An SVM Based Approach to Feature Selection for Topical Relevance Judgement of Feeds

Yongwook Shin and Jonghun Park
Information Management Lab.
Seoul National University
599 Gwanak-ro, Gwanak-gu, Seoul, Korea
{yongwook, jonghun}@snu.ac.kr

ABSTRACT

Relevance judgement method is an essential part of a feed search engine, which determines candidates for ranking query results. However, the different characteristics of feeds from traditional web pages may make existing relevance judgement approaches proposed for web page retrieval produce unsatisfactory result. Compared to web pages, feed is a structured document in that it contains several data elements, including title and description. In addition, feed is a temporal document since it dynamically publishes information on some specific topics over time. Accordingly, the relevance judgement method for feed retrieval needs to effectively address these unique characteristics of feeds. This paper considers a problem of identifying significant features which are a feature set created from feed data elements, with the aim of improving effectiveness of feed retrieval. We conducted extensive experiments to investigate the problem using support vector machine on real-world data set, and found the significant features that can be used for feed search services.

Categories and Subject Descriptors

H.3.3 [**Information Systems**]: Information Search and Retrieval; H.3.5 [**Information Systems**]: Online Information Services; H.5.4 [**Information Systems**]: Hypertext/Hypermedia

General Terms

Experimentation, Measurement, Performance

Keywords

Information Retrieval, Search Engine, Feed Search, Feed, Feature Selection, Relevance Judgement, Text Classification

1. INTRODUCTION

The number of web sites publishing feeds is dramatically increasing and it is now common to acquire information by subscribing to feeds. Feed is an XML document published by a web site to facilitate syndication of its contents to subscribers. News and blogs are common sources for feeds, and recently social media and microblogging sites are increasingly delivering information through feeds.

Feed search engine that can help users effectively seek for feeds becomes necessary in order to address the challenges imposed by recent explosion of feeds. Through feed search engine, users can discover feeds for the purpose of subscription in order to keep themselves updated with recent information. Feed search engine takes a query from users and generates ranking candidates which are feeds judged to be relevant to the query. Afterwards, it returns a ranked list of feeds by scoring the candidates based on a specific feed retrieval model. Accordingly, relevance judgement method that detemines the ranking candidates plays an important role in enhancing feed search quality.

Under binary relevance judgement framework, the relevance judgement method classifies feeds as relevant or non-relevant against a user query in order to construct the ranking candidates [11]. In particular, in relevance judgement for feed retrieval using feature-based model, the relevance is defined by using a feature set constructed from a user query and a feature set from a feed. It is necessary that a feature set for a feed is constructed by selecting specific features suited to feed relevance judgement. Unfortunately, existing relevance judgement methods for web page retrieval may not be appropriate for relevance judgement for feed retrieval since they are different in terms of available features.

Specifically, relevance judgement problem for feed retrieval poses interesting challenges due to two different properties of feeds, compared to web pages. Feed is a structured as well as temporal document in a sense that it contains several data elements, including a feed title, a feed description, and multiple entries each of which consists of a title and description dynamically published over time, whereas web pages tend to be rather static. With the structural nature of feed, the problem is to find out which data elements need to be selected to define a feature set for relevance judgement. The temporal charateristic of feed raises a problem of determining how many entries need to be considered for relevance judgement.

To the best of authors' knowledge, there has been no study that attempted to define features used for feed relevance judgement method by considering the unique structural and temporal characteristics of feed. In this paper, we attempt to identify significant features for feed relevance judgement with respect to a user query. The significant features are

ACM SIGIR'10 19-23 July 2010, Geneva, Switzerland

defined as a feature set constructed from feed data elements that can improve effectiveness of feed retrieval.

Ranking function for blog feed retrieval model was studied previously, taking account of unique properties of feeds. Research in [4] investigated whether it is more effective for feed retrieval to view a feed as a single document or multiple documents composed of entries. However, it did not pay attention to feature selection problem for feed relevance judgement. In addition, temporal characteristics of feed as well as structural elements such as feed title and feed description were not considered in [4].

An enhanced ranking model for blog posts, named PTRank, was proposed to improve search quality beyond the simple keyword matching, utilizing various information available from blog feeds [7]. Yet, it suggested the scoring function to assess the degree of relevance between a query and blog posts, instead of selecting the significant featues for feed relevance judgement. Furthermore, temporal characteristic of feed was not considered by PTRank.

In this paper, we attempt to identify the significant features to judge relevance for feed retrieval through feed classification based on a topic using a topic-labeled feed data set. The rest of the paper is organized as follows. Section 2 defines the problem and presents our proposed approach to identify the significant features. In Section 3, we report experimental results and finally, we give concluding remarks in Section 4.

2. PROPOSED APPROACH

Feed is an XML file that contains partial or full descriptions of web page articles along with links to the original contents and other information. Two popular feed formats are RSS (Really Simple Syndication) and Atom [6]. Figure 1 shows a typical example that illustrates major elements of an Atom feed.

Specifically, the schema of Atom 1.0 includes two core elements, namely <feed> and <entry>. <feed> is the root element of an Atom document and contains feed title and description. <feed> should have at least one <entry> element, and it contains mandatory sub-elements, <title> and <subtitle>. <entry> element includes <title> and <summary> elements. While <title> contains the title of a web page article, <summary> element has a short summary or full body of the article.

Among the various feed elements, we consider four elements in identifying the significant features for feed relevance judgement. We refer to <title> element in <feed> as a feed title, and <subtitle> element in <feed> as a feed description. We also refer to <title> element of <entry> as an entry title and <summary> element of <entry> as an entry description.

This study concentrates on identification of the significant features for feature-based relevance judgement method in feed retrieval. The significant features are defined as a feature set created from data elements available from a feed that can maximize effectiveness of relevance judgement.

Specifically, we use a vector space model for representing the features. The notations used in our problem are presented as follows. Given the set of m feeds, $F = \{f_1, f_2, ..., f_m\}$, and the set of terms, T, let $E_i = \{e_{ij} | j = 1, ..., n\}$ represent the set of entries for $f_i, i = 1, ..., m$, constructed by choosing the most recent n entries from the set of all entries published from f_i. Our vecter space model is based on a bag of terms representation, and we denote the term bags of feed title and feed decription as ft_i and fd_i, respectively. The term bag of title in e_{ij} is represented as et_{ij}, while the term bag of description in e_{ij} is represented as ed_{ij}.

```
<?xml version="1.0" encoding="utf-8"?>
<feed xmlns="http://www.w3.org/2005/Atom">
    <title>Example Feed</title>
    <subtitle>A subtitle.</subtitle>
    <link href="ttp://example.org/feed/" rel="self" />
    <link href="http://example.org/" />
    <updated>2003-12-13T18:30:02Z</updated>
    <entry>
        <title>Atom-Powered Robots Run Amok</title>
        <link href="http://example.org/2003/12/13/atom03" />
        <link rel="alternate" type="text/html"
        href="http://example.org/2003/12/13/atom03.html"/>
        <updated>2003-12-13T18:30:02Z</updated>
        <summary>Some text.</summary>
    </entry>
</feed>
```

Figure 1: An example Atom feed

In the following, we use notation B for representing a bag of terms, and $|B|$ for the total number of term occurrences in B. In addition, $tf(t, B)$ represents term frequency (TF) of term $t \in T$ in term bag B.

Let Q denote a set of queries, and $q \in Q$ a term set representation of a query. Given query q and E_i, f_i is modeled as a feature vector represented by ($\sigma(q, ft_i)$, $\sigma(q, fd_i)$, $\sigma(q, et_{i1})$, ..., $\sigma(q, et_{in})$, $\sigma(q, ed_{i1})$, ..., $\sigma(q, ed_{in})$). $\sigma(q, ft_i)$, $\sigma(q, fd_i)$, $\sigma(q, et_{ij})$, and $\sigma(q, ed_{ij})$ for $i = 1, ..., m$ and $j = 1, ..., n$ are scoring functions defined as normalized TF, and they indicate feature scores specified for feed title, feed description, entry title, and entry description, respectively. We consider normalized TF among other alternatives, since it is reported that normalized TF is effective in many text classfication problems [8]. We call normalized TF as TF in the following.

Under the assumption of normalized TF, $\sigma(q, B)$ for term bag B is defined as:

$$\sigma(q, B) = \begin{cases} \sum_{t \in q} tf(t, B)/|B| & \text{if } q \in B \\ 0 & \text{otherwise} \end{cases}$$

We attempt to find the significant features through examining various alternatives for constructing the feature vector, considering the unique properties of feed mentioned in the previous section. First, feed data elements constituting the significant features are identified by investigating several combinations of structural data elements in feature vector construction. Second, we investigate the number of entries considered for the feature vector by varying n to address the temporal characteristics of feed.

We employ support vector machine (SVM) for our feed classification task, since it is reported that SVM is one of the most popular and effective supervised learning methods for text classification problems [3], [10]. In this paper, SVM treats feed classification problem as the one involving binary class labels, which we refer to as "non-relevant" and "relevant". SVM builds a model using the given labeled tranining data, which predicts whether an instance of feature vector of a feed falls into "non-relevant" or "relevant".

To evaluate effectiveness of relevance judgement, we use precision, recall, and F measure which are commonly used to compare the relative performance with different features [2].

Table 1: Results produced by various feature vector construction methods

		P	R	FM
FT	0	**.977**	.390	.557
FD	0	.948	.432	.594
FT+FD	0	.956	.556	.703
ET	40	.941	.425	.586
	80	.946	.418	.580
	120	.947	.426	.588
	160	.946	.429	.590
	200	.947	.436	.597
ED	40	.936	.513	.663
	80	.943	.517	.668
	120	.944	.522	.672
	160	.947	.523	.674
	200	.944	.521	.671
FT+FD +ET	40	.953	.599	.736
	80	.959	.586	.727
	120	.959	.576	.720
	160	.959	.564	.710
	200	.959	.557	.705
FT+FD +ED	40	.954	**.612**	**.746**
	80	.955	.609	.744
	120	.957	.610	.745
	160	.959	.604	.741
	200	.958	.598	.736
FT+FD +ET+ED	40	.957	.603	.740
	80	.961	.589	.730
	120	.959	.577	.720
	160	.961	.566	.712
	200	.958	.557	.704

Precision, P, is a measure of the usefulness of feeds judged as "relevant" and recall, R, is a meaure of the completeness of a relevance judgement method. F measure, FM, is defined as the harmonic mean of recall and precision. FM provides a single metric for comparison across different experiments.

In what follows, among possible configurations of features, the significant features achieve the best performance on a specific evaluation metric. We use notation $SF(EM)$ for representing the significant features identified from experiments on evaluation metric, EM, where EM can be P, R, or FM.

3. EXPERIMENTS

A topic-labeled feed data set for our experiments was collected from "Bundles from Google", which is provided by Google Reader [5], a Google's web-based feed reading service. Several feeds for a topic are bundled in order to serve feed recommendation to Google reader users in "Bundles from Google". We call "Bundles from Google" as "Google data set". Google data set has various topics and broad types of feeds from public media sites, blogs, podcasts, and social media sites. Our data set consists of 3,050 feeds and the size of the most recent entries for each feed is 200. The total number of topics taken as queries is 435, and for each topic, the number of feeds belonging to the topic varies from 3 to 20. On the average, there are 7 feeds per topic.

Baseline features are selected among the features available from feed under the assumption that relevance judgement method using as many as feed data elements and entries possible is most effective. Similarly to the significant features, baseline features are defined in terms of the feed data elements used for constructing the feature vector and the number of entries. Given F, we select ft_i, fd_i, et_{ij}, and ed_{ij}, $i = 1, ..., m$, $j = 1, ..., n$ as structural data elements for baseline features, and set $n = 200$ for the time span of feed

Table 2: Results summarized by feed data elements

	P	R	FM
FT	**.977**	.396	.557
FD	.948	.432	.594
FT+FD	.956	.556	.703
ET	.947	.436	.597
ED	.947	.523	.674
ET+ED	.952	.505	.660
FT+ET	.961	.535	.687
FT+ED	.959	.584	.726
FT+ET+ED	.960	.558	.706
FD+ET	.952	.543	.690
FD+ED	.950	.599	.733
FD+ET+ED	.956	.570	.711
FT+FD+ET	.959	.599	.736
FT+FD+ED	.959	**.612**	**.746**
FT+FD+ET+ED	.961	.603	.740

since the largest number of entries per feed available from our data sets is 200.

For our experiments, we used the libSVM toolkit [1] to train the SVM models. We work with linear SVM models since the existing literature on text classification indicates that the nonlinear versions of SVM gain very little in terms of performance, compared to the linear version [9]. A five fold cross-validation technique is used to evaluate the performance.

Table 1 shows our experiment design and results for TF, and it contains representative results selected from complete results, which include the significant features, the baseline features, feed data elements, and some combinations of them. Table 2 summarizes feed data elements and their combinations from complete results in order to investigate which feed data elements need to be used for constructing a feature vector. In Table 2, precision, recall and F measure of feed data elements and their combinations represent the maximum values produced in our experiments.

In the result tables, FT stands for feed title, FD for feed description, ET for entry title, and ED for entry description. The first column on the left side in the result tables indicates feed data elements and their combinations. We denote combination of feed data elements for the feature vector with "+". In particular, the second column of Table 1 indicates n, where $n = 0$ means that entries are not used for constructing the feature vector. For instance, when feed data element combination is FT+ET and n=40, f_i is modeled as a feature vector represented by $(\sigma(q, ft_i), \sigma(q, et_{i1}), ..., \sigma(q, et_{i40}))$, $i = 1, ..., m$ without features for feed description and entry description. The first row on the top side of the result tables represents evaluation metrics.

In Table 1, numbers in gray boxs are precision, recall and F measure results of the baseline features defined earlier, and bold numbers in white boxs correspond to precision of $SF(P)$, recall of $SF(R)$ and F measure of $S(FM)$. For $SF(P)$, significant features were constructed from FT without entry, and for $SF(R)$ and $SF(FM)$, significant features were built from FT+FD+ED with 40 entries. From the experimentation results, we found that the significant features outperform the baseline features as Table 1 shows. We present a detail analysis based on feed data element and n with respect to precision and F measure results in the following.

Table 2 indicates that FT achieves the best precision, and precision values of FD, ET, and ED are almost same. Also,

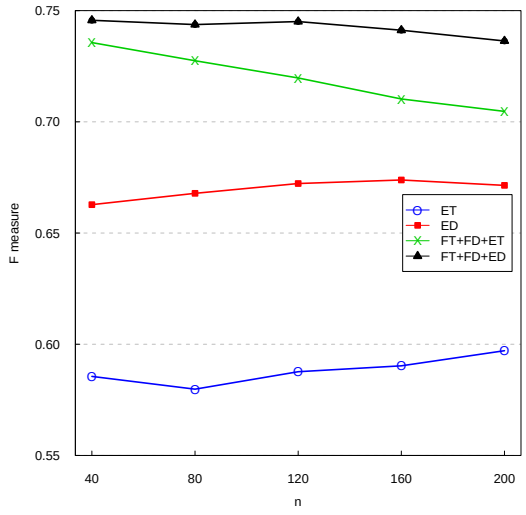

Figure 2: Performance evaluation of feed data elements in various n's

it is remarkable that FT and FD produce higher precision by combining with ET and ED than ET and ED alone. From these observations, we can conclude that FT and FD are more competitive than other feed data elements in terms of the precision performance of feed relevance judgement method.

On the other hand, FT+FD+ED produces the best result for F measure in Table 2. It is interesting to see that FT+FD is more effective than ET and ED for F measure. Furthermore, Table 2 shows that FD and ED with a large number of terms have higher F measure than FT and ET with a small number of terms.

From the fact that FT and FD ahieve low recall and high precision results, it can be concluded that many feeds put their topics into FT and FD, and there is a high probability that terms in FT and FD represent topics of feeds. In addition, FT+FD+ED does not completely outperform FT+FD+ET for F measure, where ED requires high cost for computing features as it has a large number of terms. The difference between F measure of FT+FD+ED and that of FT+FD+ET is only 0.01, suggesting that FT+ FD+ET is more competitive than a combination of all feed data elements, considering the cost for computing feature scores.

Moreover, we examine an effect of n on the performance in Table 1. In general, large n's give better precision results than small n's when entries are used for the feature vector. However, it is interesting to see that precision hardly improves beyond n of 120 as represented in Table 1.

Figure 2 depicts that large n's have better F measure performance than small n's in ET, ED, and ET+ED. On the contrary, small n's yield better performance than large n's for combinations of ET, ED, FT and FD. It turns out that use of FT and FD together with ET and ED is more appropriate to judge relevance of feed than use of ET and ED alone. Accordingly, these results lead us to the conclusion that a large number of entries including old entries are not necessary for constructing the feature vector, but instead that a small number of recent entries are enough to judge relevance of a feed.

Finally, we remark that feed relevance judgement method based on FT+FD+ET and a small number of entries can produce comparable performance to the significant features, $SF(FM)$, while at the same time reducing cost for computing feature scores for entry descriptions.

4. CONCLUSION

In this paper, we investigated the significant features for relevance judgement method in feed retrieval. New feature selection approach for feed relevance judgement was proposed based on the vector space model, considering unique characteristics of feeds.

Extensive experiments were conducted using a data set collected from Google reader. From the experimental results, we found that when a small number of entries are used, feed title, feed description, and entry description become feed data elements for the significant features. In addition, our experimental results show that the relevance judgement method based on feed title, feed descripion and entry title produces similar performance to the identified significant features. It is expected that our research results will contribute to enhancing the retrieval performance of feed search engines.

5. ACKNOWLEDGMENT

This work was supported by the Korea Science and Engineering Foundation (KOSEF) grant funded by the Korean government (MOST) (No. R01-2007-000-11167-0).

6. REFERENCES

[1] C. Chang and C. Lin, *LIBSVM : a library for support vector machines*, [Software]. Available: `http://www.csie.ntu.edu.tw/~cjlin/libsvm`. 2001.

[2] W. B. Croft, D. Metzler, and T. Stronhman, *Search engines: information retrieval in practice*, Addison Wesley, 2010.

[3] H. Drucker, D. Wu, and V. Vapnik, "Support vector machines for spam categorization," *IEEE Trans. Neural Networks*, vol. 10(5), pp. 1048-1054, 1999.

[4] J. L. Elsas et al., "Retrieval and feedback models for blog feed search," *Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, pp. 347-354, 2008.

[5] Google reader, `http://reader.google.com`.

[6] D. Johnson, *RSS and Atom in action: web 2.0 building blocks*, Manning Publications, 2006.

[7] S. Han et al., "Exploring the relationship between keywords and feed elements in blog post search," *World Wide Web*, vol. 12, pp. 381-398, 2009.

[8] Y. H. Li and A. K. Jain, "Classification of text documents," *The Computer Journal*, vol. 41(8), pp. 537-546, 1998.

[9] D. Mladenić et al., "Feature selection using linear classifier weights: interaction with classification models," *Proc. 27nd Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, pp. 234-241, 2004.

[10] L. Zhang, J. Zhu, and T. Yao, "An evaluation of statistical spam filtering techniques," *ACM Trans. Asian Language Information Processing*, vol 3(4), pp. 243-269, 2004.

[11] Z. Zheng et al., "A regression framework for learning ranking functions using relative relevance judgments," *Proc. 30th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR)*, pp. 287-294, 2007.