Linking Wikipedia Entries to Blog Feeds by Machine Learning

Mariko Kawaba
NTT Cyber Space Laboratories, NTT Corporation, Yokosuka, Kanagawa, 239-0847, JAPAN

Hiroyuki Nakasaki Daisuke Yokomoto Takehito Utsuro
University of Tsukuba
Tsukuba, 305-8573, JAPAN

Tomohiro Fukuhara
University of Tokyo, Kashiwa
277-8568, JAPAN

ABSTRACT

This paper studies the issue of conceptually indexing the blogosphere through the whole hierarchy of Wikipedia entries. This paper proposes how to link Wikipedia entries to blog feeds in the Japanese blogosphere by machine learning, where about 300,000 Wikipedia entries are used for representing a hierarchy of topics. In our experimental evaluation, we achieved over 80% precision in the task.

Categories and Subject Descriptors

H.3.0 [**INFORMATION STORAGE AND RETRIEVAL**]: General

General Terms

Keywords

blogosphere, Wikipedia, blog feed retrieval, topics

1. INTRODUCTION

Weblogs or blogs are considered to be one of personal journals, market or product commentaries. While traditional search engines continue to discover and index blogs, the blogosphere has produced custom blog search and analysis engines, systems that employ specialized information retrieval techniques. With respect to blog analysis services on the Internet, there are several commercial and non-commercial services such as *Technorati*[1], *BlogPulse*[2] [6], *kizasi.jp*[3], and *blogWatcher*[4] [13]. With respect to multilingual blog services, *Globe of Blogs*[5] provides a retrieval function of blog articles across languages. *Best Blogs in Asia Directory*[6] also provides a retrieval function for Asian language blogs. *Blogwise*[7] also analyzes multilingual blog articles.

In terms of conceptual indexing of the blogosphere, existing services for blog retrieval can be roughly divided into two types. The first type is that of keyword based blog search function of search engines such as *Yahoo! blog search*[8] and *Google blog search*[9]. In this type of indexing, not only keywords, but also subjective expressions as well as time series changes are used for indexing. This type of indexing is too fine-grained compared to actual needs for indexing the blogosphere. Since the number of indices is extremely huge, it is definitely impossible for users to grasp the whole structure of the index hierarchy. Therefore, unless each user comes up with queries appropriate for their search needs in the blogosphere, it is difficult for them to easily access the blogosphere with certain information needs. The second type is that of manually indexing the blogosphere through a directory of manually created categories such as *Technorati*. This type of indexing is, on the other hand, too coarse-grained compared to actual needs for indexing the blogosphere, and such indexing lacks coverage in the whole blogosphere. It is also quite difficult to manually update such a directory of categories when blog feeds of new topics are created in the blogosphere.

Based on this observation, this paper takes an approach of conceptually indexing the blogosphere through the whole hierarchy of Wikipedia entries. In our approach, we regard Wikipedia as a large scale ontological knowledge base for conceptually indexing the blogosphere. We regard Wikipedia also as a large scale encyclopedic knowledge base which includes well known facts and relatively neutral opinions. In its Japanese version, about 615,000 entries are included (checked at September, 2009). For the purpose of conceptually indexing the blogosphere, Wikipedia has an advantage over any other ontological knowledge resource. Although many blog feeds with new topics keep being created rapidly, in Wikipedia, new entries for describing those new topics are also rapidly created, and existing entries also keeps being updated rapidly.

More specifically, this paper proposes how to link Wikipedia

[1] `http://technorati.com/`
[2] `http://www.blogpulse.com/`
[3] `http://kizasi.jp/` (in Japanese)
[4] `http://blogwatcher.pi.titech.ac.jp/` (in Japanese)
[5] `http://www.globeofblogs.com/`
[6] `http://www.misohoni.com/bba/`
[7] `http://www.blogwise.com/`
[8] `http://blog-search.yahoo.co.jp` (in Japanese)
[9] `http://blogsearch.google.co.jp` (in Japanese)

entries to blog feeds in the Japanese blogosphere, where about 300,000 Wikipedia entries are used for representing a hierarchy of topics. In the following sections, first, we examine correlation between the number of hits of a Wikipedia entry title and existence of blog feeds to be linked from the entry. We empirically examine the range of the number of hits and conclude that the entries with the range 10,000 $\sim$ 500,000 tend to have relevant blog feeds. Actually, according to our manual evaluation of this range, about 80% of Wikipedia entries have at least one relevant blog feed. Second, we apply SVMs to the task of judging whether a blog feed is relevant to a given Wikipedia entry. In this task, we train three classifiers, where the first one is applicable to all the Wikipedia entries with the hits 1,000 $\sim$ 10,000, the second is applicable to all the Wikipedia entries with the hits 10,000 $\sim$ 500,000, and the third is applicable to all the Wikipedia entries with the hits over 500,000. In our experimental evaluation, we achieved over 80% precision in those tasks.

2. WIKIPEDIA

In the evaluation of this paper, we collected about 400,000 Japanese entries in November, 2007, and removed entries such as data logs and historical eras as noise. The resulting 305,986 entries are used in the evaluation. The hierarchical structure of Wikipedia can be represented as a undirected graph of categories, where at each category, one or more Wikipedia entries are listed. The version we used in this paper has 29,970 categories and from the root category, topmost 8 categories, namely, *"academia", "technology", "nature", "society", "geography", "humans", "culture"*, and *"history"* are directly connected. From those topmost 8 categories, about 700 categories are directly connected.

3. CRITERION ON JUDGING RELEVANCE BETWEEN A WIKIPEDIA ENTRY AND A BLOG FEED

In this paper, in order to judge whether a blog feed is relevant to the description in a Wikipedia entry, we roughly follow the criterion studied in the blog distillation task [10] in TREC 2007 blog track. The blog distillation task can be summarized as *Find me a blog with a principle, recurring interest in X*. For a given target X, systems should suggest feeds that are principally devoted to X over the time span of the feed, and would be recommended to subscribe to as an interesting feed about X. Here, systems analyze the multiple posts of a given feed.

4. ANALYZING CORRELATION BETWEEN HITS OF THE WIKIPEDIA ENTRY TITLE AND EXISTENCE OF BLOG FEEDS

In this section, before applying machine learning techniques to the task of judging relevance between a Wikipedia entry and a blog feed, we examine the issue of the number of hits of each Wikipedia entry title in the blogosphere. More specifically, in the analysis on existence of blog feeds to be linked from a Wikipedia entry, we examine correlation between the number of hits of a Wikipedia entry title and existence of blog feeds to be linked from the entry.

4.1 Preliminary Analysis

We first identified the range of the numbers of the hits of a Wikipedia entry title in the Japanese blogosphere, where blog feeds to be linked from each entry most exist compared to the rest range. The resulting range is over 10,000. We further examined the tendency of blog feeds linked from Wikipedia entries and observed certain differences between the range of 10,000 $\sim$ 500,000 and that over 500,000. In the range over 500,000, most Wikipedia entry titles can be regarded as general terms and some of the blog feeds linked from those entries should be linked rather from descendant concepts of those entries. With this result, we classify the numbers of the hits in the Japanese blogosphere into the three ranges, i.e., 1,000 $\sim$ 10,000, 10,000 $\sim$ 500,000, and over 500,000. Out of the titles of the whole 305,986 Wikipedia entries, about 8% (24,075 entries) are with the number of hits as zero, about 56% (172,471 entries) are with 1 $\sim$ 1,000, about 21% (63,835 entries) are with 1,000 $\sim$ 10,000, about 14% (40,852 entries) are with 10,000 $\sim$ 500,000, and about 1% (4,753 entries) are with those over 500,000.

4.2 Sample Wikipedia Entries for Evaluation

In the strategy of sampling Wikipedia entries for evaluation, we take an approach of selecting Wikipedia entries through Wikipedia categories. Here, we prefer sample entries to be distributed over various categories, simply because future application of this work is to estimate the topic distribution in the Japanese blogosphere. The detailed procedure is given below:

1. We first heuristically allocate each Wikipedia entry to three of the topmost 8 categories or the second topmost 700 categories. For each Wikipedia entry, we ranked categories according to ascending order of the distance (here, we use the number of edges) from the entry, and selected the topmost three categories. We then restrict the candidate categories as those with more than 50 entries allocated in this procedure. Finally, we randomly select the following 35 categories from those candidates, where each category is shown with the number of allocated entries:

   arts(7526), transportation(5410), computing(3877), landforms(3466), hobbies(2704), infrastructure(2405), sound(2403), subcultures(1806), real estate (1464), family(1198), law(1112), chemistry(950), information technology (940), biology(838), human geography (817), history of technology (798), lists(601), military equipment (578), technology of network (532), social issues (510), psychology(468), linguistics(383), forests(340), interpersonal relationships (292), qualifications(245), paleontology(202), fire(199), artistic techniques (190), sales techniques (182), automotive technologies (180), minorities(165), earth sciences (146), energy(116), human rights (89), historians(50)

2. Next, from each of the 35 categories, we selected 75 entries from the range of 1,000 $\sim$ 10,000 of the hits in the Japanese blogosphere, 168 from that of 10,000 $\sim$ 500,000, and 149 from that over 500,000, which amount to 392 in total.

Table 1: Manual Analysis on Existence of Blog Feeds to be Linked from a Wikipedia Entry (with top ranked 20 feeds): Criterion

Label	Description
C1	20∼10 feeds relevant to the given entry.
C2	9∼5 feeds relevant to the given entry.
C3	4∼1 feed(s) relevant to the given entry.
HU	At least one feed is relevant to an immediate ascendant concept of the given entry.
HL	At least one feed is relevant to an immediate descendant concept of the given entry.
HR	At least one feed is relevant to a related concept of the given entry.
E	None of above.

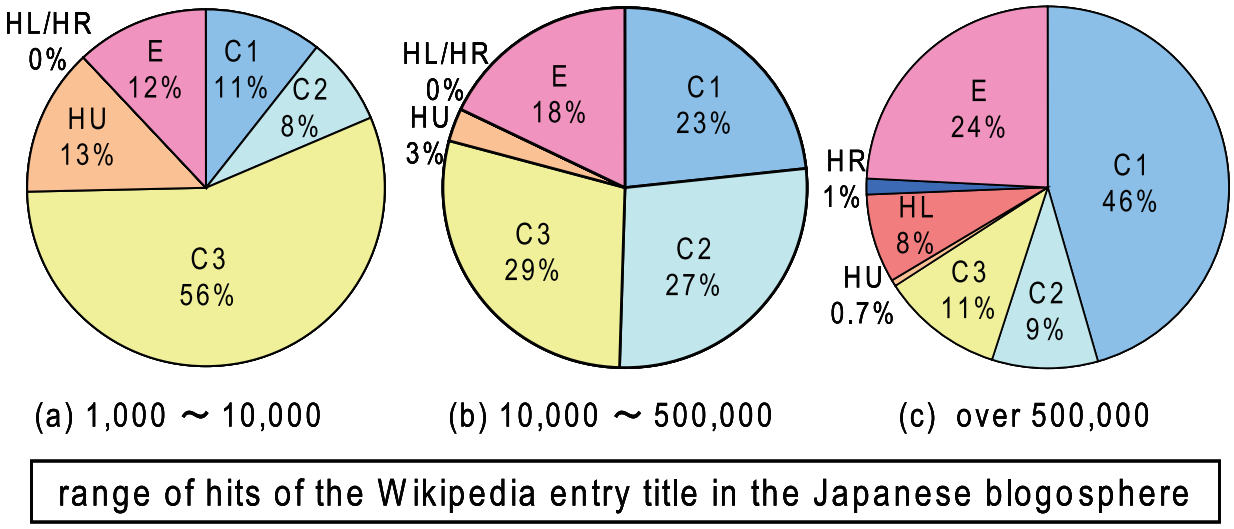

Figure 1: Manual Analysis on Existence of Blog Feeds to be Linked from a Wikipedia Entry: Distribution per Range of Hits of the Wikipedia Entry Titles

4.3 Collecting Blog Feeds for Evaluation

For each of the sample Wikipedia entries for evaluation, we consider the title of the entry as the query for collecting blog feeds for evaluation. This section briefly describes how to retrieve Japanese blog feeds for evaluation given a query [9, 12].

First, in order to collect candidates of blog feeds for a given query, we use existing Web search engine APIs, which return a ranked list of blog posts, given a topic keyword. We use the Japanese search engine "Yahoo! Japan" API[10], where blog hosts are limited to major 11 ones[11].

Next, we employ the following procedure for collecting blog feeds:

i) Given a query keyword, a ranked list of 1,000 blog posts are returned by the Web search engine API.

ii) A list of blog feeds is generated from the returned ranked list of blog posts by simply removing duplicated feeds.

iii) Re-rank the list of blog feeds according to the number of hits of the query keyword in each blog feed. The number of hits for a query keyword in each blog feed is simply measured by the search engine API used for collecting blog posts above in i), restricting the domain of the URL to each blog feed.

iv) Finally, for each query keyword, top-ranked 20 blog feeds are collected, and are used for evaluation.

In terms of the criterion presented in section 3, [9, 12] reported that the procedure above outperformed the original ranking returned by "Yahoo! Japan" API, although the evaluation is small scale.

It is important to note here that future application of this work is not to generally detect the topic of a given blog feed, but to estimate the topic distribution in the Japanese blogosphere. In this application, it is more important to judge whether there exist blog feeds to be linked from a given Wikipedia entry. According to our preliminary analysis, whether there exist blog feeds to be linked from a given

[10] `http://www.yahoo.co.jp/` (in Japanese)

[11] `FC2.com,yahoo.co.jp,rakuten.ne.jp,ameblo.jp,goo.ne.jp,livedoor.jp,Seesaa.net,jugem.jp,yaplog.jp,webry.info.jp,hatena.ne.jp`

Table 2: Features of Linking Wikipedia Entries to Blog Feeds

Feature Label	Description
t-hits	hits of the given Wikipedia entry title, where the number of hits is counted in the blog feed
H-hits	sum of the hits of all the related terms in the range of hits over 500,000, where the numbers of hits are counted in the blog feed
M-hits	sum of the hits of all the related terms in the range of hits 10,000 $\sim$ 500,000, where the numbers of hits are counted in the blog feed
L-hits	sum of the hits of all the related terms in the range of hits below 10,000, where the numbers of hits are counted in the blog feed
H-num-b	the number of the related terms in the range of hits over 500,000, which are observed in the blog feed
M-num-b	the number of the related terms in the range of hits 10,000 $\sim$ 500,000, which are observed in the blog feed
L-num-b	the number of the related terms in the range of hits below 10,000, which are observed in the blog feed
all-num-b	the number of all the related terms, which are observed in the blog feed
H-num-w	the number of the related terms in the range of hits over 500,000, which are collected from the given Wikipedia entry
M-num-w	the number of the related terms in the range of hits 10,000 $\sim$ 500,000, which are collected from the given Wikipedia entry
L-num-w	the number of the related terms in the range of hits below 10,000, which are collected from the given Wikipedia entry
all-num-w	the number of all the related terms, which are collected from the given Wikipedia entry
char-len	character length of the given Wikipedia entry title
1-char	true if the given Wikipedia entry title is one kanji (Chinese character) word

Wikipedia entry can be mostly judged considering the top-ranked 20 blog feeds returned by the procedure above.

4.4 Results of the Analysis

For each of the 392 entries selected in section 4.2, 20 blog feeds are collected according to the procedure of the previous section, and their relevance are manually judged based on the criterion presented in section 3. Then, based on the judgments of 20 blog feeds, each entry is assigned one of 7 labels listed in Table 1. Final statistics is shown in Figure 1, where the distribution of those 7 labels is given for each of the three ranges.

As can be seen from this result, it is clear that entries with the number of hits over 10,000 tend to have many relevant blog feeds in the Japanese blogosphere. Furthermore, for the entries with the number of hits around over 500,000, some of the blog feeds linked from them are judged to be more relevant to an immediate descendant concept of those entries. Examples of the entries with the number of hits around 10,000 $\sim$ 500,000 which have relevant blog feeds are *"kitchen garbage", "adoption", "department store's basement food floor"*, and *"guide dog"*. On the other hand, entries with those around 1,000 $\sim$ 10,000 tend to have relatively small number of relevant blog feeds. Examples of those entries are usually very specific ones such as *"Xenopus"*, which is a genus of highly aquatic frogs native to Sub-Saharan Africa. Finally, as for examples of entries with the number of hits over 500,000, entries such as *"piano"* have many relevant blog feeds.

5. LINKING WIKIPEDIA ENTRIES TO BLOG FEEDS BY SVM

5.1 The Procedure of Training/Testing

This section describes the procedure of applying Support Vector Machines (SVMs) [16] to the task of judging relevance between a Wikipedia entry and a blog feed.

As a tool for learning SVMs, we use TinySVM (`http://chasen.org/~taku/software/TinySVM/`). Each training/test instance of SVMs learning is represented as a tuple $\langle b_e, c \rangle$, where b_e denotes a blog feed collected with a title $t(e)$ of a Wikipedia entry e, and the class c denotes whether b_e is relevant to the Wikipedia entry e (i.e., "$c = +$") or not (i.e., "$c = -$").

The 392 sample entries selected in section 4.2 are divided into three groups according to the three ranges of the number of the hits in the blogosphere. In this paper, we assume that the optimal set of features may vary according to the three ranges of the number of the hits in the blogosphere. Therefore, we evaluate the SVMs learning technique separately against each of the three groups, and construct sets of training and test instances for each group[12].

For each sample entry, those 20 blog feeds which are manually annotated with relevance judgment are used. Furthermore, in this evaluation, we intend to examine optimal performance of SVMs classifier against the task of this section. To do this, we require that both training and test sets are balanced in terms of the distribution of class labels. For all

[12]We are now experimentally examine whether this assumption is correct or not by simply learning a single classifier for all of the three ranges of the number of the hits. The result will be included in the final version of the paper.

Table 3: Linking Wikipedia Entries to Blog Feeds by SVM: Evaluation Results (%) (*w/o* blog feeds relevant to related concepts)

(a) range of hits of the Wikipedia entry title: 1,000 $\sim$ 10,000

condition	feature set	precision/ recall/ F-measure
baseline	t-hits	62.4/32.6/42.8
maximum F-measure (without lower bound of confidence)	M-hits + H-num-b	63.0/67.0/**64.9**
(maximum precision, without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b	75.1/56.6/64.5
maximum precision (lower bound of confidence as 0.9)	t/M-hits + H/M/L-num-b	**85.1**/24.7/38.3

(b) range of hits of the Wikipedia entry title: 10,000 $\sim$ 500,000

condition	feature set	precision/ recall/ /F-measure
baseline	t-hits	55.6/77.0/64.6
maximum F-measure (without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + char-len	77.2/69.3/**73.0**
(maximum precision, without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + all-num-w + char-len	78.3/61.2/68.7
maximum precision (lower bound of confidence as 1.5)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + char-len	**91.5**/22.2/35.7

(c) range of hits of the Wikipedia entry title: over 500,000

condition	feature set	precision/ recall/ F-measure
baseline	t-hits	60.8/35.7/45.0
maximum F-measure (without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + 1-char	76.9/68.2/**72.3**
(maximum precision, without lower bound of confidence)	t-hits + M-num-b	79.9/54.0/64.5
maximum precision (lower bound of confidence as 1.1)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + 1-char	**85.7**/28.6/42.9

of the three groups, the number of the tuples with the class c as "$c = +$" is smaller than that of the tuples with the class c as "$c = -$". Thus, we randomly ignore some of those with the class c as "$c = -$". Finally, we have 408 instances for the range of 1,000 $\sim$ 10,000, 1,852 for that of 10,000 $\sim$ 500,000, and 2,096 for that over 500,000.

As the kernel function, we compare the linear and the polynomial (2nd order) kernels, where the latter performs better. In this paper, we show the results with the polynomial (2nd order) kernel. Furthermore, in the actual application of the proposed technique, we intend to prefer precision rather than recall, and to detect blog feeds which can be confidently relevant to the given Wikipedia entry. In order to realize this, In the testing of an SVMs classifier, we regard the distance from the separating hyperplane to each test instance as a confidence measure. We then introduce a lower bound against the distance from the separating hyperplane to each test instance, where blog feeds with this distance smaller than the lower bound are rejected. In the actual evaluation, we require that F-measure be at least around 35%, and then the lower bound of the confidence measure is determined so as to maximize the precision.

5.2 Features

Features employed in this paper are summarized in Table 2, where all of them are independent of specific Wikipedia entries. Thus, the classifiers trained with those features are applicable to arbitrary Wikipedia entries.

Most features are based on terms that are closely related to the given Wikipedia entry, and are automatically extracted from the body text of the given entry. The related terms we extract from the body text of the given entry can be categorized as follows: bold-faced terms, anchor texts of hyperlinks, and the title of a *redirect*, which is a synonymous term of the title of the target page. From each entry, we extracted 15 related terms on the average. Then, we further classify those related terms into three ranges according to the number of hits in the Japanese blogosphere. This is simply from the observation in section 4 on the correlation between the number of hits of a Wikipedia entry title and existence of blog feeds to be linked from the entry.

For each of the features 't-hits', 'H-hits', 'M-hits', and 'L-hits', sum of the hits are classified into 5 ranges, and each of the 5 ranges is represented as a binary feature, whose value indicates whether the sum of the hits is within the corresponding range.

5.3 Evaluation Results

Table 3 summarizes the evaluation results by 10-fold cross-validation for each of the three ranges. As the evaluation measure, precision/recall/f-measure of detecting tuples with the class c as "$c = +$" are employed. Here, the performance only with the feature 't-hits' are shown as baseline. This

Table 4: Linking Wikipedia Entries to Blog Feeds by SVM: Evaluation Results (%) (*with* blog feeds relevant to related concepts)

(a) range of hits of the Wikipedia entry title: 1,000 $\sim$ 10,000

condition	feature set	precision/ recall/ F-measure
baseline	t-hits	68.0/33.7/45.1
maximum F-measure (without lower bound of confidence)	M-hits	60.0/70.4/**64.8**
(maximum precision, without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w	70.1/58.6/63.9
maximum precision (lower bound of confidence as 0.8)	t-hits + M-num-b	**77.5**/25.1/37.9

(b) range of hits of the Wikipedia entry title: 10,000 $\sim$ 500,000

condition	feature set	precision/ recall/ /F-measure
baseline	t-hits	56.4/76.3/64.8
maximum F-measure (without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + char-len	77.2/68.8/**72.7**
(maximum precision, without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + all-num-w	79.1/62.0/69.5
maximum precision (lower bound of confidence as 1.5)	t/M-hits + H/M/L-num-b	**91.6**/22.3/35.9

(c) range of hits of the Wikipedia entry title: over 500,000

condition	feature set	precision/ recall/ F-measure
baseline	t-hits	57.0/34.4/42.9
maximum F-measure (without lower bound of confidence)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + 1-char	78.1/68.3/**72.9**
(maximum precision, without lower bound of confidence)	M-hits	80.3/57.4/62.9
maximum precision (lower bound of confidence as 1.7)	t/H/M/L-hits + H/M/L-num-b + H/M/L-num-w + 1-char	**91.8**/22.4/36.0

baseline can be considered as optimizing the ranking of blog feeds according to the number of hits of the Wikipedia entry title in each blog feed, which we employed in section 4.3. We also evaluate all the combination of features as well as several values for lower bounds of confidence, and then pick up results with maximum F-measure and maximum precision with F-measure at least around 35%[13].

We achieved F-measures almost over 65%. On the other hand, the maximum precision is achieved with the lower bound of confidence. We achieved precision over 85%. Those evaluation results clearly show that the features based on statistics of related terms which are obtained both from the Wikipedia entry and from the blog feed are quite effective in this task. The best performance also outperforms the baseline. Following the discussion in section 4.3 and in [9, 12], this result also indicates that the proposed technique outperforms the original ranking returned by "Yahoo! Japan" API.

We further apply SVMs to another task of judging relevance between a Wikipedia entry and a blog feed, where we incorporate blog feeds relevant to related concepts including ascendant concepts and descendant concepts of the given entry. In this new task, each training/test instance of SVMs learning is represented as a tuple $\langle b_e, c\rangle$, where b_e denotes a blog feed collected with a title $t(e)$ of a Wikipedia entry e, and the class c denotes whether b_e is relevant to the Wikipedia entry e or to related concepts (including ascendant concepts and descendant concepts) of e (i.e., "$c = +$") or not (i.e., "$c = -$").

Again, we require that both training and test sets are balanced in terms of the distribution of class labels. For the range of 1,000 $\sim$ 10,000 and that of 10,000 $\sim$ 500,000, the number of the tuples with the class c as "$c = +$" is smaller than that of the tuples with the class c as "$c = -$". So, we randomly ignore some of those with the class c as "$c = -$". For that over 500,000, on the other hand, the number of the tuples with the class c as "$c = -$" is smaller than that of the tuples with the class c as "$c = +$", and we randomly ignore some of those with the class c as "$c = +$". Then, we have 498 instances for the range of 1,000 $\sim$ 10,000, 1,990 for that of 10,000 $\sim$ 500,000, and 2,470 for that over 500,000.

Table 4 summarizes the evaluation results by 10-fold cross-validation for each of the three ranges. Most important differences compared to the results in Table 3 are that we have

[13]Here, for all of the three ranges, maximum F-measure is achieved when without lower-bound of confidence, while maximum precision is achieved with lower bound of confidence as certain values and their values vary in the ranges (a), (b), and (c). Maximum precision when without lower bound of confidence is also shown for comparison.

improvement in the precision for the range over 500,000, while we have damage in the precision for the range of 1,000 $\sim$ 10,000. This is mainly because, for the range of 1,000 $\sim$ 10,000, blog feeds added in this new task are relevant to ascendant concepts of the given Wikipedia entry, while for that over 500,000, those are relevant to descendant concepts of the given Wikipedia entry. We manually examine the evaluation results and found that blog feeds which are added for the range of 1,000 $\sim$ 10,000 and are relevant to ascendant concepts of the given Wikipedia entry tend to have less related terms extracted from the given Wikipedia entry. This means that, for those blog feeds, less features are active in SVMs training/testing.

6. RELATED WORKS

The blog distillation task [10] in TREC 2007 blog track is related to the task we examine in this paper. Among the participants of the task, the best performing system was [4], which employs query expansion using hyperlinks in Wikipedia. Compared to this approach, we integrate more information other than hyperlinks, such as bold-faced terms and the title of a *redirect*, where those terms are carefully distinguished in terms of their numbers of hits in the Japanese blogosphere. We further examine their effectiveness through features of the SVM learning framework.

Previous works on automatically suggesting tags to blog posts [2, 11, 14] are also related to our work, where major differences are in that those works study to suggest tags to blog posts but not to blog feeds. [11, 14] studied to suggest existing tags such as those available from *Technorati*, where, as we discussed in section 1, the approach of indexing the blogosphere through a directory of manually created categories has its own limitation. [2] studied to automatically extract tag candidates from blog posts.

Previous works on text classification [5, 17] as well as text clustering [7, 8] using Wikipedia knowledge are also based on techniques which extract related terms such as hyponyms, synonyms, and associated terms. Among them, text classification using Wikipedia knowledge [5, 17] apply machine learning techniques to the task of classifying documents into certain number of classes, where Wikipedia knowledge are used as features. Major differences between our work and those works are: i) first of all, the underlying purpose of our work is to estimate topic distribution in the blogosphere, where Wikipedia is used as the topic hierarchy, ii) the overall frameworks of evaluation differ. In our evaluation, given a Wikipedia entry as a sample topic, candidate blog feeds are collected and their relevance to the given topic are examined through the proposed technique, iii) in our machine learning framework, the specific design of features is different from previous works.

Compared to previous works on classifying documents into a hierarchical directory [3, 15, 1], this paper mostly focuses on the use of the hierarchy of Wikipedia entries as one of the largest hierarchy of concepts. We further show that the related terms extracted from the body text of each Wikipedia entry are quite effective in the task of relevant judgment of blog feeds.

7. CONCLUDING REMARKS

This paper studied the issue of conceptually indexing the blogosphere through the whole hierarchy of Wikipedia entries. More specifically, this paper proposed how to link Wikipedia entries to blog feeds in the Japanese blogosphere, where about 300,000 Wikipedia entries are used for representing a hierarchy of topics. In our experimental evaluation, we achieved over 80% precision in those tasks.

8. REFERENCES

[1] G. Adami, P. Avesani, and D. Sona. Clustering documents in a Web directory. In *Proc. WIDM*, 2003.

[2] C. Brooks and N. Montanez. Improved annotation of the blogopshere via autotagging and hierarchical clustering. In *Proc. 15th WWW*, 2006.

[3] S. Dumais and H. Chen. Hierarchical classification of Web content. In *Proc. 23rd SIGIR*, pages 256–263, 2000.

[4] J. Elsas, J. Arguello, J. Callan, and J. Carbonell. Retrieval and feedback models for blog distillation. In *Proc. TREC-2007 (Notebook)*, pages 170–175, 2007.

[5] E. Gabrilovich and S. Markovitch. Overcoming the brittleness bottleneck using Wikipedia: Enhancing text categorization with encyclopedic knowledge. In *Proc. 21st AAAI*, pages 1301–1306, 2006.

[6] N. Glance, M. Hurst, and T. Tomokiyo. Blogpulse: Automated trend discovery for Weblogs. In *WWW 2004 Workshop on the Weblogging Ecosystem: Aggregation, Analysis and Dynamics*, 2004.

[7] J. Hu, L. Fang, Y. Cao, H.-J. Zeng, H. Li, Q. Yang, and Z. Chen. Enhancing text clustering by leveraging Wikipedia semantics. In *Proc. 31st SIGIR*, pages 179–186, 2008.

[8] A. Huang, E. Frank, and I. H. Witten. Clustering document using a Wikipedia-based concept representation. In *Proc. 13th PAKDD*, pages 628–636, 2009.

[9] M. Kawaba, H. Nakasaki, T. Utsuro, and T. Fukuhara. Cross-lingual blog analysis based on multilingual blog distillation from multilingual Wikipedia entries. In *Proc.ICWSM*, pages 200–201, 2008.

[10] C. Macdonald, I. Ounis, and I. Soboroff. Overview of the TREC-2007 blog track. In *Proc. TREC-2007 (Notebook)*, pages 31–43, 2007.

[11] G. Mishne. AutoTag: A collaborative approach to automated tag assignment for Weblog posts. In *Proc. 15th WWW*, 2006.

[12] H. Nakasaki, M. Kawaba, T. Utsuro, T. Fukuhara, H. Nakagawa, and N. Kando. Cross-lingual blog analysis by cross-lingual comparison of characteristic terms and blog posts. In *Proc. 2nd International Symposium on Universal Communication*, pages 105–112, 2008.

[13] T. Nanno, T. Fujiki, Y. Suzuki, and M. Okumura. Automatically collecting, monitoring, and mining Japanese weblogs. In *WWW Alt. '04: Proceedings of the 13th international World Wide Web conference on Alternate track papers & posters*, pages 320–321. ACM Press, 2004.

[14] S. Sood, S. Owsley, K. Hammond, and L. Burnbaum. TagAssist: Automatic tag suggestion for blog posts. In *Proc. ICWSM*, 2007.

[15] A. Sun and E.-P. Lim. Hierarchical text classification and evaluation. In *ICDM*, pages 521–528, 2001.

[16] V. N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998.

[17] P. Wang and C. Domeniconi. Building semantic kernels for text classification using Wikipedia. In *Proc. 14th SIGKDD*, pages 713–721, 2008.