The OHSUMED Dataset in LETOR

Jun Xu, Tie-Yan Liu, and Hang Li
Microsoft Research Asia
{junxu, tyliu}@microsoft.com

OHSUMED is one dataset available in the LETOR package. This dataset contains features extracted from query-document pairs in the OHSUMED collection, and the corresponding relevance labels. It also includes the evaluation results of several baseline ranking algorithms using the data. In this document, we first introduce the original OHSUMED collection, and then the features. After that, we describe the training, validation and test sets prepared, as well as the baseline experimental results using the data.

1. The Original OHSUMED Collection

The original OHSUMED collection [7] was created for information retrieval research. It is a subset of MEDLINE, a database on medical publications. The collection consists of 348,566 records (out of over 7 million) from 270 medical journals during the period of 1987-1991. The fields of a record include title, abstract, MeSH indexing terms, author, source, and publication type.

There are 106 queries. For each query, there are a number of documents associated. Each query is about a medical search need, and thus is also associated with patient information and topic information. The relevance degrees of documents with respect to the queries are judged by humans, on three levles: *definitely*, *possibly*, or *not relevant*. There are a total of 16,140 query-document pairs with relevance judgments. The results are saved in the file called 'judged'.

(Note: There are five queries for which there are no definitely relevant documents. The five queries are 8, 28, 49, 86, and 93.)

The MEDLINE documents have the same file format as those in the SMART system, with each field defined as below (NLM designator in parentheses):

- *.I* sequential identifier
- *.U* MEDLINE identifier (UI)
- *.M* Human-assigned MeSH terms (MH)
- *.T* Title (TI)
- *.P* Publication type (PT)
- *.W* Abstract (AB)
- *.A* Author (AU)

- *.S* Source (SO)

For each query in the OHSUMED collection, the patient and topic information are defined in the following way:

- *.I* Sequential identifier
- *.B* Patient information
- *.W* Information request

Many research papers [3][4][12] have been published using the original OHSUMED collection. However, since the features and the data partitions used in these papers are different, direct comparisons between the experimental results may not be meaningful. In Lector, we try to adopt the 'standard' features proposed in the IR community and make careful data partitions in the dataset construction. We hope that the created dataset can be widely used in future research on learning to rank. We refer to it as "the OHSUMED Dataset in LETOR".

2. Feature Extraction for the OHSUMED Dataset

We extracted features from each judged query-document pair in the original OHSUMED collection. We index the fields of *.T* and *.W* for documents and the field *.W* for queries. For both documents and queries, the field *.I* is used as id.

For selection of features, we follow the following principle:

(1) to cover all the standard features proposed in IR.
(2) to reproduce the features proposed in publications at SIGIR conferences in recent years that also used OHSUMED in the experiments.

Accordingly, we extracted both 'low-level' and 'high-level' features for the OHSUMED Dataset. Low-level features include term frequency (tf), inverse document frequency (idf), document length (dl) [1], and their combinations. High-level features include the outputs of BM25 [11] and LMIR [13] algorithms. In particular, for LMIR, different smoothing methods (DIR, JM, ABS) [13] were utilized. In total, we extracted 25 features (10 from title, 10 from abstract, and 5 from 'title + abstract'). Note that, when extracting features, we conform to the original documents or papers. If the authors mentioned parameter tuning with regard to the feature, we also conducted tuning based on the whole dataset. If the authors only provide a default parameter and have not mentioned parameter tuning, we will use their default parameter directly in our feature extraction process.

1) Low-level Features

There are 10 low-level features from the fields of title and abstract respectively. As a result, a total of 20 features were extracted. Table 1 shows the details of the features. In the table, we refer to those features proposed in recent SIGIR papers as “Feature in SIGIR paper”.

Table 1. Low-level Features and their descriptions

Features	Formulations	Descriptions	References
L1	$\sum_{q_i \in q \cap d} c(q_i, d)$	Term frequency (tf)	[1]
L2	$\sum_{q_i \in q \cap d} \log(c(q_i, d) + 1)$	Feature in SIGIR paper	[3]
L3	$\sum_{q_i \in q \cap d} \frac{c(q_i,d)}{\lvert d \rvert}$	Normalized tf	[1]
L4	$\sum_{q_i \in q \cap d} \log\left(\frac{c(q_i,d)}{\lvert d \rvert} + 1\right)$	Feature in SIGIR paper	[3]
L5	$\sum_{q_i \in q \cap d} \log\left(\frac{\lvert C \rvert}{df(q_i)}\right)$	Inverse document frequency (idf)	[1]
L6	$\sum_{q_i \in q \cap d} \log\left(\log\left(\frac{\lvert C \rvert}{df(q_i)}\right)\right)$	Feature in SIGIR paper	[3]
L7	$\sum_{q_i \in q \cap d} \log\left(\frac{\lvert C \rvert}{c(q_i,C)} + 1\right)$	Feature in SIGIR paper	[3]
L8	$\sum_{q_i \in q \cap d} \log\left(\frac{c(q_i,d)}{\lvert d \rvert} \log\left(\frac{\lvert C \rvert}{df(q_i)}\right) + 1\right)$	Feature in SIGIR paper	[3]
L9	$\sum_{q_i \in q \cap d} c(q_i, d) \log\left(\frac{\lvert C \rvert}{df(q_i)}\right)$	tf*idf	[1]
L10	$\sum_{q_i \in q \cap d} \log\left(\frac{c(q_i,d)}{\lvert d \rvert} \frac{\lvert C \rvert}{c(q_i,C)} + 1\right)$	Feature in SIGIR paper	[3]

2) High-level Features

We extracted 5 high-level features as follows from the combination of title and abstract.

Table 2. High-level Features and their descriptions

Features	Descriptions	References
H1	BM25 score	[11]
H2	log(BM25 score)	[11]
H3	LMIR with DIR smoothing	[13]
H4	LMIR with JM smoothing	[13]
H5	LMIR with ABS smoothing	[13]

3) List of All Features

The 25 features are listed below, in the same order as they appear in the feature files.

Table 3. All the features for the OHSUMED dataset

Feature ID	Descriptions
1	L1, for the *.T* filed (Title)
2	L2, for the *.T* filed (Title)
3	L3, for the *.T* filed (Title)
4	L4, for the *.T* filed (Title)
5	L5, for the *.T* filed (Title)
6	L6, for the *.T* filed (Title)
7	L7, for the *.T* filed (Title)
8	L8, for the *.T* filed (Title)
9	L9, for the *.T* filed (Title)
10	L10, for the *.T* filed (Title)
11	L1, for the *.W* field (Abstract)
12	L2, for the *.W* field (Abstract)
13	L3, for the *.W* field (Abstract)
14	L4, for the *.W* field (Abstract)
15	L5, for the *.W* field (Abstract)
16	L6, for the *.W* field (Abstract)
17	L7, for the *.W* field (Abstract)
18	L8, for the *.W* field (Abstract)
19	L9, for the *.W* field (Abstract)
20	L10, for the *.W* field (Abstract)
21	H1, for the joint of *.T* (Title) and *.W* fields (Abstract)
22	H2, for the joint of *.T* (Title) and *.W* fields (Abstract)
23	H3, for the joint of *.T* (Title) and *.W* fields (Abstract)
24	H4, for the joint of *.T* (Title) and *.W* fields (Abstract)
25	H5, for the joint of *.T* (Title) and *.W* fields (Abstract)

3. Files in the OHSUMED Dataset

1) File Format

The label of a query-document pair is either 0, 1, or 2, where 0 stands for “not relevant”, 1 for “possibly relevant”, and 2 for “definitely relevant”.

We adopt the format of SVM^{light} (http://svmlight.joachims.org/) input files to store the extracted features. Each line in the file represents a feature vector for a query-document pair, as shown below.

```
<label> <query id>:<value> <feature id>:<value> ... <feature id>:<value> # <info>
```

Where <label> takes values from {0, 1, 2}, <query id> is an integer, <feature id> is as shown in Table 3, <value> is a float value of the corresponding feature, and document id is given at the end of each line as <info>.

An example line is shown below,

```
2 qid:1 1:3.00000000 2:2.07944154 3:0.27272727 …  25:-3.87512000 #docid = 40626
```

It means that for query id “1” and document id “40626”, the label is “2” (definitely relevant). The 25 features extracted for the query-document pair are (3.00000000, 2.07944154, …, -3.87512000).

2) Directory and Dataset Partitioning

The feature file for the whole dataset of OHSUMED is stored in the directory “OHSUMED\Data\All”. Furthermore, we partitioned the whole dataset into five subsets S1, S2, S2, S4 and S5, in order to conduct 5-fold cross validation. For each fold, we used three subsets for training, one subset for validation, and the remaining one for testing. The validation set is used to tune the parameters of ranking algorithms, such as the number of iterations in Neural Network [2], the number of iterations in Boosting, and the combination coefficient in the objective function of Support Vector Machines. In this way, we generated five datasets for cross-validation experiments, in the directories “OHSUMED\Data\Fold1”, “OHSUMED\Data\Fold2”, “OHSUMED\Data\Fold3”, “OHSUMED\Data\Fold4” and “OHSUMED\Data\Fold5” respectively.

Table 4. Data Partitioning for 5-fold Cross Validation

Sub Directories	Trainingset.txt	Validationset.txt	Testset.txt
Fold1	{S1, S2, S3}	S4	S5
Fold2	{S2, S3, S4}	S5	S1
Fold3	{S3, S4, S5}	S1	S2
Fold4	{S4, S5, S1}	S2	S3
Fold5	{S5, S1, S2}	S3	S4

We suggest that the users of the OHSUMED Dataset conduct five-fold cross validation when using the data.

4. Baselines on the OHSUMED Dataset

We ran several state-of-the-art learning to rank algorithms on the OHSUMED Dataset, such as Ranking SVM [6] and RankBoost [5]. When applying an algorithm to the dataset, we conducted query-based normalization for each feature. Suppose there are $N^{(i)}$ documents $\{d_j^{(i)} | j = 1, \ldots, N^{(i)}\}$ with respect to query i in the dataset. A feature of document $d_j^{(i)}$ is represented as $x_j^{(i)}$ ($j = 1, \ldots, N^{(i)}$). Then after normalization, the feature will become $\frac{x_j^{(i)} - \min\{x_k^{(i)}, k=1,\ldots,N^{(i)}\}}{\max\left\{x_k^{(i)}, k=1,\ldots,N^{(i)}\right\} - \min\{x_k^{(i)}, k=1,\ldots,N^{(i)}\}}$.

For Ranking SVM, we use its linear version. For RankBoost, a weak learner is defined on the basis of a single feature, and takes values from {0, 1}. We use the training set to train several ranking models, with respect to different parameters in an algorithm (e.g. the combination coefficient in the objective function of Ranking SVM, and the number of iterations for RankBoost), and then use the validation set to select the best model. After that, we evaluate the best model on the test set as the experimental result. We put the experimental results in the file “OHSUMED\Baselines\ReferenceAlgs_OHSUMED.xls”. In this file, we report both the performance for each of the five folds and the average performance, in terms of NDCG [8][9], Precision, and MAP. Users of the OHSUMED Dataset can take these results as the baselines.

We also evaluated the effectiveness of each single feature when regarding it as a weak ranker. The corresponding results are placed in the file “OHSUMED\Baselines\SingleFeatures_OHSUMED.xls”. These results can help users to understand the features and design their experiments.

6. Additional Note

Users of the OHSUMED Dataset need to sign the “Microsoft Research Shared Source License Agreement (Non-commercial Use Only)” provided at the web site, when they download the dataset. For any question or request regarding to the use of this dataset, please send email to tyliu@microsoft.com.

This document was last updated on March 1st, 2007.

7. References

[1] Baeza-Yates, R. and Ribeiro-Neto, B. Modern Information Retrieval. Addison Wesley, May 1999.

[2] Burges, C.J.C., Shaked, T., Renshaw, E., Lazier, A., Deeds, M., Hamilton, N., Hullender, G. Learning to Rank using Gradient Descent, 22nd International Conference on Machine Learning, Bonn, 2005.

[3] Cao, Y., Xu, J., Liu, T.-Y., Li, H., Huang, Y., and Hon, H.-W. Adapting ranking SVM to document retrieval. Proceedings SIGIR 2006, pp.186–193, 2006.

[4] Cummins, R., O'Riordan, C., An evaluation of evolved term-weighting schemes in information retrieval, Proceedings of CIKM 2005.

[5] Freund, Y., Iyer, R., Schapire, R., and Singer, Y., An efficient boosting algorithm for combining preferences. Journal of Machine Learning Research, 2003 (4).

[6] Herbrich, R., Graepel, T., & Obermayer, K. Large margin rank boundaries for ordinal regression. Advances in Large Margin Classifiers, MIT Press, pp.115-132, 2000.

[7] Hersh, W. R., Buckley, C., Leone, T. J., Hickam, D. H. OHSUMED: An interactive retrieval evaluation and new large test collection for research, Proceedings of SIGIR 1994, pp.192-201, 1994.

[8] Jarvelin, K., and Kekalainen, J. IR evaluation methods for retrieving highly relevant documents. Proceedings of SIGIR 2000, pp.41-48, 2000.

[9] Jarvelin, K., and Kekalainen, J. Cumulated gain-based evaluation of IR techniques, ACM Transactions on Information Systems, 2002.

[10] Robertson, S. E., and Hull, D. A. The TREC-9 filtering track final report. Proceedings of TREC 2000, pp.25–40, 2000.

[11] Robertson, S. E. Overview of the okapi projects, Journal of Documentation, Vol. 53, No. 1, 1997, pp. 3-7.

[12] Xin Yan, Xue Li, and Dawei Song, Document Re-ranking by Generality in Bio-medical Information Retrieval, Proceedings of WISE 2006.

[13] Zhai, C. and Lafferty, J. A study of smoothing methods for language models applied to Ad Hoc information retrieval. Proceedings of SIGIR 2001, pp. 334-342, 2001.