The Anti-Social Tagger – Detecting Spam in Social Bookmarking Systems

Beate Krause Christoph Schmitz
Andreas Hotho Gerd Stumme
Knowledge & Data Engineering Group University of Kassel, Germany
http://www.kde.cs.uni-kassel.de

ABSTRACT

The annotation of web sites in social bookmarking systems has become a popular way to manage and find information on the web. The community structure of such systems attracts spammers: recent post pages, popular pages or specific tag pages can be manipulated easily. As a result, searching or tracking recent posts does not deliver quality results annotated in the community, but rather unsolicited, often commercial, web sites. To retain the benefits of sharing one's web content, spam-fighting mechanisms that can face the flexible strategies of spammers need to be developed.

A classical approach in machine learning is to determine relevant features that describe the system's users, train different classifiers with the selected features and choose the one with the most promising evaluation results. In this paper we will transfer this approach to a social bookmarking setting to identify spammers. We will present features considering the topological, semantic and profile-based information which people make public when using the system. The dataset used is a snapshot of the social bookmarking system BibSonomy and was built over the course of several months when cleaning the system from spam. Based on our features, we will learn a large set of different classification models and compare their performance. Our results represent the groundwork for a first application in BibSonomy and for the building of more elaborate spam detection mechanisms.

Categories and Subject Descriptors

H.3 [**Information Storage and Retrieval**]: Miscellaneous

General Terms

Algorithms, Experimentation

Keywords

spam detection, folksonomy, Web 2.0

AIRWeb '08, April 22, 2008 Beijing, China.

1. INTRODUCTION

Web spam detection is a well known challenge for search engines. Spammers add specific information to their web sites that solely serve to increase the ranking and not the quality or content of a page. They thereby increase the traffic to their web sites be it for commercial or political interests or to disrupt the service provided. Ranking algorithms need to detect those pages using elaborate techniques.

Not only search engines need to fight with malicious web content. Social bookmarking systems also have become an attractive place for posting web spam. These systems allow users to annotate and share bookmarks. Within the last few years, a large community of users who add, share and work with the content of these systems has evolved. del.icio.us[1] is a popular example, but also other systems targeting more specific communities such as the scholarly world, exist (Connotea[2], CiteULike[3], BibSonomy[4]).

Spammers (mis)use the popularity and the high PageRank of social bookmarking systems for their purposes. All they need is an account; then they can freely post entries which bookmark the target spam web site. In recent months, different spamming techniques have been developed to frequently show up on popular sites, recent post sites or as highly ranked posts after the search for a specific tag. For instance, spammers request several accounts and publish the same post several times. Besides appearing on the recent post page, the bookmark may show up on the popular page, since "many" users have considered the bookmark. Another technique is to add diverse tags to the bookmark or use popular tags.

In order to retain the original benefits of social bookmarking systems, techniques need to be developed which prevent spammers from publishing in these systems, or at least from having their malicious posts published. The problem can be considered as a binary classification task. Based on different features that describe a user and his posts, a model is built from training data to classify unknown examples (on a post or user level) either as ("spam" or "non-spam"). As we consider "social" systems in which users interact with each other and one incentive to use the system is to see and be seen, an exclusion of non-spammers from publishing is a severe

[1] http://del.icio.us
[2] http://www.connotea.org
[3] http://www.citeulike.org
[4] http://www.bibsonomy.org

error which might prevent the user from further participation. Similar to other spam detection settings, this problem needs to be taken into consideration when classifying users.

The adaptation of classification algorithms to this task consists of two major steps. The first one is to select features for describing the users. The second step is the selection of an appropriate classifier for the problem. In this paper, we introduce a set of initial features that can be used for spam classificiation. These features are evaluated with well-known classifiers (SVM, Naive Bayes, J48 and logistic regression) against a simple baseline of representing a user by the usage of tags.

This article is organized as follows. In Section 2 we will discuss related work in the field of spam detection. Section 3 introduces the concept of a folksonomy and the dataset. Section 4 describes the setting and the features of the classification task. In Section 5 the results are presented and Section 6 concludes our findings and discusses future work.

2. RELATED WORK

One of the first publications dealing with folksonomies, also referred to as tagging systems, is [18]. The authors of [16, 19] first describe the structure of these systems which can be viewed as a tripartite graph, whereby the nodes are composed of users, tags and resources. In [6] a first analysis of del.icio.us. is provided. [7, 17, 18, 23] give further insights into the structure and dynamics of tagging systems. Rankings and recommender systems for folksonomies are proposed in [11, 12].

Research on spam detection in social media has been conducted by the blog and wikipedia community. Methods to detect comment spam and spam blogs have been proposed by [13, 14, 20]. A first reference to the spam detection problem in folksonomies is given in [2]. [9, 15] are the first to deal with spam in tagging systems explicitly. The authors identify anti-spam strategies for tagging systems and construct and evaluate models for different tagging behaviour. In contrast to [9, 15] we present a concrete study using machine learning techniques to combat spam on a real-world dataset. We focus on social bookmarking systems as tagging systems and present features derived from the identity of contributors, the semantic of tags and the link or behaviour analysis of users.

Considering our task of classifying users posting web sites, web spam detection is a further related area. [1, 5, 21, 25] represent some of the research on feature selection and classification. Many features and classical machine learning techniques can be transferred from this area (e. g., content based and linked based spam detection), however the nature of social tagging systems allows for further features such as profile information of the active user, the usage of tags and co-occurrences of tags and resources among spammers and non-spammers.

3. BASICS

In this section, we will formally define the structure of a social bookmarking system, the phenomenon of spam and introduce the dataset used for evaluation.

3.1 Spam in folksonomies

Social bookmarking systems are collaborative tagging systems which allow users to add keywords to shared content (bookmarks). Their underlying data structure is called a *folksonomy*. We will make use of the formal definition of folksonomies that we provided in [11].

Definition A *folksonomy* is a tuple $\mathbb{F} := (U, T, R, Y)$ where U, T, and R are finite sets, whose elements are called *users*, *tags* and *resources*, resp., and Y is a ternary relation between them, i. e., $Y \subseteq U \times T \times R$. The elements of Y are called *tag assignments (TAS)*. A *post* is a triple (u, T_{ur}, r) with $u \in U$, $r \in R$, and $T_{ur} := \{t \in T \mid (u, t, r) \in Y\}$ such that $T_{ur} \neq \emptyset$.

As proposed in [9], we consider spam in folksonomies as (1) content which legitimate users do not wish to share and (2) content which is tagged in a way to mislead other users. The first part refers to web spam: For commercial or political interests, to simply distract the system, or to run down other companies, spammers try to score high with their web sites by posting their content in the system. The second part considers the tagging behaviour: spammers add keywords that do not match the content of the bookmarks. Again the motivation may be self-promotion (users looking for a specific tag will receive advertisements) or to distract and destroy the serendipitous browsing facilities that make folksonomies special.

Figure 1 shows examples of spam from the bookmarking system BibSonomy: The first and the last entry advertise medical treatments.

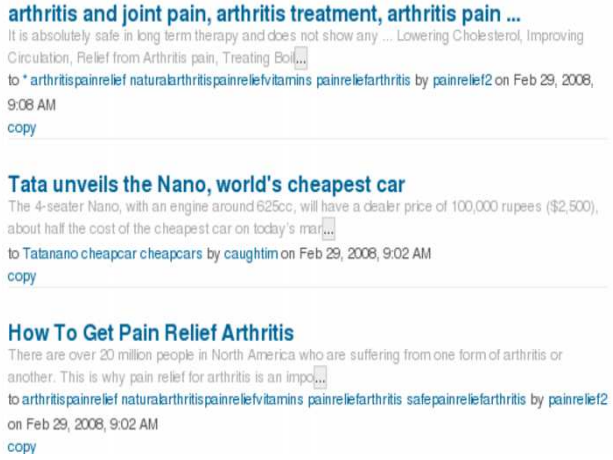


Figure 1: Examples of spam posts in the social bookmarking system BibSonomy

3.2 Dataset

BibSonomy[4] is a social bookmarking site that allows users to share bookmarks (i. e., URLs) as well as publication references [10]. The authors are part of the team behind BibSonomy and take part in the (up to now manual) removal of spam. The resulting spam dataset will be used in this paper for evaluation.[5] It comprises users, tags, resources

[5]BibSonomy dumps (without private information) are available for scientific purposes from the BibSonomy site.

Table 1: Figures of the dataset used for evaluation

Spammers	Users	Tags	Resources	TAS
18,681	1,411	306,993	1,219,053	8,709,417

and a user's profile information of all BibSonomy users until the end of 2007 (Table 1). Considering only bookmarks, the system consists of 1,411 legitimate users and 18,681 users that were flagged as spammers.

The dataset was created in the course of the year 2007. In order to prevent spammers from publishing, the system administrators created a simple interface, that allows authorized users (mainly the system administrators and some researchers) to flag users as spammers. If a user is flagged as a spammer, his posts are no longer visible for other users. In particular, this means that general pages such as the popular page do not show a spam post anymore. However, spammers can still see and manage their own posts on their own user page.

The flagging of spammers by different evaluators is a very subjective process. There were no official guidelines, but a common sense of what distinguishes users from spammers, based on the content of their posts. To narrow down the set of potential spammers, the evaluators normally looked at a user's profile (e. g., name, e-mail address), the composition of posts (e. g., the semantics of tags, the number of tags) before assessing the content of the bookmarked web sites. Borderline cases were handled from a practical point of view. BibSonomy intends to attract users from research, library and scholarly institutions. Therefore, entries referring to commercial advertisements, Google Ad clusters, or the introduction of specific companies are considered as spam. The marked spammers are shown on the administration interface and can be unflagged by all authorized users. However, evaluators rarely cross-checked the evaluations, so a certain amount of noise in the dataset is therefore probable.

Another option of spam prevention in social bookmarking systems is to define spam on the post level instead of the user level. This would mean that individual posts are marked as spam or not, whereas currently all or none of the posts of a user are marked as spam. Our justification for the latter approach is that users either have a malicious intention and use non-spam posts to hide their motivations or are "clean" users. This approach reduces in particular the workload for the administration team when manually checking the content for spam. In future work, however, it will be interesting to consider spam detection on the post level as well.

4. FRAMEWORK

An automatic classification of spammers demands features that distinguish legitimate users from malicious ones. In this section we describe the features we have chosen and outline the basic setting to evaluate those features.

4.1 Evaluation

We generated training and test instances from the dataset described in Section 3.2. As the practical objective of this evaluation is to predict spam in the next month/week/day,

Table 2: Training and Test dataset

	Users	Tags	Resources	TAS
Train	17,202	282,473	1,097,458	7,904,735
Test	2,890	49,644	121,595	804,682

Table 3: Confusion matrix

Actual/Labeled	Spam	Non-spam
Spam	TP	FP
Non-spam	FN	TN

we split the instances chronologically: the training set comprehends all instances until the end of November 2007, the test set all instances of the month December 2007 (Table 2). In future work, we want to evaluate more granular splits.

The results of a classification algorithm can be presented in a confusion matrix, see Table 3. TP is the number of spam instances that were correctly classified, FP is the number of non-spam instances that were incorrectly classified as spam, FN is the number of instances that were incorrectly classified as non-spam, and TN denotes those instances that were correctly classified as non-spam.

Precision is defined as $\frac{TP}{TP+FP}$, the true positive rate (or recall) as $\frac{TP}{TP+FN}$ and the false positive rate as $\frac{FP}{FP+TN}$. For our evaluation, we consider the F-measure which is the harmonic mean of precision and recall $F = \frac{2PR}{P+R}$ and the area under a ROC curve (AUC). AUC estimates the probability that a randomly chosen positive instance will be ranked higher than a randomly chosen negative instance [4]. It assesses the portion of the area of the unit square under the *receiver operating characteristics (ROC) curve*. These curves show the relative tradeoffs between benefits (true positives rates) and costs (false positives rates), see Figures 2, 4, and 5(a)–6(b). The curves are plotted according to a predetermined order of the test instances – for instance, the Naive Bayes classifier provides an instance probability which can be used for such a ranking. An advantage of using ROC curves for evaluation is that these curves are independent of the underlying class distribution. Therefore, the skewed class distribution as it is present in our dataset, is not considered. Another – more practical – reason for considering the ROC curve is that we want to leave the obvious decisions to the classifier, and to control the suggested classification of the borderline cases before finalizing the decision. The former ones are those at the beginning of the ROC curve. The steeper the curve starts, the fewer miss-classifications occur. Once the curve becomes less steep, we have to control the outcome of the classifier.

4.2 Baseline

The simplest baseline we can consider is to always predict the majority class in the data, in our case "spammer". In our skewed dataset, this would yield a precision of 0,965, and a F-measure of 0,982 (for the spam class). However, all non-spammers would be classified as spammers.

A more substantial baseline is to consider the tags used to

Table 4: Baseline with all tags as features (frequency)

	Spam	Non-Spam
Spam	466	2324
Non-Spam	0	100

Table 5: Baseline with all tags as features (tfidf)

	Spam	Non-Spam
Spam	530	2260
Non-Spam	0	100

describe a resource as features and use a classifier that has been shown to deliver good results for text classification such as Naive Bayes. Each user u can then be represented as a vector $\vec{u}$ where each dimension corresponds to a unique tag t. Each component of $\vec{u}$ is assigned a weight. We consider two different settings. In the first case, the weight corresponds to the absolute frequencies the tag t occurs with the user u. In the second case, each tag is assigned a tfidf value. The tfidf value for a specific tag t_i in a post p of user u is defined as $tfidf(i;p,u) = \frac{tf_{ip}}{maxtf_{jp}} log \frac{|P|}{|P_i|}$ where tf_{ip} denotes the tag frequency of the tag t_i in the post p, $maxtf_{jp}$ is the maximum frequency of a tag t_j in this post p, $|P|$ is the total number of posts, and $|P_i|$ the number of posts which contain the tag t_i.

Tables 4 and 5 show the TP, FP, FN, TN values for the absolute frequencies and the tfidf values. When computing the baseline with the tfidf measure, the misclassification of spammers slightly improves, so that more spammers are identified. The ROC area value for the frequency baseline is 0.801, the F-measure 0.286. For the tfidf baseline the ROC area value is 0.794, the F-measure 0.319. Figure 2 shows the ROC curve progression of the two baselines. The curves are similar at the beginning. The tfidf-baseline curve shows a steeper progression in the beginning, but is then exceeded by the frequency-baseline.

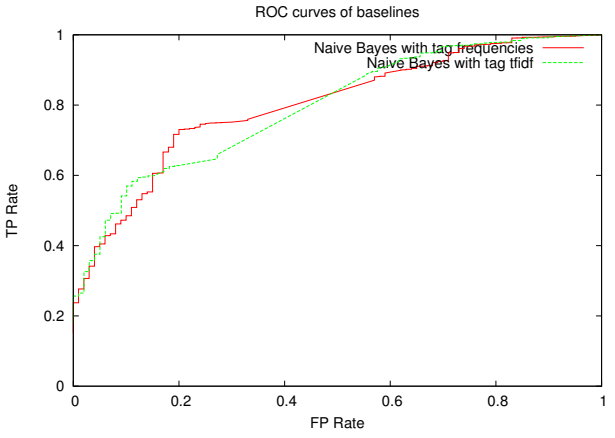


Figure 2: ROC curves of the frequency and tfidf tag features

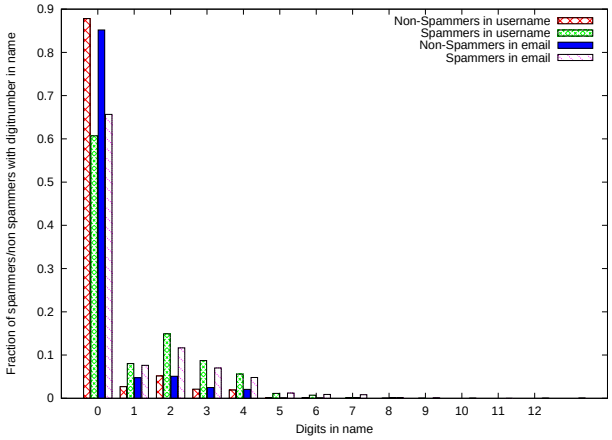


Figure 3: Histogram of the number of digits in the username and email of spam vs. nonspam users

4.3 Features

Overall we considered 25 features. In order to describe them, we identified four different feature categories. The first one comprehends all information in the user's profile. The second category refers to the location a user publishes bookmarks from, or which is given as the domain in his e-mail address. The third category concerns the interaction with the system and the last one considers semantic features hidden in the choice and usage of tags. Tables 6–9 summarize all features.

An instance in the training or test set can be seen as a vector with all features and the correspondig values. Each feature is normalized over the total set of users by dividing a user's feature value by the difference of the maximum and the minimum value of this specific feature.

4.3.1 Profile features

The profile features are extracted from a user's profile which he or she reveals when requesting an account in BibSonomy. Table 6 show the features corresponding to a user's profile.

Most of the fields to fill in at registration are not obligatory, however, users need to indicate a name and a valid e-mail-address. Spammers often differentiate from normal users in that they use names or e-mail addresses with many numbers. For instance, the spammer's names of the example in Figure 1 are "styris888" and "painrelief2". Figure 3 shows the histogram of the spam/non-spam distribution of the number of digits in the username and the email address (*namedigit*, *maildigit*). As can be seen, besides the peak at the bin with 0 numbers, spammers show further peaks at the two-digit bin. The *namelen*, *maillen* and *realnamelen* features refer to the length of the usernames, email addresses and realnames. The *realname2* and *realname3* features are binary and set to one, if the user has indicated two or three names. The features were derived from the observation, that legitimate users often register with their full names.

4.3.2 Location based features

Location based features refer to describing the user's location and domain. Table 7 summarizes the location based

features.

Often, the same spammer uses several accounts to publish the same content. These accounts show the same IP address when they are registered. Thus, if one user with a specific IP is already marked as a spammer, the probability that other users with the same IP are also spammers is higher (*spamip*). When considering the users in the training dataset, 6,637 of them have at least one ip address in common with another spam user. Out of these, 6,614 users are marked as spammers. The same phenomenon holds for users of specific domains (*domaincount, tldcount*). The probability that a user who is from a rare domain which hosts many spammers is also a spammer is higher than average (and vice versa). For instance, 16 users have registered with the domain "spambob.com" and 137 with the domain "rhinowebmail", all of which were classified as spammers.

4.3.3 Activity based features

Activity properties (Table 8) consider different kinds of user interactions with the social bookmarking system. While normal users tend to interact with the system first thing after the registration (e. g., by posting a bookmark), spam users often wait a certain time after they submit their first post. This timelag can be considered when characterizing spam (*datediff*).

Furthermore, the number of tags per post varies (*tasperpost*). Spammers often add many different tags to a resource, be it to show up more often when searching for many different tags, be it to include "good" tags in order to confuse spam detection mechanisms. Considering the BibSonomy dataset, spammers add in average eight tags to a post, while non-spammers add four. The average number of TAS (see definition in Section 3) is 470 for spammers and 334 for users (*tascount*).

4.3.4 Semantic features

Semantic features (Table 9) relate to the usage and content of the tags which serve as an annotation for a bookmark.

There are several "simple" properties which we found when manually cleaning the system from spam. For instance, 1,916 users added "$group=public" as a tag or part of a tag to a resource. 1,914 of these users are spammers (*grouptag*). This specific tag is used by a software to generate spam in social bookmarking systems. We also have a blacklist of tags which contains tags that are very likely to describe a spam post. For instance, "pornostars", "jewelry" or "gifts" are contained in this list. One feature is to calculate the ratio of such spam tags to all tags published by a specific user (*spamtag*).

Cooccurrence information can be extracted by building an undirected graph with users as nodes. A link between two users u_i and u_j exists if they share at least one resource or at least one tag or at least one tag-resource pair.

For our feature calculation, we considered each case twice, resulting in the six features *co(no)spam(r/t/tr)*. In the first case, a link between u_i and u_j is only set if u_j is a spammer, in the second a link is set if u_j has been marked as a non-spammer. The assumption is that spammers show high values in the first case, as they apply the same vocabulary and resources other spammers use; non-spammers show higher values in the second case. To give an idea of the range of these values: a non-spammer shares resources with about 18 other non-spammers in average. A spammer only shares resources with about 0.5 non-spammers in average. We also computed the ratio of each spam and non-spam pair (*spamratiot, spamratior, spamratiotr*).

Table 6: Profile features

Feature name	Description
namedigit	name contains digits
namelen	length of name
maildigit	email address contains digits
maillen	length of mail address
realnamelen	length of realname
realnamedigit	realname contains digits
realname2	two realnames
realname3	three realnames

Table 7: Location based features

Feature name	Description
domaincount	number of users in the same domain
tldcount	number of users in the same top level domain
spamip	number of spam user with this IP

5. EXPERIMENTS

We selected different classification techniques to evaluate the features we introduced in the previous section. For the first three algorithms, we used the Weka implementation [24], for the SVM we used the LibSVM package [3]

Naive Bayes, a statistical classifier, has been shown to be successful and simple in classification. The algorithm is based on the Bayes' theorem using the joint probabilities of sample observations to estimate the conditional probabilities of classes given an observation [8]. The C4.5 decision tree classifier [22], in Weka implemented as J48, builds a binary classification tree. Determined by a splitting criterion, attributes are selected as branching points that separate the two classes in the training dataset best. Logistic Regression is a generalized linear model to apply regression to categorical (in our case binary) variables. Finally, support vector machines (SVMs) aim at searching for a hyperplane that separates two classes of data with the largest margin (the margin is the difference between the hyperplane and the point closest to it).

Table 8: Activity based features

Feature name	Description
datediff	difference between registration and first post
tasperpost	number of tags per post
tascount	number of total tags added to all posts of this account

Table 9: Semantic features

Feature name	Description
co(no)spamr	user cooccurrences (related to resources) with (non) spammers
co(no)spamt	user cooccurrences (related to tags) with (non) spammers
co(no)spamtr	user cooccurrences (related to tag-resources pairs) with (non) spammers
spamratio(r/t/rt)	ratios of spam/non spam cooccurrences
grouptag	number of times 'group=public' was used
spamtag	ratio of spam tags to all tags of a user

Table 10: Evaluation values all features

Classifier	ROC area	F1	FP	FN
Naive Bayes	0.906	0.876	14	603
SVM	0.936	0.986	53	23
Logistic Regression	0.918	0.968	30	144
J48	0.692	0.749	11	1112

We tried different scenarios. Each will be described in the following paragraphs, together with the evaluation outcomes.

5.1 Classification combining all features

Table 10 shows the ROC area, F1 measure, and the absolute false positive values and false negative values for all algorithms, based on all features. Figure 4[6] depicts the ROC curves for all classifiers. The best classifier with an AUC of 0.936 is the SVM, followed by the logistic regression classifier. Even though the progression of the SVM's ROC shows that the false positive instances are the ones with less probability, over half of the non-spammers are ranked as spammers. Section 5.3 therefore introduces costs for misclassifying non-spammers. The AUCs of the two baselines (0.801 and 0.794) yield lower results.

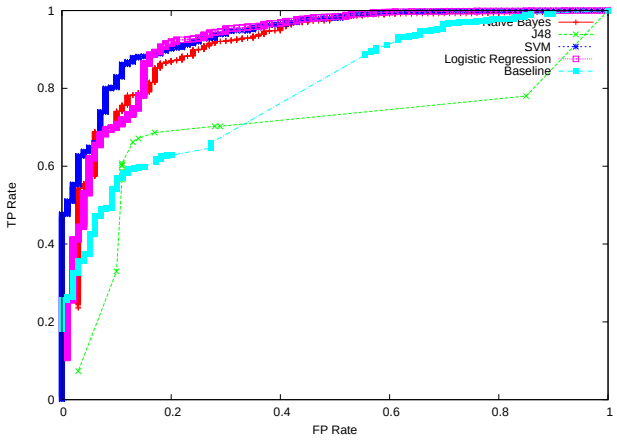


Figure 4: ROC curves of classifiers considering all features. The steepest progression shows the SVM classifier.

5.2 Feature groups

In order to find out about the contribution of the different features, we have analyzed each feature group separately.

[6]We only included one baseline (tfidf) to reduce the number of curves.

Table 11: Evaluation values of feature groups

Features	ROC area	F1
Profile features (log. reg.)	0.77	0.982
Location features (SVM)	0.698	0.407
Activity features (SVM)	0.752	0.982
Semantic features (J48)	0.815	0.981
Cooccurrence features (log. reg.)	0.927	0.985

The best results are given by the cooccurrence features (co(no)spamr, co(no)spamt, co(no)spamrt, spamratior, spamratiot, spamratiotr). Figures 5(a)–6(b) present the ROC curves and evaluation values. The semantic features were split in two subgroups – cooccurrence features (and the ratios) and the spamtag/grouptag.

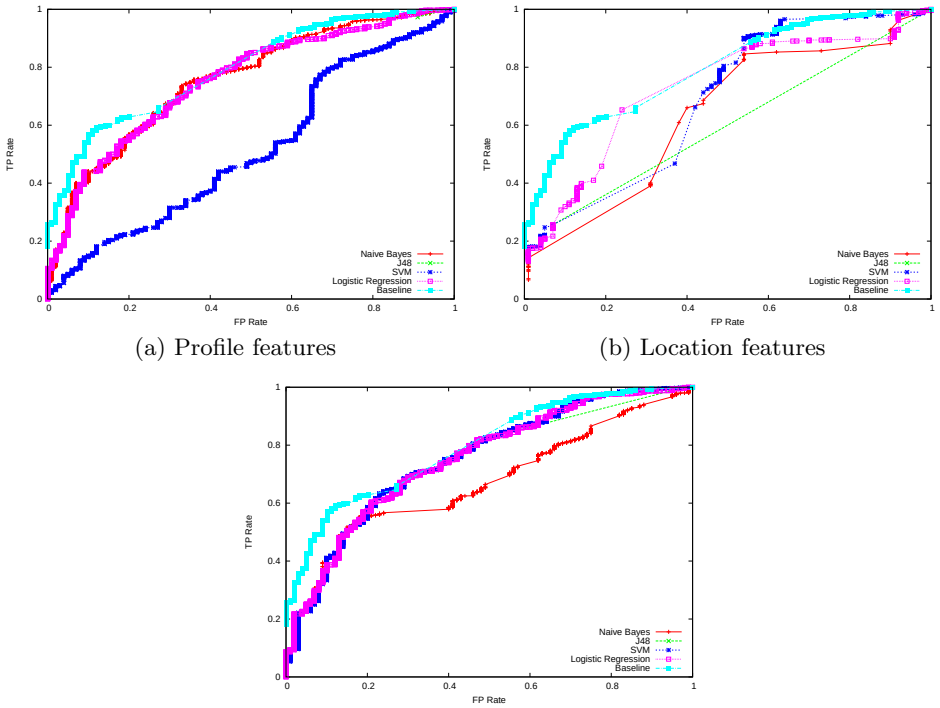
Table 11 shows, for each feature group, the evaluation values of the algorithm which optimizes the ROC area. Interestingly, there is not a single algorithm which performs best on all feature groups.

Overall, none of the feature groups reaches the classification performance obtained when combining the features. This shows that in our setting, no dominant type of spam indicator exists. A variation of different kinds of information is helpful. The cooccurrence features describing the usage of a similar vocabulary and resource usage are most promising.

5.3 Costs

The ROC curves inherently introduce costs in that they order instances according to classification probabilities. However, most classifiers do not use cost information when building their models. As seen above, the SVM for the combination of all features nearly perfectly separates 40% of spammers from non-spammers until an error takes place. However, over half of the non-spammers are classified as spammers in the final result.

In order to penalize the wrong classification of non-spammers, we introduced cost sensitive learning [24]. Before a model is learned on the training data, the data is reweighted to increase the sensitivity to non-spam cases (i. e., the data consists of more non-spam classified instances than before). We experimented with different cost options and found that a penalty of ten times higher than the neutral value (one) delivered good results for the SVM. We also recalculated the other classifiers using cost options. Table 12 shows an overview of the changed F1 and false positive rates of classification using all features. Cost-sensitive learning on all features with logistic regression returns the best results.



(a) Profile features (b) Location features

(c) Activity features

Figure 5: ROC curves of the different feature groups

Table 12: Evaluation with a cost sensitive classifier

Classifier	F1	FP
SVM	0.924	15
J48	0.794	11
Logistic Regression	0.927	12
Naive Bayes	0.855	11

6. CONCLUSIONS

This paper introduced a variety of features to fight spam in social bookmarking systems. The features were evaluated with well-known machine learning methods. Combining all features shows promising results exceeding the AUC and F1 measure of the selected baseline. Considering the different feature groups, cooccurrence features show the best ROC curves.

Our results support the claim of [9], that the problem can be solved with classical machine learning techniques – although not perfectly. The difference to web spam classification are the features applied: on the one hand, more information (e. g., email, tags) is given, on the other hand spammers reveal their identity by using a similar vocabulary and resources. This is why cooccurrence features tackle the problem very well.

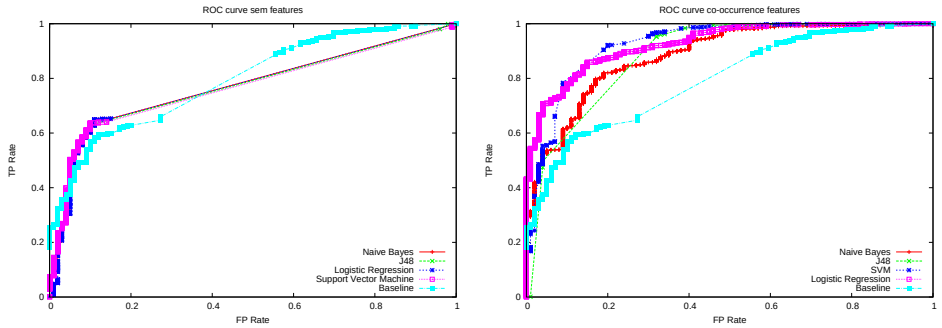
Several issues considering our approach need to be discussed. First of all, a switch from the user level to the post level is an interesting next step to consider. This would also facilitate the handling of borderline cases, as users, though some of their posts were flagged as spam, can still participate. A consideration of a multiclass classification introducing classes in between "spam" and "non spam" or a ranking of classified instances may also help to identify those borderline users a moderator needs to manually classify. A further issue regards the evaluation method chosen. In future work, we want to consider more than one chronological separated training/test set. This may also help to reduce the ratio between training and test data. The large ratio between spam and non-spam users could be reduced by identifying spammers which have created several user accounts and therefore are counted several times. Finally, the feature groups presented have been intuitively chosen – they may be extended in different ways. We also think of adding more features such as topological information, clustering coefficients and tag similarity in posts.

Overall, our contribution represents a first step towards the elimination of spam in social bookmarking systems using machine learning approaches. Currently, we are constructing a spam detection framework to flexibly combine features and learning algorithms. Besides the practical need to eliminate spam, we intend to use this platform to develop and evaluate further social spam detection mechanisms.

Acknowledgement. This research has been partly supported by the TAGora project funded by the Future and Emerging Technologies program (IST-FET) of the European Commission under the EU RD contract IST-034721.

7. REFERENCES

[1] C. Castillo, D. Donato, A. Gionis, V. Murdock, and F. Silvestri. Know your neighbors: web spam



(a) Semantic features (grouptag, spamtag) (b) Semantic features (cooccurrences + ratios)

Figure 6: ROC curves of the semantic feature groups

detection using the web topology. In *Proc. SIGIR '07*, pages 423–430, New York, NY, USA, 2007. ACM.

[2] C. Cattuto, C. Schmitz, A. Baldassarri, V. D. P. Servedio, V. Loreto, A. Hotho, M. Grahl, and G. Stumme. Network properties of folksonomies. *AI Communications*, 20(4):245 – 262, 2007.

[3] C.-C. Chang and C.-J. Lin. Libsvm: a library for support vector machines (version 2.31).

[4] T. Fawcett. An introduction to roc analysis. *Pattern Recogn. Lett.*, 27(8):861–874, 2006.

[5] Q. Gan and T. Suel. Improving web spam classifiers using link structure (s). In *Proc. AIRWeb '07*, pages 17–20, New York, NY, USA, 2007. ACM.

[6] S. Golder and B. A. Huberman. The structure of collaborative tagging systems. *Journal of Information Science*, 32(2):198–208, April 2006.

[7] T. Hammond, T. Hannay, B. Lund, and J. Scott. Social Bookmarking Tools (I): A General Review. *D-Lib Magazine*, 11(4), April 2005.

[8] J. Han and M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann, September 2000.

[9] P. Heymann, G. Koutrika, and H. Garcia-Molina. Fighting spam on social web sites: A survey of approaches and future challenges. *IEEE Internet Computing*, 11(6):36–45, 2007.

[10] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. BibSonomy: A social bookmark and publication sharing system. In *CS-TIW '06*, Aalborg, Denmark, July 2006. Aalborg University Press.

[11] A. Hotho, R. Jäschke, C. Schmitz, and G. Stumme. Information retrieval in folksonomies: Search and ranking. In *Proc. ESWC '06*, pages 411–426, Budva, Montenegro, June 2006. Springer.

[12] R. Jäschke, L. B. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme. Tag recommendations in folksonomies. In *Proc. PKDD '07*, Berlin, Heidelberg.

[13] P. Kolari, T. Finin, and A. Joshi. SVMs for the Blogosphere: Blog Identification and Splog Detection. *AAAI Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.

[14] P. Kolari, A. Java, T. Finin, T. Oates, and A. Joshi. Detecting Spam Blogs: A Machine Learning Approach. *AAAI '06*, 2006.

[15] G. Koutrika, F. A. Effendi, Z. Gyöngyi, P. Heymann, and H. Garcia-Molina. Combating spam in tagging systems. In *Proc. AIRWeb '07*, pages 57–64, New York, NY, USA, 2007. ACM.

[16] R. Lambiotte and M. Ausloos. Collaborative tagging as a tripartite network. *Lecture Notes in Computer Science*, 3993:1114, Dec 2005.

[17] B. Lund, T. Hammond, M. Flack, and T. Hannay. Social Bookmarking Tools (II): A Case Study - Connotea. *D-Lib Magazine*, 11(4), April 2005.

[18] A. Mathes. Folksonomies – Cooperative Classification and Communication Through Shared Metadata, December 2004. http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html.

[19] P. Mika. Ontologies are us: A unified model of social networks and semantics. In *Proc. ISWC '05*, LNCS, pages 522–536. Springer, 2005.

[20] G. Mishne, D. Carmel, and R. Lempel. Blocking blog spam with language model disagreement. In *Proc. AIRWeb '05*, pages 1–6, New York, NY, USA, 2005. ACM.

[21] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *Proc. WWW '06*, pages 83–92, New York, NY, USA, 2006. ACM.

[22] J. R. Quinlan. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[23] S. Sen, A. M. Lam, Shyong K.and Rashid, D. Cosley, D. Frankowski, J. Osterhouse, M. F. Harper, and J. Riedl. tagging, communities, vocabulary, evolution. In *Proc. CSCW '06*, pages 181–190, New York, NY, USA, 2006.

[24] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann, October 1999.

[25] B. Wu and B. D. Davison. Detecting semantic cloaking on the web. In *Proc. WWW '06*, pages 819–828, New York, NY, USA, 2006.