7-2008

# On Profiling Blogs with Representative Entries

Jinfeng ZHUANG
*Nanyang Technological University, Singapore*

Steven C. H. HOI
*Singapore Management University*, chhoi@smu.edu.sg

Aixin SUN
*Nanyang Technological University, Singapore*

# On Profiling Blogs with Representative Entries

Jinfeng Zhuang, Steven C.H. Hoi, and Aixin Sun
School of Computer Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798
{zhua0016,chhoi,axsun}@ntu.edu.sg

## ABSTRACT

With an explosive growth of blogs, information seeking in blogosphere becomes more and more challenging. One example task is to find the most relevant topical blogs against a given query or an existing blog. Such a task requires concise representation of blogs for effective and efficient searching and matching. In this paper, we investigate a new problem of profiling a blog by choosing a set of $m$ most representative entries from the blog, where $m$ is a predefined number that is application-dependent. With the set of selected representative entries, applications on blogs avoid handling hundreds or even thousands of entries (or posts) associated with each blog, which are updated frequently and often noisy in nature. To guide the process of selecting the most representative entries, we propose three principles, i.e., *anomaly*, *representativeness*, and *diversity*. Based on these principles, a greedy yet very efficient entry selection algorithm is proposed. To evaluate the entry selection algorithms, an extrinsic evaluation methodology from document summarization research is adapted. Specifically, we evaluate the proposed entry selection algorithms by examining their blog classification accuracies. By evaluating on a number of different classification methods, our empirical results showed that comparable classification accuracy could be achieved by using fewer than 20 representative entries for each blog compared to that of engaging all entries.

## Categories and Subject Descriptors

H.3.1 [**Content Analysis and Indexing**]: Abstracting methods; H.3.3 [**Information Search and Retrieval**]: Selection process, Information filtering

## General Terms

Experimentation

## Keywords

Blog profiling, Entry selection, Blog classification


*AND'08*, July 24, 2008, Singapore


## 1. INTRODUCTION

Blogs, or Weblogs, are online diaries created and maintained by individuals or organizations. In the era of Web 2.0, blog has become an important channel for people to express themselves, share information, and communicate among each other. According to Technorati[1], a popular blog search engine, the number of blogs doubles about every six months. Currently various applications on blogs have been developed for capturing this emerging and rapidly growing search and advertisement market. For instance, most web search engines have provided blog search services. The boom of blogs has attracted a surge of research attention in text mining, information retrieval, social studies, and other areas.

In general, a blog contains a number of elements, including entries (or posts), tags, comments, links, and others. Among these elements, entries are the most informative and important element for content analysis of a blog. Although other elements could also be beneficial, to simplify the problem, in our study, each blog is treated as a set of entries or a sequence of entries ordered by their publication time if the time order is necessary.

### 1.1 Motivation

Before the popularity of blogs, most applications developed for facilitating information seeking tasks on the web (e.g., web search) work on the granularity of web pages, not on the sets of pages. As a result, it is often not very effective for the existing applications to handle queries at the blog level, where each blog consists of a set of entries each of them is treated as a web page. For example, by existing search engines, it is not straightforward to find the blogs whose topic is "photography". A photography blog refers to a blog whose entries are mostly about techniques or information on photography; yet the blog may contain a few entries on other topics as blogs are noisy and informal by their nature. In fact, some websites are trying to list topical blogs in blog directories either manually or collaboratively, such as BlogFlux[2], BlogCatalog[3], and BOTW[4]. These applications call for effective and efficient technique for analyzing the contents of blogs. To enhance the effectiveness, in this paper, we investigate a new research problem of profiling a blog by choosing a set of most representative entries. Given a blog with a number of entries, **blog profiling** is a task of selecting a set of $m$ entries that best represent the main

[1]http://www.technorati.com, accessed on May 23, 2008.
[2]http://dir.blogflux.com, accessed on May 23, 2008.
[3]http://www.blogcatalog.com, accessed on May 23, 2008.
[4]http://blogs.botw.org, accessed on May 23, 2008.

topic of the blog. The number of representative entries $m$ is a predefined parameter and is application-dependent.

Blog profiling is an important research problem, which is beneficial for several applications including the follows:

- *Data cleaning* is a critical step in most data mining tasks. Due to the informal writing style, a blog may contain duplicate, noisy entries. For example, an entry may be simply a single URL that has little semantic. More than just removing such entries, blog profiling would single out the most informative entries for facilitating subsequent data mining tasks (e.g., blog classification). Given that the set of representative entries is more concise and less noisy, blog profiling enhances the efficiency and effectiveness of blog data mining tasks.
- *Blog summarization* is a straightforward application from blog profiling. In fact, the selected most representative entries serve as a summary of the entire blog. With them, one can avoid reading a large number of entries (if not all) before deciding if a blog is of his/her interest. This is extremely important when being required to make a decision within a short time or to process a large number of blogs. In addition, the summarization with blog profiling also helps to improve the blog presentation. For example, when a blog search engine returns a list of blogs for a query, it would be better to display those representative blog entries that reflect the blogger's main interest.
- *Blog classification*(BC) refers to the task of automatically assigning class labels to a blog based the main topics of its entries. By working on only those representative blog entries, we can improve the efficiency of the classification task. In this paper, we will employ blog classification to assess the performance of our proposed entry selection techniques.

## 1.2 Challenges and Contribution

The key to blog profiling is to efficiently sample a set of most representative entries from a collection of blog entries in a blog. This is challenging due to several reasons:

First, it is difficult to determine if a small set of selected entries contains the most valuable information of a blog. According to *no free lunch* theorem, one entry can by no means be ascertained better than another without a priori quality measure $f$. The challenge lies in how to identify a good measure $f$ such that the selected entries can represent the blog in a large variety of circumstances.

Second, the selection process must be efficient. Specifically, the time complexity of the selection algorithm should be no worse than that of the subsequent data mining algorithms. This constraint is reasonable since one of the objectives is to improve data mining efficiency. Unfortunately, for a blog $B_i$ of size $N_i$, the total number of possible combinations for forming a set of $m$ entries $S_i$ is $C_{N_i}^m$, which is huge for a large scale application. Hence, it is prohibitive for applying a naive exhaustive enumeration for the selection.

Third, there is no universal evaluation metric. It is infeasible to request bloggers themselves to indicate whether a selected entry is representative or not. Conducting a large scale user study is also challenging as one may have to read all entries in a blog before telling which are the representative ones given that one blog may contain thousands of entries.

To the best of our knowledge, our work is the first comprehensive study on *representative entry selection* towards blog data mining tasks. Our contributions in this paper are summarized below:

- We formally define the representative entry selection problem for profiling blogs. We further formulate the problem into an optimization framework. The framework is generic and can be instantiated with a variety of representativeness measure functions under different principles.
- We propose two greedy search algorithms that can efficiently solve the resulting optimization task. The proposed algorithms are simple and effective although they do not guarantee global optima.
- To evaluate the performance of entry selection methods, we introduce the *blog classification* task and conduct a series of experiments to evaluate selection results by examining the classification accuracies.

The rest of the paper is organized as follows. We review the related work in Section 2 and propose the techniques for profiling blogs in Section 3. We formulate the selection problem into an optimization framework and propose the greedy algorithm for solving it in Section 4. In Section 5, we discuss the possible methods of evaluating the selected entries. The experiments and results are presented in Section 6, followed by the conclusion in Section 7.

# 2. RELATED WORK

Our work is related to several research topics, including blog analysis, sampling (or instance selection), and multi-instance learning. We briefly review the related work on each topic.

Blog has attracted much research attention in recent years. Kumar *et al.* studied blog communities' evolution as early as 2003 [16]. On the content analysis of blog data, Durant *et al.* evaluated the performance of Naive Bayes and Support Vector Machines (SVM) classifiers on mining the sentiment from political blog entries [15]. Mishne investigated the mood classification problem with blogs [10]. Kolari studied the blog identification problem for detecting spam blogs [20]. Ni *et al.* investigated a blog classification problem for classifying two classes of blog entries: informative and affective [19]. Very recently, single blog entry summarization algorithms incorporating comments information were proposed [13]. Our work suggests to represent a blog more compactly with informative entries. It can serve as data preprocess for most of the above work.

Our entry selection problem can be viewed as instance selection (or relevant example selection) problem, which has been actively studied in the machine learning community. Some overviews of general issues and solutions can be found in [17, 21]. The work on instance selection can be briefly grouped into *filter* and *embedded* approaches. With the filter approach, the sampling procedure is executed as data preprocess such that the resultant data can be fed into any following data mining task. The most straightforward method is random sampling where each instance has equal chance to be drawn. In stratified sampling, the whole data is separated into a number of disjoint subsets; samples are drawn from each subset independently. With the embedded approach,

the selection process is implicitly embedded in a data mining algorithm (e.g., a classifier). Bagging [3] and boosting [22] fall into this category. Our work aligns in the filter approach. We profile a blog by sampling a subset of its entries before other data mining task is conducted.

Moreover, our work is also close to *feature selection*, which is to select a subset of features for data mining tasks. Some survey work on feature selection can be found in [14, 6]. Although blog entry selection is similar to feature selection, there are certain differences between them. One is that in the subset of features selected by feature selection is used for all examples in the dataset, but the subset of entries selected by entry selection is independent for each blog. For classification, feature selection is done only during the training stage, while entry selection should be applied during both the training and test stages. Hence, a lot of methods for feature selection cannot be directly applied to blog entry selection.

Finally, our work is related to multi-instance learning (MIL) as classification is employed in this study for performance evaluation. The task of MIL is to predict labels for ambiguous examples: each example has several instances (or feature vectors) describing it, some of which may be responsible for the observed class label of the example. In training data, class label is assigned to the example instead of its instances. A number of MIL algorithms have been proposed, such as Axis-parallel Rectangle (APR) [7], algorithms based on diverse density [18], Citation-kNN [25], and SVM variants [1, 8, 4]. In this paper, we evaluate the performance of entry selection algorithms by examining the classification accuracy on the blogs that are represented by the selected entries. The blog classification task could be formulated as an MIL problem in the sense that each blog consists of a set of entries.

# 3. ENTRY SELECTION PRINCIPLES

We first formally define the entry selection problem, then discuss three principles for solving the problem, i.e., *anomaly*, *representativeness*, and *diversity*.

## 3.1 Entry Selection

Let $B_i$ denote a blog. We then represent blog $B_i$ by a set of $N_i$ entries, i.e., $B_i = \{B_{i1}, \ldots, B_{iN_i}\}$, where $B_{ij}$ is the $j$-th entry of $B_i$, and $N_i$ is the total number of entries in $B_i$, i.e., $N_i = |B_i|$. The problem of entry selection is defined as:

DEFINITION 1 (ENTRY SELECTION). *Given a blog $B_i$ and a predefined number of entries to be selected $m$, the entry selection problem is to select a subset of entries $S_i \subseteq B_i$, where $|S_i| = \min\{m, |B_i|\}$, such that the selected entries $S_i$ best represent the blog $B_i$.*

Intuitively, a simple way to solve the entry selection task is to randomly sample a subset of $m$ entries. We refer to this *random* entry selection method as a baseline method in our study. Apparently, such an approach is insufficient for solving the problem effectively, particularly when $m$ is small. That is, the selected entries might be less representative with respect to all entries in the blog.

As aforementioned, there are two major challenges for solving the problem effectively. First, it requires quality measure principles for guiding the entry selection task. Second, it needs effective algorithms for sampling the entries efficiently for achieving the targeting principles. Next we present three principles for guiding the entry selection task.

## 3.2 Principles for Entry Selection

Consider a blog $B_i$ and a set of selected entries $S_i$, where $S_i = \emptyset$ at the beginning. The key for the entry selection task is to formulate a quality evaluation function $f$ for measuring the quality $f(B_{ij}; B_i, S_i)$ for an entry $B_{ij} \in B_i$. We propose three principles for defining the quality evaluation function and guiding the entry selection task:

- *Anomaly.* Blogs often contain noisy entries not related to its main topics. Such noisy entries can deteriorate the performance of the subsequent data mining algorithms. It is therefore important to avoid the **noisy** entries in an entry selection task.
- *Representativeness.* To gain informative entries, one key is to select the entries that are most **representative** to the main theme of the blog. For example, we may want to choose the entry that is closest to the centroid of the blog.
- *Diversity.* The last principle is to choose the **diverse** entries such that the overall information of the selected entries can be maximized. This is important to avoid selecting the *redundant* entries.

Note that the first two principles, *anomaly* and *representativeness*, may not be orthogonal. For example, a noisy entry is usually considered as a less representative entry. For simplicity, we can combine them together and focus on measuring the representativeness. On the other hand, it is reasonable to assume that *representativeness* and *diversity* is orthogonal. As a result, for a subset of entries $S_i$ and a candidate entry $B_{ij} \in B_i \backslash S_i$, we can define the quality evaluation function as follows:

$$f(B_{ij}; B_i, S_i) = r(B_{ij}; B_i) + \lambda d(B_{ij}; S_i) \quad (1)$$

where the function $r(B_{ij}; B_i)$ measures the representativeness of a candidate entry $B_{ij}$ with respect to the set of entries in $B_i$, the function $d(B_{ij}; S_i)$ measures the diversity by comparing the candidate entry $B_{ij}$ with the selected entries in $S_i$, and $\lambda$ is a parameter to balance the tradeoff.

*Remark.* To avoid choosing the noisy entries, we can apply some outlier detection techniques to remove the noisy entries. We will discuss this in the formulation of representativeness functions below.

## 3.3 Representativeness Measure

We propose two methods for the representativeness measure as follows. One is without the explicit outlier detection, and the other includes outlier detection.

### 3.3.1 Centroid Based Measure.

For a given $B_i = \{B_{ij} | j = 1, \ldots, N_i\}$, we calculate its centroid of its entries as: $centroid_i = \frac{1}{m_i} \sum_{j=1}^{m_i} B_{ij}$. Then we can define the representativeness measure of $B_{ij}$ below:

$$r(B_{ij}; B_i) = sim(B_{ij}, centroid_i) \quad (2)$$

where $sim(\cdot, \cdot)$ is some similarity function, such as cosine similarity between two feature vectors.

The underlying intuition is that the more closer $B_{ij}$ to $centroid_i$, the more related to the theme of $B_i$. It would be very effective if most of the entries follows the same theme or event semantically.

### 3.3.2 Cluster Based Measure

Clustering is often used to reveal the inherent data structure. In this approach, we employ clustering techniques to remove the noisy entries and define the representativeness measure on the major clusters. Specifically, we first cluster the entries in the blog into $k$ clusters and then treat the "small" clusters as outliers and drop them before the entry selection phase. Let $C_1, C_2, \ldots, C_k$ denote the obtained clusters, where $|C_1| \geq |C_2| \geq ... \geq |C_k|$, the outliers are defined as:

$$\{C_j | \kappa^* < j \leq k\} \text{ where } \kappa^* = \arg\min_{\kappa} \sum_{j=1}^{\kappa} \frac{|C_j|}{N_i} \geq 1 - \alpha$$

where $\alpha \in [0, 1]$ is the fraction of outliers. Entries falling into the small clusters will be dropped before the selection process. In the next phase, for an entry $B_{ij} \in B_i$, we measure its representativeness below:

$$r(B_{ij}; B_i) = sim(B_{ij}, c_j^*) \tag{3}$$

where $c_j^*$ is the cluster center nearest to the entry $B_{ij}$. Further, we notice that the length of an entry would also influence the representativeness. Typically, the longer an entry, the more information it conveys. Hence, it is reasonable to keep those longer entries.We modify the representativeness measure as follows:

$$r(B_{ij}; B_i) = sim(B_{ij}, c_j^*) \times \frac{l(B_{ij})}{max_j\ l(B_{ij})} \tag{4}$$

where $l(B_{ij})$ is the number of distinct words in the entry $B_{ij}$.

The motivation for the cluster based representativeness measure lies in that a blog often contains multiple sub-topics. It is often insufficient to employ the *centroid* only. We hope to capture the major sub-topics by keeping the centers of major clusters, and hence could profile the blog $B_i$ more accurately and completely.

## 3.4 Diversity Measure

In general, the diversity between two entries $B_{ij}$ and $B_{ik}$ can be defined by using their similarity value:

$$d_{jk} = d(B_{ij}, B_{ik}) = 1 - sim(B_{ij}, B_{ik}), \tag{5}$$

where $sim(\cdot, \cdot)$ is a similarity value in $[0, 1]$.

Further, to measure the diversity of a given entry $B_{ij}$ with respect to a set of entries $S_i$, we propose two approaches. One is the *mean* based diversity measure defined below:

$$d(B_{ij}, S_i) = \frac{1}{|S_i|} \sum_{B_{ik} \in S_i} d(B_{ij}, B_{ik}). \tag{6}$$

# 4. FORMULATING ENTRY SELECTION AS AN OPTIMIZATION TASK

The above principles show that the goal of an entry selection task is to choose the subset of entries that are most *representative* and least *redundant*. One subset $S$ may have more representative entries but less diverse than $S'$. It is often difficult to solve such multi-criterion problem. Alternatively, we combine them into one objective function by introducing non-negative weight. Based on this idea, we formulate the entry selection problem into a formal optimization task:

$$\max_{S_i \subseteq B_i} \quad \sum_{B_{ij} \in S_i} f(B_{ij}; B_i, S_i \setminus B_{ij}) \tag{7}$$

$$s.t. \quad |S_i| = m\ . \tag{8}$$

By introducing a variable $\mathbf{z} \in \mathbb{R}^{|B_i|}$ and using the quality evaluation function $f$ in (1), we can rewrite the above optimization into the following:

$$\max_{\mathbf{z}} \quad \sum_{j=1}^{|B_i|} z_j r(B_{ij}; B_i) + \frac{\lambda}{2} \sum_{j=1}^{|B_i|} z_j d(B_{ij}; \cup_{z_k=1} \{B_{ik}\}) \tag{9}$$

$$s.t. \quad z_j \in \{0, 1\}, j = 1, \ldots, n$$

$$\mathbf{1}^\top \mathbf{z} = m\ .$$

where $z_j = 1$ (or 0) indicates that the entry $B_{ij}$ is selected (or excluded). Further, using the *mean* based diversity measure, we can further write the above optimization into:

$$\max_{\mathbf{z}} \quad \sum_{j=1}^{|B_i|} z_j r_j + \frac{\lambda}{2m} \sum_{j=1}^{|B_i|} \sum_{k=1}^{|B_i|} z_j z_k d_{jk} \tag{10}$$

$$s.t. \quad z_i \in \{0, 1\}, i = 1, \ldots, n$$

$$\mathbf{1}^\top \mathbf{z} = m\ .$$

where $\mathbf{r} \in \mathbb{R}^{|B_i|}$ with element $r_j = r(B_{ij}; B_i)$, and $D \in \mathbb{R}^{|B_i| \times |B_i|}$ with element $d_{jk} = d(B_{ij}, B_{ik})$ and all diagonal elements $d_{jj} = 0$.

Unfortunately, the above optimization belongs to a typical 0-1 integer programming (IP) problem, which is known as NP-hard [2]. To find approximate algorithms for solving the problem efficiently, there are two general ways to be considered. One is to approximate the nonconvex problem into a convex optimization by some relaxation [2]. For example, we can replace the constraint $z_i \in \{0, 1\}$ with the convex constraint $z_i \in [0, 1]$. As a result, we can approximate it into the following problem:

$$\max_{\mathbf{z}} \quad \mathbf{z}^\top \mathbf{r} + \frac{\lambda}{2} \mathbf{z}^\top D \mathbf{z} \tag{11}$$

$$s.t. \quad \mathbf{1}^\top \mathbf{z} = m,$$

$$0 \preceq \mathbf{z} \preceq 1\ .$$

which is a standard quadratic program (QP) that can be solved with global optima by some existing convex optimization techniques. The time complexity of such QP solutions, however, is often of $\mathcal{O}(n^3)$, which is inefficient and not scalable for real large-scale Web applications.

Another general way to find approximate solutions efficiently is to investigate some greedy search algorithms. In this paper, we adopt the second way in sake of its practical efficiency. The main idea of the proposed greedy algorithm is to iteratively select an entry with the best quality value, and then update other entries' quality values once the entry is selected. The similar ideas for greedy search approaches have also been used for solving some integer programming problems in previous research [9, 27, 12, 11, 5]. Algorithm 1 shows the details of the proposed algorithm. From the pseudo code, we can see that, if ignoring the computation of diversity and representative measure, the complexity of this algorithm for selecting $m$ entries from a set of $n$ entries is $\mathcal{O}(n \times m)$. Thus, it is a linear algorithm that can be done very efficiently for large-scale applications. We

**Algorithm 1** A Greedy Entry Selection Algorithm (**GES**)

**input**: $B_i$, $m$, $\lambda$
**output**:$S_i$
1: **set** $S_i^0 = \emptyset$;
2: **for** each $B_{ij} \in B_i$,
3: $f_{ij} = f(B_{ij}; B_i, S_i^0) = r(B_{ij}; B_i)$;
4: Compute the diversity matrix $D \in \mathbb{R}^{|B_i| \times |B_i|}$;
5: **for** t = 1,...,m
6: (1) select $B_{ij}^* = \arg\max_{B_{ij} \in B_i \backslash S_i^{t-1}} f(B_{ij}; B_i, S_i^{t-1})$
7: update $S_i^t = S_i^{t-1} \cup \{B_{ij}^*\}$, $B_i = B_i - \{B_{ij}^*\}$
8: (2) for each $B_{ik} \in B_i \backslash S_i$, update $f_{ik} = f_{ik} - \frac{\lambda}{m}(1 - d_{jk})$
9: $S_i = S_i^m$.

refer to this greedy entry selection algorithm as "**GES**" for short.

The proposed GES algorithms is simple and efficient for finding an effective approximate solution to the optimization in (1). Although it does not guarantee the global optima for the optimization, we found that the greedy algorithm is rather effective in achieving good empirical results from our experimental studies.

# 5. BLOG CLASSIFICATION FOR PERFORMANCE EVALUATION

## 5.1 Motivation and Problem Definition

It is not a trivial task to evaluate the entry selection algorithm. Based on the proposed principles, we give a formulation as objective function for measuring the quality of $S_i$. However, one may argue that optimizing (1) does not mean $S_i$ can represent $B_i$ well. Considering our goal is to profile a blog, the most reliable way is to catch up with the blogger of $B_i$ for assessment of $S_i$. Let the blogger decide whether $S_i$ ensembles the most important entries in his or her mind. Unfortunately, this is not practical. Another method is to employ some people to manually label representative entries. However, this method may suffer from labelers' subjective criterion and is time-consuming for large problems.

As aforementioned, a natural evaluation strategy for entry selection algorithms is to adopt the resulting entry subset in a subsequent data mining task and examine the performance. One important task in blog data mining is to automatically detect the common topics of a blog by analyzing its discussed content. Such a task can be formalized as a "blog classification" (BC) problem. Specifically, to evaluate the effectiveness of an entry selection algorithm, we can compare the difference of classification accuracies achieved by two different blog classification approaches: one uses the set of all original entries $B_i$ and the other adopts the set of selected entries $S_i$. If the one with $S_i$ can produce the comparable accuracy, it is reasonable to claim that $S_i$ represents $B_i$ well.

More formally, let $\mathcal{B}$ denote a blog space and $\mathcal{C} = \{c_1, \ldots, c_t\}$ denote a set of $t$ predefined categories. Given a training set with $N$ blogs $\{(B_i, Y_i), i = 1, \ldots, N\}$, where $Y_i \subseteq \mathcal{C}$ is the label set of the blog $B_i$, the problem of *blog classification* (BC) is to learn a classification model, $f : \mathcal{B} \mapsto \mathcal{C}$, for predicting the label set $Y_j$ of an unseen test blog $B_j \in \mathcal{B}$ accurately.

The proposed blog classification is crucial to many Web blog search and browsing applications. If all blogs in WWW can be automatically classified, one can provide some blog directory service similar to Yahoo! Directory for facilitating users' browsing. In fact, some websites have provided such services. For example, *BlogFlux* classifies blogs into 161 flat topical categories; *BlogCatalog* organizes blogs into hierarchical topical categories with 49 top-level categories; and *BOTW* lists blogs in a hierarchy with 12 top-level categories. To our knowledge, the class labels of these blogs are often assigned manually, which is very expensive and cannot be updated efficiently. Therefore, an automatic blog classifier is quite necessary. Considering the importance of BC, we use its accuracy to measure whether our entry selection result can represent the original blog well.

## 5.2 Blog Classification Methods

First we consider to transform BC into a single instance learning (SIL) problem. We calculate the entry centroid $C_i$ to represent $B_i$ and pass $C_i$ to a SVM learner (referred as SIL-Cen).

Another classification methodology is to develop multi-instance kernels and deploy them into SVMs [24]. First we consider linear normalized set kernel (NSK) [8]:

$$k_{nsk}(B_i, B_j) = \frac{k_{set}(B_i, B_j)}{\sqrt{k_{set}(B_i, B_i)}\sqrt{k_{set}(B_j, B_j)}}$$

where $k_{set}(B_i, B_j) = \sum_m \sum_n k_{inst}(B_{im}, B_{in})$ and $k_{inst}$ is a kernel defined on entries. We simply use cosine similarity.

The other method is to define kernels based on some set distance measure $d$ [26]:

$$k_{RBF}(B_i, B_j) = e^{-\gamma d(B_i, B_j)^2}$$

where $\gamma \in \mathbb{R}^+$. We can adapt some existing set distance measures directly. For example, the *Hausdorff distance*, one of the most well-known set distance measure, is defined as:

$$H(B_i, B_j) = \max\{h(B_i, B_j), h(B_j, B_i)\}$$

where $h(B_i, B_j) = \max_m \min_n d^*(B_{im} - B_{jn})$ and $d^*(B_{im} - B_{jn}) = 1 - \cos(B_{im}, B_{jn})$ measures the distance between two entries. Similarly, we can also measure the distance between two sets by computing their average distance, minimal distance, or maximal distance. We denote this generalized RBF kernel by MIL-$K_{RBF}^{HAU}$, MIL-$K_{RBF}^{AVG}$,MIL-$K_{RBF}^{MIN}$, and MIL-$K_{RBF}^{MAX}$, respectively.

Finally, we also evaluate some conventional multi-instance learning technique for blog classification. Although there are many existing multi-instance learning algorithms [7], most of them cannot be applicable to blog classification directly. In this work, we only evaluate the *citation-kNN* [25] algorithm ("cit-KNN").

# 6. EXPERIMENTS

## 6.1 Dataset

We have crawled a blog dataset from BlogFlux [5]. In this experiment, we form a datset with 5,000 blogs containing 840,150 entries written in English. These blogs belong to 10 popular categories and each blog belongs to one or more categories. For experimental evaluation, we partition the dataset into two parts: half for training and half test. Tables 1 and 2 show the statistics of our data set. Figure 1 plot the entry number distribution. A small number of

[5]http://dir.blogflux.com

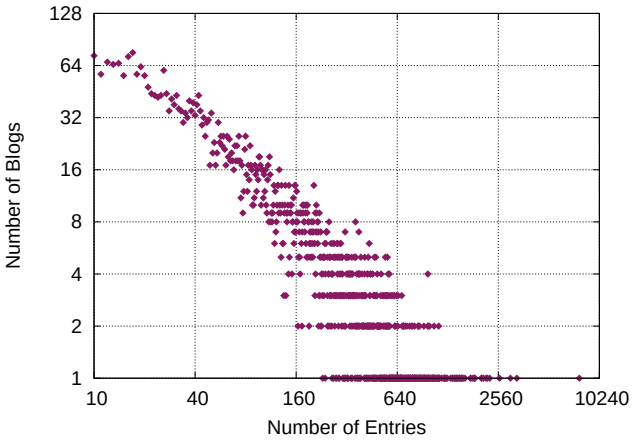



**Figure 1: The distribution of blog entry numbers in our testbed.**

blogs contain more than several thousands entries. For text preprocessing and feature extraction, we employ the *Lucene* toolkit[6] for tokenization, by which terms are stemmed and stop words are removed. The TF-IDF features are extracted to represent each entry.

**Table 1: The statistics of our experimental dataset**

| | Total | Training Set | Test Set |
|---|---|---|---|
| # blogs | 5,000 | 2,500 | 2,500 |
| # entries | 840,150 | 424,948 | 415,202 |

**Table 2: The numbers of blogs with 10 categories**

| | personal | business | politics | ent.† | health |
|---|---|---|---|---|---|
| train | 662 | 235 | 222 | 250 | 211 |
| test | 664 | 246 | 222 | 234 | 223 |
| | sports | art | humor | travel | religion |
| train | 239 | 203 | 230 | 159 | 170 |
| test | 216 | 193 | 247 | 154 | 170 |

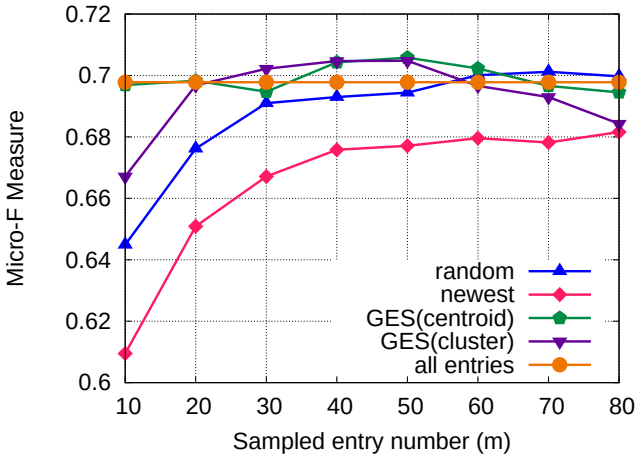
† *"ent." stands for "entertainment" for short.*

## 6.2 Experimental Setup

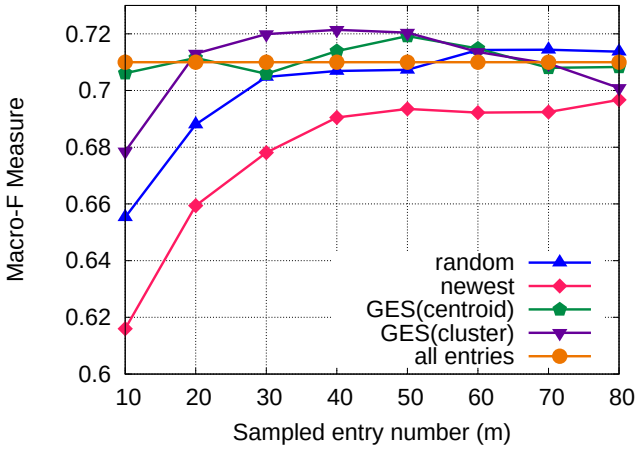
To examine the performance of the proposed entry selection technique, we conduct two set of experiments. First, we comprehensively compare the proposed entry selection technique with several baseline approaches for blog classification. Second, we evaluate the performance of several different classification methods together with the entry selection technique for blog classification. We give a brief analysis of the efficiency advantage of using the selected $S_i$ for blog classification tasks.

In our experiments, for learning SVM classifiers, we employ the popular LIBSVM package [7] for all experiments. The penalty parameter $C$ of SVM and the regularization parameter $\lambda$ of the proposed entry selection algorithm are all determined by cross-validation on the training set. The result of the *random* sampling method is averaged over 5 runs. For the cluster-based method, we adopt the *CLUTO* toolkit [8]. We run the clustering procedure for 10 times and choose the result that achieves the best intrinsic clustering objective function. Outliers will also be discarded by the clustering process.

For performance measure, we use the standard $F_1$ metric, which is widely adopted in text categorization task [23]. The F1 measure is defined as: $F1 = 2 \times P \times R/(P + R)$, where $P$ and $R$ are precision and recall, respectively. For comparison, we employ *Macro-F1* and *Micro-F1* measures over 10 categories to evaluate different entry selection algorithms. All of our experiments were conducted on a Windows PC with 3.4GHz CPU and 3GB RAM.




(a) Micro-F1 Measure




(b) Macro-F1 Measure

**Figure 2: Comparison of the proposed GES algorithm with other entry selection methods.**

[6]http://lucene.apache.org/java/docs/
[7]http://www.csie.ntu.edu.tw/ cjlin/libsvm
[8]http://glaros.dtc.umn.edu/gkhome/views/cluto

Table 3: Performance evaluation of different entry selection methods when the sampled entry number is 20

| | GES(cluster) | GES(centroid) | random | newest | all entries |
|---|---|---|---|---|---|
| Micro-F1 | .7001 | .6967 | .6762 | .6509 | .6978 |
| Macro-F1 | .7093 | .7151 | .6880 | .6594 | .7100 |

## 6.3 Comparison of Entry Selection Techniques

In this experiment, we calculate the centroid $C_i$ of the selected $S_i$ and pass $C_i$ to SVM classifier (SIL-Cen). The linear kernel between blogs' mean vector is essentially average cosine similarity between entries. So it's robust on noisy entries (*noisy* means the negative entries in a positive blog). Besides the robustness advantage, it has linear time complexity on the number of features. In this section we use it to compare different entry selection methods.

To examine the effectiveness of the proposed algorithm, we compare the proposed algorithm with other heuristic sampling approaches including: (1) a random sampling approach (*random*, the result is averaged over 5 times run), (2) a time-based sampling method with newest entries (*newest*), and (3) a reference method with all entries.

Figure 2 shows the experimental results. Several observations can be drawn from the results. First of all, we can see that the proposed algorithm is significantly better than the *random* and *newest* sampling approaches. Meanwhile, we found that the smaller the $m$ value, the more significant improvement can be achieved by the proposed algorithms. This result shows that the proposed technique is effective and especially crucial when selecting a small subset of entries. In addition, the proposed GES algorithm is comparable or even better than the performance of using all entries. Finally, we found that there is no significant difference between the *cluster*-based measure and the *centroid*-based measure. This is somewhat surprising as we expect to reveal the multiple topics better through the cluster-based measure than centroid-based measure. This may be explained that most of the blogs have only one category in our dataset, which indicates that most of the entries follow only one theme within a blog. At last, we list the numerical results in Table 3 where the sampled entry number is set to 20.

## 6.4 Comparison of Classification Algorithms

To further examine the effectiveness of the set of selected entries $S_i$, we conduct an experiment for comparing a number of competing classification techniques on the set of selected entries $S_i$ with the proposed entry selection technique.

### 6.4.1 Evaluation of Different Classification Methods

Table 4 shows the performance evaluation for comparing a number of classification algorithms based on 10 selected entries of each blog by GES(centroid). From the experimental results, we can see that the algorithms based on average entry distances, including SIL-Cen, MIL-$K_{RBF}^{AVG}$ and MIL-NSK, perform significantly better than those based on distance between a single entry pair, including MIL-$K_{RBF}^{MIN}$, MIL-$K_{RBF}^{MAX}$ and MIL-$K_{RBF}^{HAU}$. This is not surprising since the blogs' entries are noisy and diverse in nature. Therefore, the distance between a single entry pair is not sufficient to determine the distance between blogs. From the encouraging result of Table 4, we also conclude that the selected $S_i$ can represent $B_i$ well because proper classification algorithms can result in comparable accuracy using as few as 10 entries for each blog with the one obtained from an inappropriate classifier using the total entries.

Table 4: Performance of classification algorithms

| Algorithms | Micro-F | Macro-F |
|---|---|---|
| SIL-Cen | .6956 | .6889 |
| MIL-$K_{RBF}^{MIN}$ | .4956 | .5151 |
| MIL-$K_{RBF}^{MAX}$ | .2895 | .2832 |
| MIL-$K_{RBF}^{AVG}$ | .6943 | .7041 |
| MIL-$K_{RBF}^{HAU}$ | .5335 | .5449 |
| MIL-NSK | **.6961** | **.7053** |
| cit-KNN | .4116 | .4373 |

### 6.4.2 Efficiency Advantage of using $S_i$

For the proposed set distance based kernels together with an SVM classifier in 5.2, the time complexity is $\mathcal{O}(N^2k^2 + N^3)$, where $N$ is the total number of blogs, and $k$ is the average number of entries in each blog. The first complexity term $N^2 * k^2$ is for pre-calculating the kernels while the second term $N^3$ is for solving the resulting SVM optimization (assume that an SVM solver with cubic complexity is applied). Such time complexity cost is challenging for a large scale or even moderates scale problem. For example, our testbed, a moderate-sized dataset, consisting of 5,000 blogs, has a total number of entries over 800,000 and around 160 entries for each blog on average.

Apparently it is computationally intensive to engage all blog entries for classification without the help of entry selection techniques. The proposed entry selection methodology can effectively reduce the computational cost for improving the efficiency of blog classification tasks. In particular, by choosing about 10 representative entries for each blog (around 6.25% of all entries on average) with the proposed entry selection algorithms, we can achieve reasonable classification performance comparable to that of using all entries. As a result, the efficiency of the classification task can be improved significantly. Due to the intractability of engaging all entries for classification, we do not report the numerical results of efficiency evaluation.

# 7. CONCLUSIONS AND FUTURE WORK

This paper investigates a new research problem of blog entry selection for profiling blogs. We first formally define the entry selection problem and then propose three principles for guiding the entry selection task, including *anomaly*, *representativeness*, and *diversity*. To develop an effective entry selection method, we further formulate the problem into a general optimization framework, which belongs to a combinatorial optimization problem in nature. To develop an efficient solution, we propose a greedy yet effective algorithm that can solve the entry selection task efficiently. To evaluate the performance of the proposed blog entry selection techniques, we have conducted a series of experiments on blog classification. The encouraging results show that

the proposed algorithms are effective and promising. In future work, we will study more effective blog entry selection techniques for improving the performance.

## Acknowledgement

The work described in this paper was supported in part by two grants: Singapore Academic Research Fund (AcRF) Tier 1 Research Grant (RG67/07) and A*STAR public sector R&D (Singapore), project number 062 101 0031.

## 8. REFERENCES

[1] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In *NIPS*, pages 561–568, 2002.

[2] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[3] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[4] R. C. Bunescu and R. J. Mooney. Multiple instance learning for sparse positive bags. In *ICML*, pages 105–112, 2007.

[5] J. G. Carbonell and J. Goldstein. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. of SIGIR*, 1998.

[6] M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1(3), 1997.

[7] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artif. Intell.*, 89(1-2):31–71, 1997.

[8] T. Gärtner, P. A. Flach, A. Kowalczyk, and A. J. Smola. Multi-instance kernels. In *ICML*, pages 179–186, 2002.

[9] X. Geng, T.-Y. Liu, T. Qin, and H. Li. Feature selection for ranking. In *Proc. of SIGIR*, pages 407–414, 2007.

[10] G.Mishne. Experiments with mood classification in blog posts. In *Proc. of Style Workshop in conj. with SIGIR*, 2005.

[11] S. C. Hoi, R. Jin, and M. R. Lyu. Large-scale text categorization by batch mode active learning. In *Proceedings of the 15th International World Wide Web conference (WWW2006)*, Edinburgh, England, UK, May 23–26 2006.

[12] S. C. Hoi, R. Jin, J. Zhu, and M. R. Lyu. Batch mode active learning and its application to medical image classification. In *Proceedings of the 23rd International Conference on Machine Learning (ICML2006)*, Pittsburgh, PA, US, June 25–29 2006.

[13] M. Hu, A. Sun, and E.-P. Lim. Comments-oriented blog summarization by sentence extraction. In *Proc. of CIKM '07*, pages 901–904, Lisbon, Portugal, 2007.

[14] A. Jain and D. Zongker. Feature selection: Evaluation , application, and small sample performance. *IEEE Trans. on PAMI*, 19(2):153–158, 1997.

[15] K.T.Durant and M. Smith. Mining sentiment classification from political web logs. In *Proc. of WebKDD workshop in conj. with ACM SIGKDD*, Philadelphia, PA, August 2006.

[16] R. Kumar, J. Novak, P. Raghavan, and A. Tomkins. On the bursty evolution of blogspace. In *WWW*, pages 568–576, 2003.

[17] H. Liu and H. Motoda. On issues of instance selection. *Data Min. Knowl. Discov.*, 6(2):115–130, 2002.

[18] O. Maron and T. Lozano-Pérez. A framework for multiple-instance learning. In *NIPS*, 1997.

[19] X. Ni, G.-R. Xue, X. Ling, Y. Yu, and Q. Yang. Exploring in the weblog space by detecting informative and affective articles. In *Proc. of WWW*, pages 281–290, 2007.

[20] T. F. Pranam Kolari and A. Joshi. Svms for the blogosphere: Blog identification and splog detection. In *AAAI Spring Symposium on Computational Approaches to Analysing Weblogs*, March 2006.

[21] T. Reinartz. A unifying view on instance selection. *Data Min. Knowl. Discov.*, 6(2):191–210, 2002.

[22] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5:197–227, 1990.

[23] F. Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, 2002.

[24] V. N. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

[25] J. Wang and J.-D. Zucker. Solving the multiple-instance problem: A lazy learning approach. In *ICML*, pages 1119–1125. Morgan Kaufmann, San Francisco, CA, 2000.

[26] A. Woznica, A. Kalousis, and M. Hilario. Distances and (indefinite) kernels for sets of objects. In *ICDM*, pages 1151–1156, 2006.

[27] D. Xin, H. Cheng, X. Yan, and J. Han. Extracting redundancy-aware top-k patterns. In *Proc. of SIGKDD*, pages 444–453, 2006.