

Overview of the Mailing Lists and Forums Track in FIRE 2010

Ayan Bandyopadhyay
Indian Statistical Institute, Kolkata
ayan_t@isical.ac.in

Debapriyo Majumdar
IBM Research - India
debapriyo@in.ibm.com

ABSTRACT

The mailing lists and forums on technical domains consist of data containing problem statements, suggested solutions or answers to problems and feedback or discussion on those problems and solutions. For a system which tries to find legitimate solutions for a given problem (query), merely matching the topic of a document with the query does not suffice. Identifying the problems, solutions, and whether the solutions actually worked needs to be an integral part of the system as well. In FIRE 2010 we organized a pilot track aiming to create benchmark dataset and relevance judgements for classification and retrieval on mailing lists and forums data and to encourage research in this topic. In this working note we describe our experience and learning while organizing this track.

1. INTRODUCTION

The primary motivation of the mailing lists and forums track is retrieval of answers from data containing technical problems and solutions. Publicly available data of that type are archives of mailing lists and forums, typically consisting of message threads, most of which are started by somebody seeking a solution to a technical problem (s)he faced. Following such questions, other members seek clarifications or more details about the problem, or reply with proposed solutions. The initial poster may explain the problem in more detail in subsequent messages, if required. The other members may help the poster to eventually reach a solution; or the problem may remain unsolved in that thread. Sometimes, the poster may be referred to an earlier thread, where the solution can be found. Occasionally, the discussion digresses into other topics as well. The mailing lists or forums are similar to the trouble ticket data in the services industry, where the clients seek for help or solutions to problems and teams providing technological services solve their problems, record the solutions and close the tickets. Such data also tends to be noisy and identifying if a record contains adequate information about the solution to a problem is a non-trivial challenge.

These aspects of the data from a mailing list or discussion forum make retrieval of the solution (i.e. finding a message, or a set of messages containing a legitimate solution) fairly complex. The objective of this track is to evaluate the effectiveness of retrieval and classification systems on this type of data. It is important to note the difference between a system searching for solutions in data such as above and question answering systems such as Yahoo! answers [1] which are based on direct user ratings.

In 2010, we started this track as a pilot track. We formed

the track proposal around mid-2009 and started our work on data collection after that. Due to the late beginning, and a lot of new challenges that we faced, we got delayed in releasing the data and the queries. We assume that due to this delay the participation in this track has not been too encouraging already, as only one group could finally submit runs to this track, but we are very hopeful that in the next year we will have a much better participation. As organizers of this track we have learned a lot and overall we are positive about the beginning.

2. THE TASKS

The goal of an ideal retrieval system for problem solution data would be to retrieve the posts containing legitimate solutions for a problem represented by a query. That led us to define two tasks, one on retrieval of solutions to a given problem, and another on identifying whether a post describes a problem or suggests a solution (or falls into some other category). For identification of problems and solutions, we defined a classification task which requires a system to classify a given post into a certain category.

2.1 Task 1: Ad-hoc retrieval

Since the complete solution to a problem may be spread across various messages in a thread, we kept it simple for this year's pilot track by defining the unit of retrieval for this task to be a complete thread (instead of one or more selective posts). A thread is regarded as relevant if it contains a "legitimate" or working solution for the problem mentioned in the topic. To make the relevance assessment process easy and consistent, we defined that a thread contains a legitimate solution only if a poster confirms that the proposed solution indeed worked. As a special case, the poster who suggests a solution also can confirm that it worked in case (s)he tried it out himself or herself. The topics of this task will be in the standard TREC / CLEF format, and will describe a technical problem. The retrieval task will be to find a working solution to the problem from the corpus.

2.2 Task 2: Classification of messages

Most of the posts of a mailing list or a forum belong to one (in some cases more than one) of the following categories (MSG-CLASS):

- **ASK_QUESTION:** Asking a question, e.g. somebody posts a problem. This is usually, but not always, the first post of a thread.

- **DITTO**: Repeating a question, e.g. “Yes, I also have the same (or a very similar) problem”.
- **ASK_CLARIFICATION**: Asking for more details about the problem, e.g. “Can you please provide more details? What kind of error message are you getting?”
- **FURTHER_DETAILS**: The person who is facing a problem provides more detailed information about it, possibly after somebody asks for more details.
- **SUGGEST_SOLUTION**: Suggesting a solution
- **SOLUTION_FEEDBACK_NEG**: Somebody tries a suggested solution and says that it did not work for him or her.
- **SOLUTION_FEEDBACK_POS**: Somebody tries a suggested solution that works, and (s)he confirms that it works. Sometimes this may be the legitimate end of a thread.

Note that the same post may belong to more than one of the above categories. For example, when somebody repeats a question, (s)he may also provide more details. The goal of this task is to classify a set of given messages (identified by MSG-ID) into one or more of the above categories.

We provided a list of pre-classified messages as training data. Participants could (in fact, they were encouraged to) build a larger training set on their own. Naturally, they must exclude the messages that are in the test set from their training data. The reason we did not want to fix a training set was that we did not want to assume the users would use only a statistical classification method. A rule based system, or one with a mix of rules and statistical inference could also be of interest.

3. THE DATA

We chose parts of two forums, namely the Tech-forums and the Tech Support Forum, and two mailing list archives, namely the Ubuntu archive and the Tug India archive to build our data collection, as described below.

Data source: Tech forums
 Sub forum: Windows Operating Systems and Software
 URL: <http://www.tech-forums.net/pc/f9>

Data source: Tech support forum
 Sub forum: Mozilla Firefox Browsers
 URL: <http://www.techsupportforum.com/alternative-computing/mozilla-firefox-browsers>

Data source: Ubuntu mailing list
 URL: <https://lists.ubuntu.com/archives/ubuntu-users>
 Timeline: September 01, 2004 to June 30, 2009

Data source: Tug India archive
 URL: <http://www.tug.org/pipermail/tugindia>
 Timeline: May 01, 2001 to June 30, 2009

The forum sub-collections are organized into files, each of which corresponds to a complete discussion thread. In contrast, each file in the mailing list sub-collections corresponds to an individual email message. We cleaned the data and converted them into the following format.

Markup for Ubuntu-users archives and Tug India archives:

```
<DOC> : Starting tag of a document.
<DOCNO> </DOCNO> : Contains document identifier.
<BODY> </BODY> : Contains message body.
</DOC> : Ending tag of a document.
```

Markup for Tech-forums and Tech support forums:

```
<DOC> : Starting tag of a document.
<DOCNO> </DOCNO> : Contains document identifier.
<TITLE> </TITLE> : Contains document title.
<MSG> : Starting tag of a message.
<MSGID> </MSGID> : Contains message identifier.
<TIMESTAMP> </TIMESTAMP> : Contains time stamp
of a message.
<POSTER> </POSTER> : Contains the name
of the poster.
<BODY> </BODY> : Contains message body.
</MSG> : Ending tag of a message.
</DOC> : Ending tag of a document.
```

Each file in the collection has a unique <DOCNO> field. The <THREAD-ID> for a document from the forum sub-collections are the same as its <DOCNO>. We reconstructed the thread structure for the mailing list sub-collections using the message ids in the “In-reply-to” and “References” fields of the messages of the mailing lists and supplied a separate file which provided a mapping between the <DOCNO> and <THREAD-ID> of each post. The <DOCNO> of the first post of a thread is taken as the <THREAD-ID> of the thread.

The files were encoded with UTF-8 encoding. The total size of the corpus was 954MB and it contained 212,132 documents.

4. THE QUERIES AND TEST SET

For the ad-hoc retrieval task we created 25 queries manually. Each query is presented in the standard TREC format with a title, a description and a narrative field, as shown below.

```
<top>
<num>2</num>
<title>Merging pdf / postscript files</title>
<desc>
How can one merge two or more pdf
and/or postscript files into a single pdf
or postscript file?
</desc>
<narr>
A relevant thread should contain instructions
or commands for merging more than one pdf or
postscript files into a single pdf or postscript file.
The method should also be able to handle a
combination of pdf and postscript files.
</narr>
</top>
```

Some of the queries were about seeking a solution to a particular problem (for example, how to merge multiple pdf files), some were about seeking knowledge (for example, what are the differences between Ubuntu and other linux distributions). The participants were asked to submit their

runs in the usual TREC / CLEF submission format with each submission file containing up to 1000 threads per topic, ranked 0-999. Each line in the file were required to have the following fields:

<Query id> Q0 <THREAD-ID> <RANK> <SIMILARITY> <Run-ID>

For the classification task we provided a small set of 206 labeled messages as examples of the message classes. The test set contained 295 messages. The labels to both the training set and the test set were assigned manually by us. The posts to label were chosen randomly, but the ones which looked ambiguous were discarded.

Message class	Examples	Test
ASK_CLARIFICATION	22	22
ASK_QUESTION	45	65
DITTO	8	13
FURTHER_DETAILS	25	32
SOLUTION_FEEDBACK_NEG	18	21
SOLUTION_FEEDBACK_POS	21	27
SUGGEST_SOLUTION	90	124

Figure 1: The labeled data

For this task each line of a submission file is required to have the following fields:

<MSG-ID> <MSG-CLASS> <CONFIDENCE-SCORE (optional)>

The field CONFIDENCE-SCORE is optional and if present, must be a value in the range 0 to 1. If a confidence score is not present, then a default value 1 is assumed. When one message is classified into multiple message classes, then there will be multiple rows for the same MSG-ID.

5. SUBMITTED RUNS AND DISCUSSION

Unfortunately only one group (Ohio State - IBM Research India) submitted runs to the mailing lists and forums track this year. We attribute this lack of participation primarily to our delay in releasing the dataset and queries was a key factor. The Ohio State - IBM Research group has submitted seven runs to the classification task and two runs to the retrieval task.

5.1 Challenges in the retrieval task

Although we received two runs to the retrieval task, the submission got significantly delayed, leaving us only about one week to evaluate that. In practice the pooling approach works reasonably well to create relevance judgement set, specially when there are some runs which are manual in nature [3]. However, in our case we received only two runs and moreover for this task, a topic match does not qualify for relevance, presence of a legitimate solution is required. A standard information retrieval technique is likely to match the query with threads describing problems in similar topic, but posts containing the solution may or may not match the keywords (or other features) present in the query.

Due to this complex nature of the problem, and lack of many submissions we have decided to create a relevance judgement set for the queries manually. Since that would take a lot of effort, we have not been able to finish that yet, and hence the evaluation of the two runs submitted to the

retrieval tasks are not complete yet. We will update the relevance judgement set on the wiki and also a discussion on the evaluation on the website at a later stage.

5.2 Evaluation of the classification task

The Ohio State - IBM Research group primarily used two approaches for their seven runs submitted to the classification task. Five of their runs submitted were based on sequence labeling using Conditional Random Field (CRF) with one CRF per class, due to small amount of training data, as they explained in their working notes [2]. The five runs (namely R1, R2, R3, R4 and R7) are all based on the seven CRF model, varying the choice of features used. The other two runs (R5 and R6) are based on statistical classifiers built on each class to perform binary classification of each post as a single document. For R5 the participants discarded DITTO because it is a rare class and for R6 all classes were considered.

After evaluating the classification runs using our manually created labels on the test set, it turned out that R2 and R4 performed marginally better than the other runs overall, according to the F-measure. For a problem solution system, we view classification of problems and solutions to be the main focus, so identifying the messages belonging to the classes ASK_QUESTION, SUGGEST_SOLUTION and SOLUTION_FEEDBACK_POS can be thought of as most important. It is interesting to observe that all the runs based on sequence labeling performed significantly better than R5 and R6 for the messages belonging to ASK_QUESTION class. In fact, for this particular class most of these runs had precision above 90% and recall about 80%, hence the F-measure above 80%. This phenomenon is probably because mostly problems are asked in the first message of a thread.

On the other hand, for classes such as SUGGEST_SOLUTION and SOLUTION_FEEDBACK_POS the runs R5 and R6 performed better than the sequence labeling approach, in particular R6 turned out to be the best of all¹.

6. LIMITATIONS OF THE DATA SET

We have discovered some formatting problems with the dataset after releasing it as one of the participants pointed it out. Although we provided the labeled set of messages as example and mentioned that participants were free to label more data to use as training set, or use some rule based approach, the only group which submitted runs used only our small set of labels as their training set. In future we would revisit this scenario and may work on providing more labels. Also, it was a very challenging task to create ad-hoc queries and we expect that there would be some issues with some queries that will be clearer during our manual work of creating relevance judgement set.

7. CONCLUSION AND FUTURE WORK

Although the pilot track lacked participation this year, it has been a positive start and we gained some valuable insights while creating the dataset and the queries. We will

¹The participants reported that they left out 65 messages from the test set because their program encountered errors trying to parse the data. Since they were the only participating group and the reason for discarding those messages had no relation with their approaches to the problems, we computed the precision, recall and F-measures for their runs considering only 230 messages that they considered.

produce a refined and mature dataset, possibly with more content and organize our tasks better in the coming year. Releasing some ad-hoc queries and relevance judgement as training set would be a better idea too, that will give the participants a clearer picture of the problem. With this note we thank the participants, the overall coordinators and look forward to continue the work in the next year.

Acknowledgement: We thank Dr. Mandar Mitra and Dr. Prasenjit Majumdar, the overall coordinators of FIRE 2010, for their continuous support and guidance. We also thank the project trainees at Indian Statistical Institute who helped us to create the queries, labeled data and test set.

8. REFERENCES

- [1] Yahoo! answers: <http://answers.yahoo.com>.
- [2] P. Raghavan, R. Catherine., S. Ikbal, and N. Kambhatla. Classification and retrieval from mailing lists and forums. In *FIRE Working notes*, 2010.
- [3] J. Zobel. How reliable are the results of large-scale information retrieval experiments? In *SIGIR*, 1998.