Automatically Generated Spam Detection Based on Sentence-level Topic Information

Yoshihiko Suhara, Hiroyuki Toda, Shuichi Nishioka, Seiji Susaki
NTT Service Evolution Laboratories, NTT Corporation
1-1 Hikari-no-oka, Yokosuka-Shi, Kanagawa, 239-0847 Japan
{suhara.yoshihiko, toda.hiroyuki, nishioka.shuichi, suzaki.seiji}@lab.ntt.co.jp

ABSTRACT

Spammers use a wide range of content generation techniques with low quality pages known as content spam to achieve their goals. We argue that content spam must be tackled using a wide range of content quality features. In this paper, we propose novel sentence-level diversity features based on the probabilistic topic model. We combine them with other content features to build a content spam classifier. Our experiments show that our method outperforms the conventional methods.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

Keywords

Spam detection, spam feature, topic model

1. INTRODUCTION

Web spam is considered to be threads to not only search engines but also any application that uses web pages as information sources. There is a wide variety of web spam [9][18]. The main purpose of web spam is to lead users from the search engines to the web page. Thus, the design of web spam takes account of the search engines' ranking algorithms.

Web spam is divided into two types: content spam and link spam [9]. In this paper, we focus on *automatically generated spam*, a type of content spam, since it accounts for a substantial fraction of web spam [6]. These are generated by copying from other web pages, or by using natural language processing techniques such as language model or text summarization. This characteristics makes it difficult to distinguish these spam from non-spam (ham) pages. In the paper, we propose a novel method to extract features from documents based on content analysis, and then employ the machine-learning based approach to detect web spam.

Content spam detection with supervised machine learning extracts the features that well capture the spam characteristics [14] from each web page and then build a spam detection classifier based on the labeled dataset. The labeled dataset contains spam/ham labels for each web page, and is usually prepared by human annotator in advance. Known feature extraction methods include language-model based feature [14][13] and statistical features [15].

However, existing methods fail to accurately capture the characteristics of automatically generated spam because it is created by copying ham pages. This property makes the language-based model ineffective. Our solution is to model topic transitions; it is an effective way of capturing the unnatural aspects of automatically generated spam and thus detecting them.

Pavlov et. al. [15] proposed a feature that models topical diversity and uniformity by using Latent Dirichlet Allocation (LDA) [3]. They reported improved spam detection accuracy with their proposed feature. However, they used the WEBSPAM-UK2007 dataset, which contains various kind of web spam, to evaluate their method so that the performance of their method in detecting automatically generated spam is still unclear. In addition, their method might not be suitable to capture topic shift over the sentences in the document because their method uses the topic distribution on the whole document.

In this paper, we propose a novel method to extract features based on sentence-level topic information. Our method first creates LDA with a ham corpus, and then applies the LDA to the unseen documents to infer the topic distribution of the sentences. Specifically, we propose two methods to capture the unnatural topic distribution over the sentences in the document. The first method uses topic assignment approach, which assigns a single topic to each sentence with the LDA model. We introduce a topic-voting heuristics using each word-topic assignment to assign a suitable topic ID to each sentence. Since we obtain a sequence of topic IDs for each document, we can model the topic transition over the sentences. The second method utilizes the topic distribution of each sentence to calculate the difference in topic distribution between adjacent sentences.

We prepared a spam blog dataset that contains automatically generated spam and conducted preliminary experiments to confirm that conventional language-model based features and conventional topical diversity features cannot capture the characteristics of automatically generated spam. We also verify that our proposed features are more accurate in detecting the automatically generated spam.

The major contributions of this paper include:

- We propose a novel method to extract features based on the sentence-level topic transition over the sentences in the document.

WWW 2013 Companion, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2038-2/13/05.

- We create a spam blog dataset to verify that our proposed features can detect the automatically generated spam with higher accuracy.

The rest of the paper is organized as follows: We review related works in Section 2. We describe the topic model with LDA and the proposed method in Section 3. We report the preliminary experiments and evaluation in Section 4 and conclude the paper in Section 5.

2. RELATED WORK

Many spam detection techniques have been proposed in recent years. Some methods were developed through competitions such as Web Spam Challenge[1], and ECML/PKDD Discovery Challenge.

Link-based web spam detection is a basic approach to detecting automatically generated web spam pages. Techniques like TrustRank [10] minimize the impact of spam pages on ranking. This method can detect web spam pages without analyzing page contents. In this paper, we don't use any link spam detection technique as we focus on the content itself. We note that link-based techniques can also be used to build spam classifiers.

Near-duplicate detection is another way of detecting automatically generated web spam. Fatterly et al. proposed using duplicate analysis to detect web spam [6]. They measured phrase-level duplication of content across the web. Vallés et al. [19] proposed a duplicate-detection-based SMS spam detection method. Unfortunately, these techniques demand that a large amount of documents be stored. Our method does not use inter-document information but uses intra-document information as a feature to build a spam classifier.

There are several papers that propose spam detection methods based on topic models. Bíró et al. developed the multi-corpus LDA [2] that builds separate LDA models for spam and ham, and uses the topic weights as classification features. They also developed linked LDA [1] which incorporates the link data into LDA model for spam classification. Pavlov et al. [15] proposed topical uniformity/diversity features based on the topic distribution calculated using LDA. These methods apply the topic model to the whole document. Our method differs from theirs in that it exploits the information contained within topical variation over sentences.

Erdélyi et al. [4] conducted a comparative study on machine learning techniques for the spam detection task. For a comprehensive discussion, they verified several machine learning technique with the features proposed in conventional studies. In this paper, we propose novel features for spam detection that can be used in general machine learning algorithms.

Jo et al. [11] proposed a modified LDA called Sentence-LDA (SLDA). SLDA imposes the constraint that all words in a sentence are generated from one topic. Because of this assumption, the generative process differs from that of LDA. This forces SLDA to use the plain Gibbs sampling method, which has high computation costs. Although our motivation is the same as theirs, we use the different approach of assigning a topic to each sentence based on ordinary LDA.

Our idea is close to the text segmentation method based on topic models by Riedl et al. [16]. They use LDA to assign a topic distribution to each sentence to calculate the topic difference between adjacent sentences. We use not only the topic distribution but also use the topic itself assigned to each sentence. Thus, we can use the topic of each sentence as a discrete label. This idea allows the use of the sentence-topic n-gram model.

[1] `http://webspam.lip6.fr/`

3. METHOD

In this section, we briefly present spam detection with supervised machine learning. Then, we describe a topic model with LDA and show how to estimate the parameters. After that, we introduce two approaches that capture the sentence-level topic information.

Our goal is to create the features that contribute to better classification performance in detecting content spam. This paper build a spam classifier in the supervised learning framework, and so we need training data that consists of instances labeled spam (+1) or ham (-1). Training data D consists of N instances $\{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, where $\mathbf{x} \in \mathbb{R}^m$ is an m-dimensional feature vector. Text information such as bag-of-words can be used as features. We can use any standard supervised learning algorithm such as Logistic Regression, or SVM to build a spam classifier $f(\mathbf{x}) : \mathbb{R}^m \rightarrow \{+1, -1\}$. Here, extracting appropriate features $\mathbf{x}$ that capture the characteristics of the spam can improve the performance of spam classifiers. Thus, we try to model the characteristics of the automatically generated spam to extract the key features.

3.1 Topic Model with LDA

LDA is a method to model the content and topics of a collection of documents. We have vocabulary V that consists of terms, a set T of K topics in N documents. For every topic z, a distribution ϕ_z on V is sampled from $Dir(\beta)$, where $\beta \in \mathbb{R}_+^V$ is a smoothing parameter. Similarly, for every document d, a distribution θ_d on T is sampled from $Dir(\alpha)$, where $\alpha \in \mathbb{R}_+^T$ is a smoothing parameter.

One method for inferencing in LDA is Gibbs-sampling [8], which iteratively samples topic assignment z for word w. Knowing z, we can estimate the topic distribution $\theta_{d,z}$ for document d as

$$\theta_{d,z} = \frac{n_d^z + \alpha}{n_d + K\alpha} \tag{1}$$

where n_d^z is the number of words assigned to topic z, and n_d is the total word number in document d.

3.2 Sentence-level Topic Assignment

Our algorithm determines the topic assignment for sentence s by using the topic assignment voting by word w in sentence s. It stores the sampling results for each iteration and finally adopts the topic that has the max count in the sentence. This is analogous to selecting the topic assignment for words in [16]. The difference is that we employ this heuristic to select the topic of the sentence unit, not the word topic.

We show an example of topic assignments for ham and spam in Table 1. Each number denotes the topic ID assigned with each sentence in the document. This example shows that the spam has more unnatural topic transitions than the ham. We use the sequence of topic IDs to extract features. The details of the feature is described in Figure 3.

We consider that the simple way to model the topic transition between adjacent sentences is to construct an n-gram

Table 1: An example of topic assignments.

ham: 1 1 1 2 3 3 1 3 3 3 3 3 3 3 3 1 2 2 2 1
spam: 3 4 4 4 5 6 5 8 2 5 6 7 9 2 5 4 6 4 2 4

model based on the assigned topic ID. We refer to it as the *sentence-level topic n-gram model*. For bi-gram models, we can calculate the document probability by

$$P(z_{s_1}, \ldots, z_{s_{|s|}}) = \prod_{i=1}^{|s|} P(z_i|z_{i-1}).$$

The maximum likelihood estimation for $P(z_i|z_{i-1})$ is

$$\frac{c(z_{i-1}z_i) + \delta}{c(z_{i-1}) + K\delta},$$

where $c(z_{i-1}z_i)$ is the bi-gram count for $z_{i-1}z_i$ and $c(z_{i-1})$ is the uni-gram count for z_{i-1}. δ is a smoothing parameter for additive smoothing [12]. We can calculate the generative probability for unseen documents if topic IDs are assigned to their sentences. In this paper, we use the entropy value per sentence in the same way as the entropy value per word [12] as a measurement of natural topic transitions.

3.3 Sentence-level Topic Vector

We introduce another way to characterize the topic information of sentences in the document. That is, the sentence-level topic assignment is a discrete method to characterize sentences. We create a sentence-level topic vector in a similarly way to calculating the topic distribution for document d.

To obtain the topic vector for sentence s, we use the mode of topic assignment described in [17]. That is, we use only the mode topic ID for word w for topic assignment instead of all topic assignments. Since $P(\mathbf{z}|s) = \theta_{s,z}$ has K-dimensional values, we use $\theta_{s,z}$ as a topic vector for sentence s. This is similar to the calculation of $\theta_{d,z}$ for document d in Eq. 2. Our method calculates a topic transition between adjacent sentences by

$$cosine\_similarity(\theta_{s_i,\mathbf{z}}, \theta_{s_{i+1},\mathbf{z}}) = \frac{\theta_{s_i,\mathbf{z}} \cdot \theta_{s_{i+1},\mathbf{z}}}{\|\theta_{s_i,\mathbf{z}}\| \|\theta_{s_{i+1},\mathbf{z}}\|}.$$

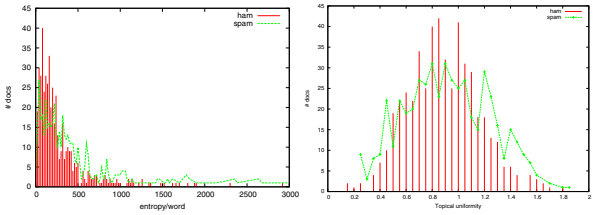
This idea is also used in [16] for the text segmentation task. To the best to our knowledge, we are first to apply similarity based on this sentence-level topic vector to the spam detection task.

4. EXPERIMENTS

We conducted preliminary experiments and spam detection experiments on a real spam blog dataset to verify our proposed method. We used C++ to implement LDA with a Collapsed Gibbs-sampler.

4.1 Datasets

We collected Japanese blog entries and set a spam or ham label to each blog page to build a spam dataset. We crawled Japanese blog pages from October 2010 to September 2011 to prepare the corpus. We then sampled blog pages and set labels. Each page was annotated by a single assessor. The spam annotation guideline was that the blog article contains information that seems to be written by the blog author (ham) or not (spam). The assessor set an additional tag (duplicate, affiliate, other) to the spam page. We selected pages with the duplicate tag and the affiliate tag as instances of automatically generated spam. As a result, we collected 1,750 spam labels and 2,000 ham labels. We note that our dataset has no more than one page from each blog site.



(a) (b)

Figure 1: (a) Entropy per word for ham and spam, (b) Topical uniformity of ham and spam.

4.2 Preliminary Experiment

As a preliminary experiment, we confirmed that language-model based features have difficulty in detecting automatically generated spam. We employed the bi-gram model as a language model with additive smoothing [12] with parameter $\delta = 0.001$. We used the Japanese blog dataset whose documents had previously been labeled as spam or ham. We used JTAG [7] as a Japanese tokenizer.

We used 1,000 ham examples to construct a language model and then calculated the average entropy per word [12] for another 1,000 ham documents and 1,000 spam documents. We used the entropy per word to evaluate the language model. Low entropy indicates that the document is likely to have been generate from the language model and high entropy indicates the document is unlikely to have been generated from the model. Thus, low entropy for ham examples and high entropy for spam examples is one desired result.

We show the result in Figure 1 (a). x axis plots the entropy value per word, and y axis plots the number of documents. The bin width of the histogram is 20. The bar plot is the count of ham documents and line graph is the count of spam documents. The graph shows that spam documents have slight higher entropy values. This result indicates that language-model based features have difficultly in detecting this kind of spam.

We also confirmed whether the topical uniformity score introduced in [15] could capture spam characteristics or not. We analyze how well the topic uniformity score works for the dataset. The histograms of the topic uniformity score of ham and spam are shown in Figure 1 (b). It indicates that the topic uniformity score has difficultly in distinguishing spam from ham.

We conducted another preliminary experiment to verify our assumption as appropriate for modeling automatically generated spam. We prepared LDA with 900 ham documents, and then applied this model to 100 ham and 100 spam documents, all previously unseen. We estimated $P(\mathbf{z}|s)$ for each sentence in the document to calculate cosine similarity between adjacent sentences. Figure 2 shows typical examples of the topic difference over sentences in ham and spam documents. We note that the y axis in the figure indicates (1 - cosine similarity) so a higher value means that a larger topic shift has occurred.

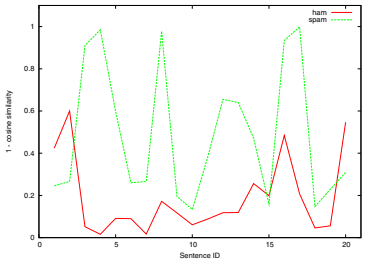


Figure 2: 1 - cosine similarity scores of adjacent sentences based on topic distribution vectors for typical ham and spam examples.

- Based on sentence-level topic assignment:
  - Entropy per sentence based on the topic bi-gram model.
  - Maximum consecutive number of the same topic ID.
  - Number of unique topic IDs.
  - Number of topic ID change over sentences.
- Based on sentence-level topic vector:
  - Mean, variance, maximum and minimum values of (1 - cosine similarity) in the list of adjacent sentences.
  - Number of (1 - cosine similarity) value that exceeds threshold $h \in \{0.1, 0.3, 0.5\}$.

Figure 3: Details of our proposed features.

4.3 Spam Detection Experiments

We conducted spam detection evaluation using the dataset. We used `LIBLINEAR`[2] as a typical SVM implementation [5]. We split the dataset into five parts for cross-validation (three training sets, one validation set, and one test set). Trade-off parameter C for SVM was chosen from {0.0001, 0.001, 0.01, 0.1, 1.0, 10.0} so as to achieve the highest accuracy against the validation set. We also prepared a LDA model to extract baseline features and proposed features. We used the training set to estimate a LDA model in each fold of cross-validation. The LDA parameters were 2,000 iterations with 1,000 burn-in phase, $\alpha = 0.5$, $\beta = 0.01$, $K = 20$.

We used the TF-IDF-based term weighting scheme as in [2] and combined them with the conventional topic diversity features proposed in [15] as a baseline (`Pavlov et al.`). As a proposed method, we also combine the TF-IDF features with our proposed features listed in Figure 3 (`Our method`). The F1-measure for spam class and ham class and AUC were used to evaluate the method in the same way as in previous works [2][15].

We show the results in Table 2. They confirm that our proposed method outperforms the conventional method with respect to F1(spam), F1(ham) and AUC values. This indicates that our method extracts the highly useful information that cannot be modeled by the features proposed in [15].

5. CONCLUSION

We proposed a feature extraction method that is based on sentence-level topic transitions in the documents. We introduced a topic-voting heuristic with conventional LDA to achieve sentence-level topic assignment. We also proposed a sentence-level topic n-gram model based on sentence-level topic assignment. Another contribution is the use of a sentence-level topic vector to extract novel features. We conducted preliminary experiments and confirmed that our method can well capture the automatically generated spam characteristics that invisible to conventional methods. We also verified that our method can yield spam classifiers with improved performance in the supervised machine learning framework. Since sentence-level topic assignment technique is not limited to spam detection task, we also plan to apply this technique to other tasks as a future work.

[2] `http://www.csie.ntu.edu.tw/~cjlin/liblinear/`

Table 2: Experimental results.

Method	F1 (spam)	F1 (ham)	AUC
`Pavlov et al.`	.702	.798	.895
`Our method`	**.757**	**.815**	**.897**

6. REFERENCES

[1] I. Bíró, D. Siklósi, J. Szabó, and A. A. Benczúr. Linked latent dirichlet allocation in web spam filtering. In *Proc. AIRWeb '09*, AIRWeb '09, pages 37–40, 2009.

[2] I. Bíró, J. Szabó, and A. A. Benczúr. Latent dirichlet allocation in web spam filtering. In *Proc. AIRWeb '08*, AIRWeb '08, pages 29–32, 2008.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[4] M. Erdélyi, A. Garzó, and A. A. Benczúr. Web spam classification: a few features worth more. In *Proc. WebQuality '11*, WebQuality '11, pages 27–34, 2011.

[5] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *J. Mach. Learn. Res.*, 9:1871–1874, 2008.

[6] D. Fetterly, M. Manasse, and M. Najork. Detecting phrase-level duplication on the world wide web. In *Proc. SIGIR '05*, SIGIR '05, pages 170–177, 2005.

[7] T. Fuchi and S. Takagi. Japanese morphological analyzer using word co-occurrence: Jtag. In *Proc. COLING '98*, pages 409–413, 1998.

[8] T. L. Griffiths and M. Steyvers. Finding scientific topics. In *Proceedings of the National Academy of Sciences*, volume 101 (suppl. 1), pages 5228–5235, 2004.

[9] Z. Gyöngyi and H. Garcia-Molina. Web spam taxonomy. In *Proc. AIRWeb '05*, pages 39–47, 2005.

[10] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with trustrank. In *Proc. VLDB '04*, pages 576–587, 2004.

[11] Y. Jo and A. H. Oh. Aspect and sentiment unification model for online review analysis. In *Proc. WSDM '11*, WSDM '11, pages 815–824, 2011.

[12] C. D. Manning and H. Schütze. *Foundations of statistical natural language processing*. MIT Press, Cambridge, MA, USA, 1999.

[13] J. Martinez-Romo and L. Araujo. Web spam identification through language model analysis. In *Proc. AIRWeb '09*, AIRWeb '09, pages 21–28, 2009.

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

[15] A. Pavlov and B. V. Dobrov. Detecting content spam on the web through text diversity analysis. In *Proc. SYRCoDIS '11*, pages 11–18, 2011.

[16] M. Riedl and C. Biemann. Sweeping through the topic space: bad luck? roll again! In *Proc. ROBUS-UNSUP '12*, ROBUS-UNSUP '12, pages 19–27, 2012.

[17] M. Riedl and C. Biemann. Topictiling: a text segmentation algorithm based on lda. In *Proc. ACL '12 Student Research Workshop*, ACL '12, pages 37–42, 2012.

[18] N. Spirin and J. Han. Survey on web spam detection: principles and algorithms. *SIGKDD Explor. Newsl.*, 13(2):50–64, 2012.

[19] E. Vallés and P. Rosso. Detection of near-duplicate user generated contents: the sms spam collection. In *Proc. SMUC '11*, SMUC '11, pages 27–34, 2011.