Text Categorisation of Racist Texts Using a Support Vector Machine

Edel P. Greevy[1], Alan F. Smeaton[2]

[1]PRINCIP Project – SALIS – Dublin City University – Dublin 9 – Ireland
[2]Centre for Digital Video Processing – Dublin City University – Dublin 9 – Ireland

Abstract

The automatic processing of text is a major challenge because of the increasing availability of textual information and the need to organise and manage such information effectively and efficiently. Automatic Text Categorisation is one of a number of functions we would like to have available to us and involves the assignment of one or more predefined categories to text documents in order that they can be effectively managed. In this paper we examine the problems associated with categorising texts documents (web pages) based on whether or not they are racist. We describe work in the PRINCIP project, which aims at the development of a system to detect racism based on the results of linguistic and statistical analysis of candidate texts. We take what we have learned from the PRINCIP research and apply machine learning techniques, specifically Support Vector Machines, to automatically categorise web pages. Our work shows that it is possible to develop automatic categorisation of web pages, based on these approaches.

Keywords: machine learning, text categorisation, support vector machines.

1. Introduction

Automatic Text Categorisation (TC) is the task of assigning predefined categories to free text documents (Yang, 1999). Texts are assigned to categories based on a confidence score that is suggested by a training set of labelled documents. This confidence score usually ranges between 0 and 1 and in order to arrive at a yes/no decision for the inclusion/exclusion of a document in a category, the confidence score is mapped onto one of the Boolean values {0,1} using thresholds. TC techniques have been successfully applied to many domains: for the classification of news stories into relevant newsgroups such as sports, politics, environmental issues, to detect spam and to assign web pages into Yahoo!-like web directories.

The PRINCIP project aims at the realisation of a multilingual system for detecting racist documents on the web. PRINCIP is primarily a linguistics-based project working to establish common methodologies across three languages (French, German and English) through the corpus-based analysis of racist content with the aim of building a linguistic knowledge base (KB). Tagging, parsing, linguistic and statistical analysis using bespoke and existing linguistic tools and software suites, all play major roles in building this knowledge base. The linguistic patterns identified during analysis of web pages can be formulated into rules and used in a categorisation system, to allow for detection of illicit content on the web.

In this paper we present an overview of the techniques we use to automatically develop and evaluate a text categorisation system in PRINCIP. Our approach is dictated by the results of linguistic and statistical analysis of the text appearing on web pages, experiments conducted during the PRINCIP project. For the purpose of this study, a Support Vector Machine (SVM), which is a machine learning method, will be trained on several representations of the datasets collected for the PRINCIP project.

2. Racism on the Web and Ways to Detect It

In this paper we are concerned with racism in English-speaking countries only. The UK and Ireland are the main English-speaking countries in Europe while the USA, Canada, Australia, New Zealand, South Africa are the main countries outside of Europe. Those countries in which English is a strong second language include Israel, Holland and the Scandinavian countries as well as many other places in Europe. Thus we can see that racism in English can originate in many corners of the globe. Targets of racism differ from country to country, with, for example, immigrants, refugees and non-nationals being strong targets in Ireland; African-Americans, Hispanics and Asians being likely targets in the USA while Aborigines are the main target of racism in Australia. It must also be noted that the targets of racism constantly shift and change. September 11th, the ongoing war in Israel and Palestine, the war in Iraq, American foreign policy, the killings of white farmers in South Africa – current affairs – shift the attention from one group to another. Our corpus of web documents was gathered in September 2002 and it is already somewhat out of date as a result of the world events since then.

Internet legislation also impacts the presence of racism on the web. This differs across the globe with the U.S. being one of the most liberal. The U.S. First Amendment entitles its citizens to the right to freedom of speech and for that reason the majority of racism found on the WWW is US-based, though some groups, for strategic reasons, take advantage of more restrictive laws (e.g. in Canada) as it warrants them more publicity.

Non-technological methods that exist for the detection and removal of hate online include the setting up of regulatory authorities. For example the Netherlands has set up the Complaints Bureau for Discrimination; hotlines exist in EU countries which allow for potential breaches of legislation to be reported. Sites are investigated and if found to be illegal, are eventually removed. Such solutions are found to be weak because of the fluidity and size of the Internet. Documents originating in the USA, where legislation is most liberal, can be accessed across the globe but belong to another jurisdiction. Technical approaches thus far implemented include Internet Content Filters or Label Bureaus, which simply label sites and filter offensive ones (Internet Content Rating Association). Email is typically filtered using regular expressions containing keywords but this approach is unreliable, as it will only filter those emails containing known keywords. The Safer Internet Action Plan, which is sponsored by the European Commission, is currently funding various filtering and rating projects some of which include: the ICRAsafe project which will create a system to allow responsible adults to restrict children's access to Internet content that may harm them or which is otherwise considered undesirable by the adult; NETPROTECTII is a European tool for Internet access filtering to provide textual filtering in eight European languages. All project descriptions can be viewed under the URL provided for the Safer Internet Action Plan.

Current methods of filtering racism rely heavily on either keywords or the labelling of offensive material. In order to implement successful systems, a considerable human effort is required, not only in the initial stages of filter construction but also in an ongoing basis as the targets of racism change, the language evolves, existing websites are edited or new websites are added. Automatic text categorisation techniques are reported to have been successful when applied to other domains such as news story categorisation or the categorisation of web pages into Yahoo!-like directories, with results comparable to human evaluation and performance. Such methods lead to vast improvements in productivity as well as savings in terms of time and manpower, as the same human effort is not required. Given racism on the web changes so rapidly, it is one area that may benefit from the application of automatic techniques to text categorisation.

3. Text Categorisation

Text Categorisation (TC) is concerned with the automatic assignment of documents to predefined categories. TC is traditionally a content-based management task and has much in common with it's neighbour Information Retrieval (IR), borrowing and applying much of the basic IR techniques. IR is concerned with the matching of a user's information need, expressed as a query, against a corpus of documents in order to rank documents in the corpus in order of their estimated relevance to the information need. As the field of TC has progressed it is no longer just concerned with the assignment of documents into categories such as sport, politics or environmental issues but has attempted to solve more complex tasks such as the classification of documents according to genre (Finn and Kushmerick, 2003; Finn *et al.*, 2002). In recent years machine learning techniques have been applied to text categorisation problems. The promise of a future of machines capable of reading, examining and making decisions about free text has generated more interest in the field.

There are three main processes involved in text categorisation, namely:

1. Indexing and other pre-processing techniques are performed on initial training and test corpora and also on documents to be categorised by the classifier when in operation.
2. Classifiers take the training data as input and learn features of that data so that when presented with unseen data, it will use those learned features to make a decision about the category in which the document is assigned.
3. An evaluation procedure is used to measure the effectiveness of the classifier. After being trained on the training data, the classifier is presented with test data and the performance of the classifier is evaluated using precision and recall.

3.1. *Indexing and Pre-processing Operations*

Because text documents are not interpretable by a classifier, an indexing procedure must be applied to the text so as to map it onto an appropriate representation that can be fed to the classification system. Almost all representations consist of a set of *terms* each of which may be assigned a *weight* in each document to reflect its degree of importance for that document. Differences between various indexing approaches are reflected by different interpretations of what constitutes a term and of how weights are measured in a document.

3.1.1. *What can a term be?*

Term generation varies in the amount of linguistic and statistical sophistication used. To form the simplest indexing language each word can be treated equally as a feature. This is a common approach referred to as the *bag-of-words (BOW)* approach. However, relationships such as polysemy and synonymy which exist between words, can lead to many errors. For this reason, more complex methods are investigated for the creation of an effective set of *terms.* Where phrases are used as indexing terms, phrases can be determined using linguistic information (i.e. identified according to the grammar of the language) or by using statistical methods (i.e. identified according to the recurring frequency of a set of words).

The application of linguistic procedures to a text allows us to express language in terms of the roles words play in a text and the relationships between words. This, in turn, provides a classifier with richer information about a document. Experiments conducted using linguistic information are inconsistent with some findings revealing they do not perform as well as BOW (Lewis, 1992; Smeaton, 1997) while others disagree (Fürnkranz *et al.*, 1998).

3.1.2. *Using Linguistic Information in Term Generation*

Some of the linguistic approaches to generating index terms reported in the literature include:

- Chandrasekar and Srinivas illustrated how syntactic information can be effective in filtering out irrelevant documents after documents have been retrieved by a search engine. They reported on the performance of two different methods of syntactic labelling namely Part of Speech (POS) Tagging and Supertagging (Chandrasekar and Srinivas, 1998).
- Fürnkranz *et al.* illustrated how the use of linguistic phrases as input features can improve precision at the expense of recall. Linguistic phrases were constructed using a system called AUTOSLOG which is an automatic method for extracting patterns from a POS tagged text. The system is fed noun phrases which are used in the construction of linguistic patterns which are in turn used by the classifier during the categorisation of documents (Fürnkranz *et al.*, 1998).
- Lewis performed tests on the use of syntactic indexing phrases, clustering of these phrases and clustering of words and found these approaches to be less effective than the frequency of occurrence of words. Lewis proved word-based features to be more effective and attractive since a greater effort, both computationally and time-wise, is required in order to build a feature set of phrases (Lewis, 1992).
- In their study on the application of TC techniques to classify documents along dimensions that are orthogonal to topic, for example whether a document is primarily facts or an expression of someone's opinion or whether a document is positive or negative, Finn, Kushermick and Smyth found the distribution of POSs to be more effective than the BOW approach (Finn and Kushmerick, 2003; Finn *et al.*, 2002).

3.1.3. *Using Statistics in Term Generation*

The following illustrate the kinds of statistics-orientated approaches to the generation of indexing terms that have been explored in the literature.

- Mladenic and Grobelnik combine feature generation with feature selection through the use of statistical methods. What is interesting about this method is that the BOW vector space is enriched with phrasal information (word sequences of between 2 and 5 characters), meaning the classifier is not relying on the performance of phrases alone. Word sequences of size 3 proved to be the most effective (Mladenic and Grobelnik, 1998).
- Typically classifiers assume that the context of a word *w* has no impact on the meaning of *w,* which is of course not true. For this reason Cohen and Singer aim to construct a classifier that allows the context of a word *w* to affect whether the presence or absence of *w* will contribute to a classification. Cohen and Singer investigated two algorithms each of which have different notions as to what constitutes context. For the RIPPER algorithm a context of a word *w* is interpreted as a number of other words that must co-occur with *w,* where order and location in the document are irrelevant. Sleeping-experts, on the other hand, interprets the context of a word *w* as consisting of words that occur near *w* and in a fixed order (Cohen and Singer, 1998).
- Fürnkranz employed an algorithm very similar to that used by (Mladenic and Groblenik, 1998) for the generation of n-grams. Each document represented *m* set-valued features, one for each n-gram size $1<m<MaxNGramSize$ meaning for example that when $m=3$ all 3-grams, 2-grams and 1-grams are included in the feature set (Fürnkranz, 1998).
- Tan *et al.* investigated the use of bigrams to enhance text classification. In this experiment Tan *et al.* used bigrams in addition to unigrams. Those unigrams that appeared in a significant number of documents were selected and used as seeds for the generation of bigrams. Bigrams were generated and chosen on the basis that at least one of the pairs of bigrams had to be a seed (Tan *et al.*, 2002).

3.1.4. *Other Pre-processing Operations*

Before a document is indexed, the normal procedure in IR and TC is to remove *stop words*. Stop words comprise those words that are neutral to the topic of the document and would therefore generally contribute very little to the classification of a document. They are often defined by a *stoplist* and include articles, prepositions, conjunctions and some high-frequency words. This technique is performed so as to reduce the number of index terms in a document, to enhance computational efficiency and to minimise the amount of superfluous information in the term space. Different methods have been explored for the generation of stoplists. In their 1996 JASIS paper, Yang and Wilbur (Yang and Wilbur, 1996) apply the Wilbur-Sirotkin stop word identification method to text classification in order to reduce the computational cost without having to trade off on categorisation effectiveness (Wilbur and Sirotkin, 1992).

Stop words are not removed in experiments using syntactic information as terms (Lewis, 1992; Chandrasekar and Srinivas, 1998). Such experiments require the presence of all words in a sentence or document in order to assign the correct POS tags. *Term* or *feature extraction* techniques are used instead to identify the most discriminating and effective patterns.

3.2. *Post-Indexing Operations*

3.2.1. *Assigning weights to terms*

A term can be weighted using binary weights i.e. 1 denotes presence and 0 denotes absence, or using frequency weights. More complex term weighting methods exist, with weights usually ranging between 0 and 1. The *TF*IDF* term weighting function (Salton and Buckley, 1988) is a commonly used method.

3.2.2. *Dimensionality Reduction (DR)*

In TC since the number of terms occurring just once in a corpus can be extremely high, in some cases, efforts are made to reduce the dimensionality of the term space from r to $r`$. Large vector spaces can be problematic and can lead to *overfitting.* A good example of overfitting as provided by (Sebastiani, 2002) is that of a classifier trained on three examples for the category CARS FOR SALE. Two of the advertisements were concerned with the sale of blue cars and therefore the classifier considered the colour of the car (i.e. blue) to be a characteristic of the category. In other words classifiers affected by overfitting tend to be exceedingly good at classifying the training data but not so good at classifying unseen data.

There are two main approaches to DR, namely *term selection* and *term extraction.* In term selection, the $r`$ terms are chosen by selecting a subset of the original $r$ terms without loss in effectiveness. Document Frequency, Information Gain, Mutual Information, Chi-square and Term Strength are popular methods for term selection. In term extraction, the $r`$ terms that are extracted may not at all resemble the original r terms. Rather the $r`$ terms are obtained through a series of alterations, combinations, transformations etc. of the original r terms. Term Clustering and Latent Semantic Indexing are popular methods for term extraction.

4. Machine Learning

There is no conventional algorithm for the task of assigning a document to a predefined category, as no accurate mathematical model of the solution exists. Given a set of examples, we might be able to define input and output values for each example but we are unable to define how, given a certain input we arrive at the desired output. The relationship between the input and desired output is too complex to be captured in an algorithm and so the only way such a problem can be dealt with is by using machine-learning techniques.

Machine learning (ML) can be broadly split into two main areas: supervised learning and unsupervised learning. In supervised learning, the machine knows the output of an input pattern and tries to learn patterns that would arrive at the desired output. In unsupervised learning, the training set consists of input patterns only and the machine is trained without having any prior knowledge of the output. Its task is to learn to adapt based on the experiences of the previous training patterns.

Binary classifiers have a binary output i.e. *{0, 1}* meaning a document either belongs to a category or it does not. Multi-class classifiers allow a document to be categorised in one of a finite number of categories. In regression models the output is a real numbered output. ML has been applied to a wide range of areas from speech recognition, hand-written character recognition, image detection, POS tagging to medical diagnosis and prognosis and even learning to fly.

4.1. *Some Approaches to Classifier Construction*

Many methods, approaches and algorithms exist for the construction of a text classification system. Some of the more popular approaches are mentioned below.

Probabilistic classifiers view the classification problem in terms of a probability that a document *D* of binary or weighted terms belongs to a category *C*. The probability is calculated by applying Bayes Theorem. Many practitioners have experimented with probabilistic classifiers in the literature (Lewis and Ringuette, 1994; Yang and Liu, 1999; Chai *et al.*, 2002; Mladenic and Grobelnik, 1998; Joachims, 1998).

A decision tree classifier consists of a tree where each internal node is labelled by a term used to test an attribute. Each branch corresponds to an attribute value representing the weight that a term has in a test document and each leaf node is labelled by a category which is used to assign a classification. A document *d* is categorised by recursively testing for the weights that the terms have in the representation of *d*. This step is repeated until a leaf node is reached where the label of the leaf node i.e. the category is then assigned to *d.* (Lewis and Ringuette, 1994; Goller *et al.*, 2000; Joachims, 1998)

Decision rules first of all create a dictionary containing the features or attributes that represent individual documents in a collection or domain. A representation maps each individual document in a training set using the dictionary. Each document is assigned a label that denotes which category it belongs to. The objective is to find sets of decision rules or patterns that distinguish one category from the others (Apté *et al.*, 1994).

The Rocchio method is a vector-space-method which is very often used in information retrieval for relevance feedback, document filtering and routing. A prototype vector (centroid) is created for each category using a training corpus and is in effect the average of all positive examples. Document vectors belonging to a category are weighted positively and other documents are weighted negatively. The Rocchio classifier rewards the closeness of a test document to the centroid of the positive training examples and its distance from the centroid of the negative training examples (Goller *et al.*, 2000; Drucker *et al.*, 2001; Joachims, 1998).

A neural network classifier consists of a network of units. Input units represent terms and output units represent categories and they are connected by edges that have weights, which represent the conditional dependence relations between the I/O units. A document *d* is classified by taking its term weights and assigning them to the input units; the units are then propagated through the network and the value that the output unit takes up determines the categorisation decision (Yang and Liu, 1999).

4.2. *Support Vector Machines*

Support Vector Machines (SVMs) are one of the newer ML approaches, introduced by Vapnik in 1992. SVMs are based on statistical methods to minimise the risk of error and offer solutions to optimise generalisation performance.

One justification for using a SVM for TC is that it is "a principled and very powerful method that in the few years since its introduction has already outperformed most other systems in a wide variety of applications" (Cristianini and Shawe-Taylor, 2000). SVMs overcome the many problems associated with efficiency of training thereby making them a very attractive learning method.

SVMs are capable of overcoming the problems associated with high dimensional spaces (e.g. overfitting) due to the sophisticated statistical learning theory used. This means solutions can always be found efficiently even for training sets with thousands of examples.

The compact representation of the hypothesis being learned (in our case the categorisation of documents) means that evaluation on unseen input is very fast thereby making it efficient when it comes to testing. Within SVM terminology, *generalisation performance* refers to how well a hypothesis correctly classifies data not in the training set and a good learning machine will optimise generalisation performance. The *capacity* of a machine is the ability of the machine to learn any training set without error. A machine can have too much or too little capacity and this affects generalisation performance – too much capacity causes *overfitting.* The *VC-dimension* (Vapnik Chervonenkis) is a direct measure of the capacity of a machine.

4.2.1. *Generalisation Theory and how SVMs work*

In SVMs, the task is to learn the relationship between input/output pairs – this is known as the *target function.* When presented with an unseen document the machine can make a decision about the *target* class of the document. The *decision function* estimates the target function and this function is chosen from a set of candidate functions referred to as *hypotheses.*

N input/output pairings are represented by a vector $x_i \varepsilon R^n$, $i = 1, \ldots, n$ and the associated truth or class y_i where y_i is *1* if a document *d* belongs to category *c* and *–1* otherwise. The task of the machine is to choose the mapping $x_i \rightarrow y_i$ that minimises the risk of error.

$R(\alpha)$ is referred to as the actual risk. This cannot be computed as it depends on the unknown probability distribution $P(x,y)$ from which the data are drawn.

$R_{emp}(\alpha)$ is the empirical risk and is measured by the mean rate of error on the training set for a fixed and finite number of observations or training sets $\{x_i,y_i\}$.

The following inequality holds with probability *1-n*, if *l* is the number of training points:

$$R(\alpha) <= Remp(\alpha) + sqrt[(h(log(2l/h) + 1) – log(n/4)) / l]$$

The right hand side of this inequality, referred to as the risk bound or the VC-confidence, can be calculated if *h* is known. Minimising the risk bound or VC-confidence puts a bound on the actual risk. In this inequality equation, *h* is known as the VC-dimension and is the maximum number of training points that can be arbitrarily labelled by a set of functions. The VC-confidence increases as *h* increases so a lower VC-dimension will give a lower bound on the risk.

4.2.2. *Structural Risk Minimisation Principle*

SVMs implement the structural risk minimisation principle which attempts to overcome the problem of choosing the set of functions (i.e. the target function from all hypotheses) that has an appropriate VC-dimension while at the same time minimising the bound on the actual risk.

The entire set of functions is divided into nested subsets with decreasing capacity – for each subset either h or a bound on h is computed. This can be done by training a series of machines i.e. one for each subset. The goal of the training is to minimise the empirical risk $R_{emp}(\alpha)$ for a given subset. The subset of functions whose sum of empirical risk and VC confidence is minimal is chosen as the trained machine.

4.3. *Evaluation of Text Categorisation Systems*

Classifiers are experimentally evaluated by presenting new unseen data to the system. The efficiency of a classifier is tested by evaluating its capability of making the right categorisation decision. Precision and Recall are commonly used evaluative methods in IR and TC has borrowed these and applied them to the case of document categorisation. Precision is a measure of how accurate the system is at classifying unseen data for a particular category and is defined as the conditional probability $P(ca_{ix} = 1 \mid a_{ix} = 1)$, i.e. the probability that if a random document d_x is classified under c_i, this decision is taken. Recall is a measure of the degree of completeness or coverage for a specific category and is defined as the conditional probability $P(a_{ix} = 1 \mid ca_{ix} = 1)$, i.e. the probability that, if a random document d_x ought to be classified under c_i, this decision is taken (Sebastiani, 2002) .

5. TC in the PRINCIP Project

The PRINCIP project aims to build a system to detect and filter racism on the Internet using those rules found during linguistic analysis. In our research we use text categorisation methods to automatically achieve the same. Detecting racism on the Internet is not just a topic-based problem, rather it is more similar to genre detection as described by Finn, Kushmerick and Smyth (Finn and Kushmerick, 2003; Finn *et al.*, 2002), in that we are not really concerned with the topic itself but we are trying to identify features that will discern the author's attitude in relation to the topic, something which is orthogonal to the topic. In their work on genre detection Finn *et al.* found the distribution of POS to outperform the BOW approach for genre detection. Our own experiments in PRINCIP revealed there to be differences in some lexical, collocation and POS distributions across racist and non-racist documents (Lechleiter and Greevy, 2003; Martin, 2003a and 2003b; Gibbon and Greevy, 2003). In this study we perform a comparative analysis of the TC of racist texts by training Support Vector Machines on three representations: bag-of-words, n-gram word sequences, and POS, approaches which are primarily driven by the results of linguistic analysis in the PRINCIP project.

All three representations have already been tried and tested, both in IR and in text categorisation (Mladenic and Grobelnik, 1998; Tan *et al.*, 2002; Finn and Kushmerick, 2003; Finn *et al.*, 2002; Fürnkranz *et al.*, 1998; Lewis, 1992; Smeaton, 1997; Chandrasekar and Srinivas, 1998). However they have been examined in the context of other domains and different types of classification problems, that is, classification problems that are related to topic or content rather than to attitude or opinion. No such experiments have been conducted on the detection of racism on the Internet.

5.1. *About the dataset*

To conduct PRINCIP experiments, a web corpus of 3 million words, (approximately 500 documents per dataset) was collected. This consists of three datasets – web pages which are racist, anti-racist and neutral, i.e. neutral documents comprise those found using the same techniques used to detect racism but which are unrelated to the topic of racism. The datasets are in two formats – plain text and POS tagged.

For this study we are concerned with a binary classification problem, that is, either a document is racist or it is not. Therefore the training and test sets will contain positive examples i.e. racist texts and negative examples i.e. non-racist texts, comprising anti-racist and neutral pages. Each of the positive and negative datasets contains 500 documents.

When building the dataset, a combination of approaches was used to avoid circularity, to target a diverse collection of documents from different domains, and to target different groups. Yahoo! and Google directories were browsed. A list of potentially racist keywords and phrases was constructed. Recent and current affairs provided useful clues for the building of the list, as did studying research on racist discourse (van Dijk, 1987; Wodak and Reisigl, 2001). The list was submitted to search engines such as Google and AlltheWeb. We assumed that racist sites (and anti-racist) link to sites of a similar nature. It followed that downloading hyperlinks in a document proved a particularly useful method in corpus building.

5.2. *SVMs for PRINCIP*

Support Vector Machines were used to learn the features of the training sets and classify new unseen documents. SVMs are a very powerful learning method that "in the few years since its introduction has already outperformed most other systems in a wide variety of applications" (Cristianini and Shawe-Taylor, 2000). SVMs overcome many of the problems associated with efficiency of training such as overfitting and they are capable of generalising well in high dimensional spaces thereby making them a very attractive learning method.

5.3. *Results*

We built three representations of each dataset i.e. BOW, n-gram word sequences and POS tagged documents. The positive and negative datasets were divided into training and test sets (see table 1). The SVM learns the trainings set and uses those learned features to classify unseen documents i.e. the test set. In this study we split the training and test sets into different sizes to evaluate the impact of larger training sets on the SVM. Table 1 outlines the different size training and test sets used. Each of the experiments conducted on the different representations is evaluated in terms of precision and recall.

	Set 1	*Set 2*	*Set 3*	*Set 4*
No. Docs in Training Set	200	400	600	800
No. Docs in Test Set	60	100	150	200

Table 1. Illustrates number of documents in each set.

5.3.1. *Bag-of-Words*

The first representation used the simplest indexing language i.e. the BOW approach. Our investigation of lexical items equally consistent in each dataset revealed some words to be thirty percent more prevalent in racist texts e.g. *must, never, once, ever, same, very, course, fact, white, race, nation*. Modals, adverbs and truth claims were among this list. The use of modals, representing the taking of absolute positions, and the use of argumentation structures such as truth claims like *fact* or *of course* are both indicative of the discourse of racist language (Gibbon and Greevy, 2003; Lechleiter and Greevy, 2003). Though these same lexical items may be used by potentially anyone, the SVM results on the BOW representation prove promising for classification problem at hand.

Dataset	Term Weight	Accuracy on Test set	Precision	Recall
Set 1	Number of occurrences	60.00%	87.50%	23.33%
Set 1	Frequency	86.67%	92.31%	80.00%

Table 2. Evaluation of different methods of measuring term weight

Using set 1, we compared two methods of measuring term weight in the BOW representation. In table 2 we see that the precision and recall figures for TC dramatically improve when frequency is used as a means of measuring term weight. The accuracy on the test set increases considerably from 60% to 86.67%.

Because of the dramatic improvement in the performance of the SVM when frequency is used, we trained the SVM on each of the datasets and observed the effect in the figure below.

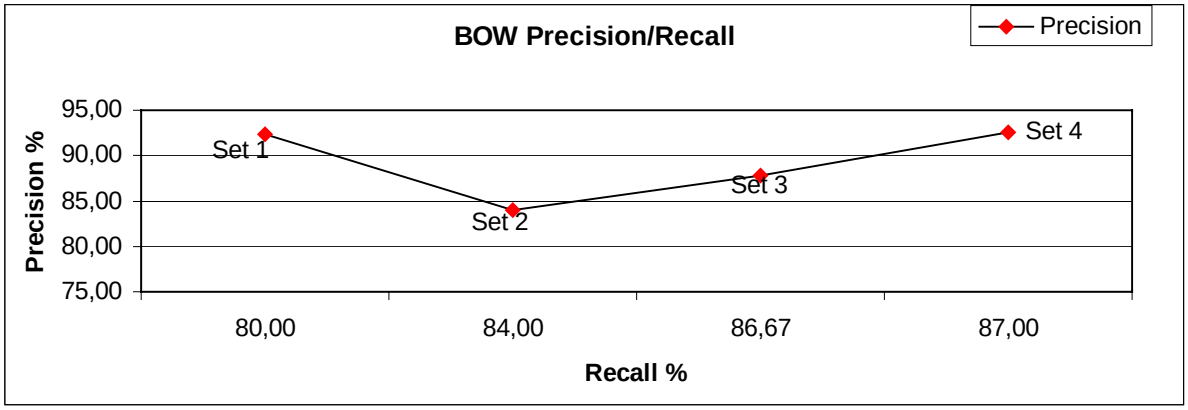

Figure 1. BOW precision and recall figures

Figure 1 illustrates how recall improves as the training set increases. Though precision figures took a drop when the training set was increased to 400, thereafter a steady increase was reported with the precision/recall figures for the final dataset achieving 92.55%/87.00%, a considerable improvement on precision/recall for set 1.

5.3.2. *N-gram word sequences*

In PRINCIP experiments, consistency analysis was performed on n-grams word sequences of length 2 and 3. Again certain n-grams (e.g. *our own kind, white civilisation, white survival, only Jews, our country)* were encountered significantly more often in the racist corpus showing that they are potentially discriminating.

Further SVM experiments will be carried out, first of all using the number of occurrences and then frequency as term weight. Both bigrams and trigrams will be investigated for each dataset. The results will then be compared to those obtained for BOW and POS.

5.3.3. *Parts-of-Speech*

The corpus was tagged using Xelda, a suite of linguistic tools made available to us by Xerox. The distribution of different POS across the three corpora was investigated.

	Racist	*Neutral*	*Anti-racist*
ADJ	8.89	8.58	7.7
ADV	4.61	3.7	3.29
NOUN	21.14	22.07	22.74
VERB	14.79	12.28	12.08
OTHER	50.57	53.38	54.19

Table 3. Distribution of the parts of speech in each corpus

The results (in table 3) shows there to be differences of between 1-3% across the board. These differences may seem insignificant and rather small but in order to put these figures into context the size of the samples must also be taken into consideration. It can be observed that the figures for the neutral corpus float in between the racist and anti-racist corpora. It is interesting to note that the racist corpus contains more adjectives and if we look at the adjective-noun ratio, (.42 for the racist and only .33 for the anti-racist) this tells us that racist discourse contains more qualifiers. The larger number of nouns in the anti-racist corpus may be indicative of a difference in register.

These differences, together with the results of Finn, Kushmerick and Smyth (Finn and Kushmerick, 2003; Finn *et al.*, 2002), are enough to justify training a SVM on POS datasets.

6. Conclusions and Future Research

In this paper we have described the field of text categorisation and its relationship to Information Retrieval and Machine Learning. We have introduced the various steps and processes involved in building a classifier and outlined the different choices to be made in doing so. We have introduced the PRINCIP problem and outlined how we are approaching the development of an automatic TC for racist texts on the Internet. We have presented our initial findings, evaluating the effectiveness of a SVM classifier trained on the bag-of-words representation. Though not 100% accurate we have shown it is possible to develop a TC system for racism on the web.

Our future research involves training Support Vector Machines for n-gram word sequences and POS – so as to identify the most effective method that will allow for the classification of racist documents on the web.

References

Apté C., Damerau F. and Weiss S.M. (1994). Automated Learning of Decision Rules for Text Categorization. *ACM Transactions on Information Systems.*

Chai K.M.A., Ng H.T. and Chieu H.L. (2002). Bayesian Online Classifiers for Text Classification and Filtering. In *Proceedings of the 25th annual international ACM SIGIR Conference on Research and Development in Information Retrieval:* 97-104.

Chandrasekar R. and Srinivas B. (1998). Using Syntactic Information in Document Filtering: A Comparative Study of Part-of-Speech Tagging and Supertagging. *Information Processing and Management,* vol. (34/5): 623-640.

Cohen W.W. and Singer Y. (1998). Context-sensitive Learning Methods for Text Categorization. In *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*.

Cristianini N. and Shawe-Taylor J. (2000). *Support Vector Machines and other kernel-based learning methods.* Cambridge University Press.

Drucker H., Shaharary B. and Gibbon D.C. (2001). Relevance Feedback using Support Vector Machines. In *Proceedings of the 18th International Conference on Machine Learning*.

Finn A., Kushmerick N. and Smyth B. (2002). Genre classification and domain transfer for information filtering. In *Proceedings of the European Colloquium on Information Retrieval Research (Glasgow).*

Finn A. and Kushmerick N. (2003). Learning to classify documents according to genre. In *IJCAI-2003 Workshop on Computational Approaches to Text Style and Synthesis (Acapulco*).

Fürnkranz J. (1998). *A Study Using N-gram Features for Text Categorization.*

Internet Content Rating Association. http://www.icra.org/ Last visited 20/10/2003.

Gibbon M. and Greevy E. (2003). *The Truth About Racism*. Faculty of Humanities Research Seminar, Dublin City University, April 2nd 2003.

Goller C., Löning J., Will T. and Wolff W. (2000). Automatic Document Classification: A thorough Evaluation of various Methods. In *Proceedings of Der zweite Workshop des MK² zumThema Automatische Dokumentenklassifikation*. http://www11.informatik.tu-muenchen.de/forschung/foren/mkmk/proceedings/dokumenten/goller.pdf – [last visited 23/07/03]

Joachims T. (1998). Text Categorization with Support Vector Machines: Learning with Many Relevant Features. In *Proceedings of the 10th European Conference on Machine Learning*.

Lechleiter H. and Greevy E. (2003). The Language of Open Racism: A Corpus Linguistic Analysis. In *Societas Linguistica Europea Conference*. Lyon, September 4th 2003.

Lewis D.D. (1992). Feature Selection and Feature Extraction for Text Categorization. In *Speech and Natural Language: Proceedings of a workshop:* 212-217.

Lewis D.D. and Ringuette M. (1994). A Comparison of Two Learning Algorithms for Text Categorization. In *Proceedings 3rd Annual Symposium on Document Analysis and Information Retrieval*.

Martin P. (2003a). So or Also: Racist use of adverbial phrases. In *Societas Linguistica Europea Conference.* Lyon, September 4th 2003.

Martin P. (2003b). Absolute Relatives – the language of online racial identity. In *Proceedings of the 30th Annual Symposium of the Royal Irish Academy*.

Mladenic D. and Grobelnik M. (1998). *Word sequences as feature in text-learning.*

Safer Internet Action Plan projects. http://www.saferinternet.org/filtering/projects.asp Last visited 20/10/2003.

Salton G. and Buckley C. (1988). Term-weighting Approaches in Automatic Text Retrieval. *Information Processing and Management,* vol. (24/5): 513-523.

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

Smeaton A.F. (1997). Information Retrieval: Still Butting Heads with Natural Language Processing? In Pazienza M.T. (Ed.), *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*. *Computer Science*, vol. (1299). Springer-Verlag Lecture Notes: 115-138.

Smeaton A.F. and Kelledy F. (1997). Automatic Phrase Recognition and Extraction from Text. In Furner J. and Harper D.J. (Eds), *Proceedings of the 19th Annual BCS-IRSG on IR Research,* Aberdeen.Springer Electronic Workshops in Computing.

Tan C.M., Wang Y.F. and Lee C.D. (2002). The Use of Bigrams to Enhance Text Categorization. *Information Processing and Management,* vol. (38/4): 529-546.

Van Dijk T. (1987). *Communicating Racism. Ethnic Prejudice in Thought and Talk*. Newbury Park, CA Sage.Wilbur J. and Sirotkin K. (1992). The Automatic Identification of Stop Words. *Journal of Information Science*, vol. (18): 45-55.

Wodak R. and Reisigl M. (2001). *Discourse and Discrimination. Rhetorics of racism and anti-Semitism*. Routledge.

Xelda http://www.mkms.xerox.com/ Last visited 15/01/2004.

Yang Y. and Wilbur J. (1996). Using Corpus Statistics to Remove Redundant Words in Text Categorization. *Journal of the American Society Information Science (JASIS).*

Yang Y. (1999). An Evaluation of Statistical Approaches to Text Categorisation. *Journal of Information Retrieval*, vol. (1): 67-88.

Yang Y. and Liu X. (1999). A re-examination of text categorization methods. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval*: 42-49.