COMP30018 Knowledge Technologies
Project 2 Description

11 September, 2012

Version 1.0

1 Basic Task Description

The basic task is to build various "add-ons" and explore their impact on a monolingual IR system, in the context of web user forum search. The project is intended to help you gain familiarity with some of the methods we have discussed in lectures in the IR section of the subject.

You will be provided with a selection of end-to-end IR systems, a fixed document collection, and a set of queries. Your job is to come up with a series of add-ons for the basic IR system, and fine-tune and evaluate each over the query set. You will also be required to write up your experiences in the form of a written report (similar in style to the report for Project 1), to be submitted at the same time as the results of the different runs.

The "documents" are forum threads from `ancestry.com`, e.g.:

```
<DOC>
<DOCNO>thread7772</DOCNO>
<TEXT>
[post_id]: 00004A
[post_title]: Aren't all Coulombe of common descent?
[post_text]: My mother-in-law received a letter from someone ...

[post_id]: 00004A0021
[post_title]: all Coulombe in North America are related.
[post_text]: Yes, we are all related to one couple married in 1670 ...

[post_id]: 00004A0021002Q
[post_title]: Henriette Coulombe, Richmond, Qc
[post_text]: Hello, I am searching for a Henriette Coulombe who ...

[post_id]: 00004A0021002R
[post_title]: Henriette Coulombe
[post_text]: Hello, I am searching for a Henriette Coulombe who ...

[post_id]: 00004A0021002S
[post_title]: Godefroi Coulombe
[post_text]: My great-great-grandfather was Godefroi Coulombe.  ...

</TEXT>
</DOC>
```

The post structure is included within each document (`[post_id]:`, etc.), but is treated as just another token by the IR engines you are provided with.

Queries are based on real-world queries to the forum the threads were crawled from, e.g.:

```
<top>
<num>1</num>
<title>julius burow; germany</title>
</top>
```

You will be provided with three IR engines to (optionally) use as the basis of your IR systems, namely:

- **ZETTAIR** (`/home/subjects/327/local/project2/zettair/` on the CSSE student machines): ZETTAIR is an open-source IR engine which is being co-developed at RMIT and the University of Melbourne.
- **LEMUR** (`/home/subjects/327/local/project2/lemur{,-src}/` on the CSSE student machines): LEMUR is an open-source IR engine which is being co-developed at the University of Massachusetts, Amherst and Carnegie Mellon University.
- **SPYDR.PY** : SPYDR.PY (provided in the project tarball) is a very basic, but hopefully highly configurable, IR engine written by Tim in Python.

All three systems provide a ranked document output for a given query, and are based on the vector space model and a TF-IDF-style term weighing function. While you will certainly learn a lot from reading through the source code of the first two systems, you are very definitely **not** expected to hack the source code in order to implement your add-ons. Note also that you **may not** bombard the ZETTAIR or LEMUR mailing lists with requests/questions relating to the system: all questions relating to the two systems should be addressed to Tim or Jeremy, preferably via the discussion forum on the LMS or alternatively via email (in which case we will almost certainly repost the original question and our response to the LMS anyway). The third search engine was written with transparency and extensibility in mind, and is the recommended engine if you wish to experiment with "kernel-level" ideas such as term weighting or expansion methods.

Note that you are welcome to implement your own IR engine should you so wish, in which case you should carry out all your experiments using your own engine (but the number of "add-ons" you need to create is unchanged).

The main interest in the project is twofold: (1) what "add-ons" you come up with (and hopefully how they improve the retrieval effectiveness of the base IR engine(s)); and (2) how the add-ons interact with each other. In the case of 3 add-ons, e.g., you will be required to experiment with the add-ons individually ($\times 3$), in pairs ($\times 3$), and all together ($\times 1$), for a total of 8 runs (including the base system without add-ons). A large part of the project will be making sense of the performance of the various runs. Note that if you are experimenting with 5 add-ons, this means you will have a total of 32 different runs, meaning that while you should present full results (over the training queries) for all runs, you won't have space in the report to describe each combination of add-ons in full detail; rather, you should identify the overall trends and distill the key findings in the prose.

2 Evaluation

System evaluation will be based mean average precision (MAP) calculated over the top-100 results for each query. The MAP calculation will be based on "pairwise preference", using the method of Carterette and Bennett (2008).

3 Terms of Use

As part of the terms of use of the dataset, in using the data you agree to the following:

1. The Information may be used for academic research of computer algorithms (the "Research").
2. Summaries, analyses and interpretations of the Information data properties relevant to the Research may be derived and published in academic journals or other media primarily of an academic nature, provided it is not possible to reconstruct the Information from these summaries.
3. Small excerpts of the Information may be displayed to others or published in a scientific or technical context, solely for the purpose of describing the research and development carried out and related issues, and only in academic journals or other media primarily of an academic nature.

4. Publication of Research based on the Information in academic journals or other media primarily of an academic nature should provide attribution to Ancestry.com as the source of the Information.
5. All efforts must be made not to a) publish any personal identifiable information of any individual, or (b) infringe the rights of any third party including, but limited to, the authors and publishers of any excerpts used in accordance with clause 3.
6. Neither Information nor the fact that Ancestry.com provided the Information may be published, distributed or displayed in other media (e.g., press releases and other non-academic publications) without Ancestry.com's prior written consent.

Note that the document collection is a sub-sample of original dataset described in Elsas (2011). As such, published results over the original dataset will be compared to those you will obtain.

4 Data Files

All necessary data files to carry out the project are contained in the following tarball, accessible from the student machines:

```
/home/subjects/327/local/project2/train-nodocs.tgz
```

This contains the following files:

- `comp30018-proj2/bin/batch-trec-zettair`: a bash script to use in indexing the document set, running a method over a set of queries, etc. when using **ZETTAIR** . The script takes the following command line options:

`-c`	clean away all ZETTAIR index files
`-p`	generate the ZETTAIR index
`-t`	run ZETTAIR over the test queries
`-i <INDEX>`	change the ZETTAIR index name to `INDEX`
`-n <N-docs>`	output the top-`N` documents
`-r <RUN-NAME>`	change the name of the current run to `RUN-NAME`
`-d <DIRNAME>`	change the location of the document set to `DIRNAME`

- `comp30018-proj2/bin/batch-trec-lemur`: a bash script to use in indexing the document set, running a method over a set of queries, etc. when using **LEMUR** . The script takes the following command line options:

`-c`	clean away all LEMUR index and temp files
`-p`	generate the LEMUR index
`-t`	run LEMUR over the test queries
`-n <N-docs>`	output the top-`N` documents
`-r <RUN-NAME>`	change the name of the current run to `RUN-NAME`
`-d <DIRNAME>`	change the location of the document set to `DIRNAME`

- `comp30018-proj2/bin/batch-trec-spydr`: a bash script to use in indexing the document set, running a method over a set of queries, etc. when using **SPYDR** . The script takes the following command line options:

`-t`	run SPYDR over the test queries
`-n <N-docs>`	output the top-`N` documents
`-r <RUN-NAME>`	change the name of the current run to `RUN-NAME`
`-d <DIRNAME>`	change the location of the document set to `DIRNAME`

- `433327-proj2/eval/lemur-params`: files for parameterising LEMUR
- `433327-proj2/topics/Topics.trec`: the set of queries

I have also placed a (read-only) copy of the documents in:

```
/home/subjects/327/local/project2/docset/
```

that you can use directly, to save disk space when working on CSSE machines, and also a tarball *with* all the documents:

```
/home/subjects/327/local/project2/train-docs.tgz
```

5 Use of External Documents

You are welcome to use external documents in your different add-ons, but make sure to clearly indicate any such use.

6 Submission

The final submission will consist of three basic parts:

1. the output of the ($\geq 7\times$ for individuals and $\geq 15\times$ for teams) different runs of your method over: (a) the training queries, and (b) the test queries **in TREC format** (see below)
2. the code used to run the different runs and pre-/post-process the document/query data, plus a shell script or makefile to generate all the different output files using your code (via the single command `make`); the code and script/makefile should be in the form of a gzipped tarball
3. a written report (see below)

The results of your different runs should be submitted in TREC format, with the output of each run over either the training or test queries contained in a single file (named `GROUPNAME_run-id.{train,test}`, e.g. `s327g100_run2.train`). **No more than 10** results should be returned for any one query, and at least one document should be returned for each query. Each line of the file represents one result, with fields as follows:

```
topic-id  Q0  document-id  rank  score  run-id
```

each of which should be separated by whitespace, and where:

- `topic-id` is the topic identifier (e.g. `001`)
- `Q0` is the query number within the topic. This field is currently unused but must be provided and must have the value `Q0` (the letter Q followed by the number zero)
- `document-id` is the official document identifier of the retrieved document (e.g. `350543`)
- `rank` is the rank at which the document was retrieved for this topic, where the document most likely to be relevant has rank 1
- `score` is the score of that document. The score must be in descending order. The evaluation routines score systems based on the scores, not the ranks. If you want the precise ranking you submit to be evaluated, the score field must reflect that ranking (i.e., break any ties numerically in the `score` values)
- `run-id` uniquely identifies each group and submitted run. It should be made up of the concatenation of your group name (e.g. `tim`), a hyphen (`-`) and the run number (e.g. `01`). Note, the value should not contain any whitespace or colons (`:`), and each submitted run should be uniquely identified.

The first few lines of a submission file might look something like this:

```
1       Q0      THREAD2213      1       11.630734       tim-run01
1       Q0      THREAD6428      2       9.341386        tim-run01
1       Q0      THREAD6692      3       9.167642        tim-run01
1       Q0      THREAD6572      4       7.648722        tim-run01
1       Q0      THREAD1188      5       7.292292        tim-run01
```

Note that the format allows for an arbitrary amount of whitespace between each field.

Your systems can be implemented in any programming language (or combination of programming languages). As with Project 1, you are welcome to run your code on any machine, but support for getting the various search engines running is only provided for the CSSE servers. Remember to submit a makefile or single-file script which runs the various combinations of add-ons both the training and test query sets (in the local directory) and outputs the results to an appropriately-named file (one per add-on combination per query set).

As for Project 1, submission will take place automatically in the form of us copying out the contents of your project directory at the time of the submission deadline.

7 Report

The report should provide a basic description of:

1. the task
2. the different aspects of the task you have focused on
3. the technical details of all you have implemented
4. evaluation of the different system configurations over the training queries

Note that I am more interested in seeing evidence of you having thought about the task and determined reasons for the relative performance of different methods, than I am in the raw scores of the different methods you select. This is not to say that you should ignore the relative performance of different runs over the data (and, indeed, a bonus mark will be awarded for the best-performing method over the test topics), but rather that you should think beyond simple numbers.

As a requirement of the project, you should use the LaTeXor RTF style file provided for Project 1.

Reports are to be submitted in the form of a single PDF file. If a report is submitted in any format other than PDF, I reserve the right to return the report with a mark of 0.

As there will not be a peer review component for project 2, please include your name and student number in the header of the report (despite what the style file says).

8 Assessment

As with Project 1, the number of distinct systems you need to develop differs according to whether your are an undergraduate or postgraduate student, and whether you are participating as an individual or a team of two, as follows:

Undergrad/postgrad	Team size	Distinct add-ons required	Report length
Undergraduate	1	3–5	2,000–2,500 words
	2	4–5	3,000–3,500 words
Postgraduate	1	4–5	3,000–3,500 words
	2	5	4,000–4,500 words

If you wish to form a team, each member needs to send email to Jeremy (`jeremymn@csse.unimelb.edu.au`) by 5:00pm 14 September, 2012 stating the name and CSSE login of your partner. Note that the default

assumption is that you will participate in the same configuration (single-person or team) as for Project 1, and that if we don't hear from you explicitly, you will be allocated to the same group as for Project 1.

The project will be marked out of 15, and is worth 15% of your overall mark for the subject. Note that there is a hurdle requirement on your combined project mark, of 15/30 (of which this project will contribute 15 marks).

The mark breakdown for the project will be:

Ranking of your best-performing system	3 marks
Creativity	4 marks
Critical Analysis	5 marks
Report clarity	3 marks
TOTAL	15 marks

For details, see the project marking sheet (which is the same as for Project 1, reweighted proportional to the respective mark allocations).

I will not accept late submissions under any circumstances. If your project work is disrupted because of medical or personal reasons, you will be required to email Tim (`tb@ldwin.net`) details, including documentation of the length of that period. Your submission will then be marked according to the proportion of the project period your productivity was disrupted for (irrespective of the relative timing of the disruption). No requests for special consideration on medical or personal grounds will be accepted after the submission deadline.

Note that computer systems are often heavily loaded near project deadlines, and unexpected network or system downtime can occur. You should plan ahead to avoid leaving things to the last minute, when unexpected problems may occur. Generally, system downtime or failure will not be considered as grounds for an extension.

While it is acceptable to discuss the project with other teams in general terms, excessive collaboration is considered cheating. I will be vetting system submissions for originality and will invoke the University's Academic Misconduct policy (`http://academichonesty.unimelb.edu.au/policy.html`) where either inappropriate levels of collaboration or plagiarism are deemed to have taken place.

9 Changes/Updates to the Project Specifications

I will use the LMS to advertise any (hopefully small-scale) changes or clarifications in the project specifications. Any addendums made to the project specifications via the LMS will supersede information contained in the hard-copy version of the project.

10 Important Dates

Release of queries and document set	11 September, 2012
Deadline for team changes	14 September, 2012 (5:00pm)
Deadline for submission of results	22 October, 2012 (3:00pm)
Deadline for submission of code and written report	22 October, 2012 (3:00pm)

References

Ben Carterette and Paul N. Bennett. Evaluation measures for preference judgments. In *Proceedings of 31st International ACM-SIGIR Conference on Research and Development in Information Retrieval (SIGIR'08)*, pages 685–686, Singapore, 2008.

Jonathan L. Elsas. Ancestry.com online forum test collection. Technical report, Carnegie Mellon University, 2011.