = 0
and (*would*) = 0
and (*caller, company*) = 0
and (*guarantee, warranty*) = 0
and (*look, looking at*) = 0
and (*human, person, someone*) = 0
and (*merchandise, product*) = 0
and (*receipt, reception*) = 0
and (*stick*) = 0
and (*week, hebdomad*) = 0
and (*out*) = 0
and (*although, though*) = 0
and (*good, well*) > 0
and (*bad*) = 0
and (*free*) = 0
and (*mobile*) = 0
then -> **positive**

Rule #17 :
if (*begin, commence, start*) = 0
and (*deal, manage*) > 0
and (*week, hebdomad*) = 0
and (*good, well*) > 0
then -> **positive**

Rule #5 :
if (*break*, *interrupt*) = 0
and (*anticipate*, *expect*) = 0
and (*fast*) > 0
then -> **positive**

Rule #6 :
if (*desire*, *want*) = 0
and (*caller*, *company*) = 0
and (*never*) = 0
and (*good*, *well*) > 0
and (*free*) > 0
and (*great*) = 0
and (*mobile*) = 0
then -> **positive**

Rule #7 :
if (*begin*, *commence*, *start*) = 0
and (*buy*, *purchase*) <= 1
and (*camera*, *photographic camera*) = 0
and (*characteristic*, *feature*) = 0
and (*job*, *problem*) = 0
and (*good*, *well*) = 0
and (*bad*) = 0
and (*decent*, *nice*) > 0
then -> **positive**

Rule #8 :
if (*begin*, *commence*, *start*) = 0
and (*flash*) > 0
and (*guarantee*, *warranty*) = 0
and (*although*, *though*) = 0
and (*good*, *well*) > 0
and (*mobile*) = 0
then -> **positive**

Rule #9 :
if (*begin*, *commence*, *start*) = 0
and (*receipt*, *reception*) = 0
and (*britain*, *UK*) > 0
and (*good*, *well*) > 0
then -> **positive**

Rule #10 :
if (*network*, *web*) > 0

Rule #18 :
if (*begin*, *commence*, *start*) = 0
and (*break*, *interrupt*) = 0
and (*continue*, *go on*) = 0
and (*caller*, *company*) = 0
and (*business*, *job*) = 0
and (*card*, *menu*) = 0
and (*human*, *person*, *someone*) = 0
and (*job*, *problem*) = 0
and (*fix*, *repair*, *restore*) = 0
and (*good*, *well*) = 0
and (*bad*) = 0
and (*additional*, *extra*) > 0
then -> **positive**

Rule #19 :
if (*appear*, *look*, *seem*) = 0
and (*desire*, *want*) > 0
and (*week*, *hebdomad*) = 0
and (*almost*, *about*)> 0
and (*good*, *well*) > 0
and (*mobile*) = 0
then -> **positive**

Rule #20 :
if (*break*, *interrupt*) = 0
and (*fix*, *repair*, *restore*) = 0
and (*attempt*, *seek*)<= 1
and (*beginner*, *initiate*)= 0
and (*easily*, *easy*)> 0
and (*brilliant*, *superb*) = 0
and (*excellent*, *fantabulous*) = 0
and (*useful*) = 0
then -> **positive**

Rule #21 :
if (*arrive*, *come*) > 1
and (*say*, *state*, *tell*) > 0
and (*end*, *terminate*) = 0
and (*guarantee*, *warranty*) = 0
and (*fast*) = 0
and (*great*) > 0
then -> **positive**

Rule #22 :

then -> **positive**

Rule #11 :

if (*break, interrupt*) = 0
and (*guarantee, warranty*) = 0
and (*damage*) > 0
and (*merchandise, product*) = 0
and (*good, well*) > 0
and (*bad*) > 0
then -> **positive**

Rule #12 :

if (*continue, go on*) = 0
and (*continue, go on*) = 0
and (*excellent, fantabulous*) > 0
then -> **positive**

Rule #13 :

if (*begin, commence, start*) = 0
and (*break, interrupt*) = 0
and (*say, state, tell*) = 0
and (*end, terminate*) = 0
and (*far*) = 0
and (*late, lately, latterly*) = 0
and (*excellent, fantabulous*) = 0
and (*great*) > 0
then -> **positive**

Rule #14 :

if (*buy, purchase*) > 0
and (*buy, purchase*) <= 1
and (*fix, repair, restore*) = 0
and (*caller, company*) = 0
and (*card, menu*) = 0
and (*immediately, directly*) > 0
and (*good, well*) = 0
and (*bad*) = 0
then -> **positive**

Rule #15 :

if (*break, interrupt*) = 0
and (*go, move*) > 0
and (*good, well*) > 0
and (*mobile*) = 0
then -> **positive**

if (*become, get*) = 0
and (*cry, shout, yell*) = 0
and (*caller, company*) = 0
and (*human, person, someone*) = 0
and (*job, problem*) = 0
and (*mho, siemens*) = 0
and (*even*) > 0
and (*immediately, directly*) = 0
and (*good, well*) = 0
and (*bad*) = 0
then -> **positive**

Rule #23 :

if (*fix, repair, restore*) = 0
and (*camera, photographic camera*)> 0
and (*good, well*) = 0
and (*bad*) = 0
then -> **positive**

Rule #24 :

if (*belief, feeling, opinion*)> 0
and (*good, well*) = 0
then -> **positive**

Rule #25 :

if (*allow, permit*) > 0
and (*fix, repair, restore*) = 0
and (*caller, company*) = 0
and (*receipt, reception*) > 0
then -> **positive**

Rule #26 :

if (*must, need*) > 0
and (guarantee, *warranty*) = 0
and (*week, hebdomad*) > 0
and (*good, well*) > 0
then -> **positive**

Discussion Points

51. What does the author mean by classification?

52. Does the research involve creation or implementation of a classification scheme?

53. How does the researcher use classification to improve the automated approach?

54. How do these methods compare to current human-generated approaches to classification?

55. How does the reported research expand our understanding of classification?

56. Does the research suggest an improvement over human-generated classification?

57. What do you think are the most important lessons learned in this research?

58. What do you think are the best practices reported in this research?

59. What would you recommend to the researcher as the next step in this approach?

60. Is there other related research that you would recommend the researcher become acquainted with?