

Statistical Language Modeling for Information Access

Practical 1: Installing and Indexing

Maarten de Rijke Edgar Meij Krisztian Balog

University of Amsterdam
Norwegian University of Science and Technology

August 1–4, 2011

Preliminaries

- Originally, Lemur and Indri were separate beings (both are converging rapidly)
 - indexes can be used interchangeably
 - some parameter (and file!) names still differ
- Slide 21: `buildindex` should be `IndriBuildIndex`
- Slide 25: Lemur 4.7 on Windows uses `PATH_TO_LEMUR/GUI`
- Slide 25: `PATH_TO_LEMUR/swig/src/java/LemurRet.jar` should be `PATH_TO_LEMUR/share/lemur/LemurRet.jar`
- Slide: 25: Add:
 - Mac OS X: it looks for the library in directories in your `DYLD_LIBRARY_PATH` environment variable

Outline of the Course

Practical

- Day 1: Installing and Indexing
- Day 2: Retrieval and Evaluation
- Day 3: Retrieval Parameters and Indri
- Day 4: Pseudo Relevance Feedback and Some More Evaluation;
Additional bells, whistles and requests

Outline

① Installing

Downloading

Building from source

② Indexing

Document types

Parameters

Building an index

Browsing an index

Downloading

<http://staff.science.uva.nl/~mdr/Teaching/Cordoba2011/>

- Binaries
 - Mac OS X
 - Windows
- Source
 - Linux
 - Windows (requires Visual Studio)

Outline

① Installing

Downloading

Building from source

② Indexing

Document types

Parameters

Building an index

Browsing an index

Building from source

- base set of applications will be compiled and placed in `/usr/local/lemur/bin` by default
- use `--prefix=` to change this location
- to install additional components, add the following flags to your `./configure` and rebuild:
 - Java: `--enable-java` and `--with-javahome=JAVA_HOME`
 - summarization components: `--enable-summarization`
 - document clustering components: `--enable-cluster`
 - distributed retrieval components (optional): `--enable-distrib`
 - php (optional): `--enable-php`
 - C# (optional): `--enable-csharp`

Paths paths paths

- Convenient to set the PATH environment variable to the right location
 - `export PATH=$PATH:PATH_TO_LEMUR/bin`
 - Control Panel → System → Advanced → Environment Variables

Outline

① Installing

Downloading

Building from source

② Indexing

Document types

Parameters

Building an index

Browsing an index

Document types

- The Lemur Toolkit can index a wide variety of documents
- Most common types of documents are:
 - TREC text
 - TREC web
 - XML
 - standard web (HTML) documents
- If you have a set of plain text documents that you wish to index, one of the easiest ways to prepare the documents is to use a script to iterate through the documents adding TREC text tags (and DOC and unique DOCNOs)

TREC Text

- Most common: plain text using TREC formatting
- Must have:
 - a <DOC> tagset surrounding the document
 - a <DOCNO> tagset enclosing the document ID
 - a <TEXT> tagset enclosing the text to be indexed
- May have multiple documents per file

- Example:

```
<DOC>
```

```
<DOCNO>document_id</DOCNO>
```

```
<TEXT>
```

```
Index this document text.
```

```
</TEXT>
```

```
</DOC>
```

TREC Web format

- Similar to TREC text with the caveat that the main body text contains HTML formatted text
- May also contain an optional <DOCHDR> tagset at the beginning that holds the header information from the HTTP request.
 - Original URL
 - Date and server from when the page text was gathered
 - Other information

XML/HTML

- Standard, well-formed XML/HTML pages
- Needs no pre-processing before indexing
- Only one document per file
- Absolute document location will be used as DOCNO
- Optional link processing (HTML only)

Other document types

- Additionally, the Lemur toolkit can handle the following other document types:
 - mbox** Unix mailbox files
 - doc** Microsoft Word documents (Windows only, requires Microsoft Office)
 - ppt** Microsoft PowerPoint documents (Windows only, requires Microsoft Office)
 - pdf** Adobe PDF
 - txt** Text documents

Outline

① Installing

Downloading

Building from source

② Indexing

Document types

Parameters

Building an index

Browsing an index

Parameters

- After your documents are prepared, you should create a parameter file that will be used to guide the indexer
 - where your source documents are
 - where to place the index
 - and more...

Example

```
<parameters>
<corpus>
<path>/path/to/text/files/</path>
<class>trectext</class>
</corpus>
<memory>256m</memory>
<index>/path/to/your/index</index>
</parameters>
```

Example

```
<corpus>  
<path>/path/to/text/files/</path>  
<class>trectext</class>  
</corpus>
```

- path** Defines where to find your source files. If this path is a directory, it will tell the indexer to index all files in the directory.
- class** Defines what type of documents the source documents in this path are (the example above uses trectext.) If the `<class>` parameter is left out, the indexer will attempt to parse the files based on their file extension, skipping over any files that it does not know how to process.

Example

```
<memory>256m</memory>
```

```
<index>/path/to/your/index</index>
```

memory Is a “soft-limit” of the amount of memory the indexer should use.

index Tells the indexer where to place the built index

Outline

① Installing

Downloading

Building from source

② Indexing

Document types

Parameters

Building an index

Browsing an index

CLEF 2006 ADHOC task

- Cross-Lingual Evaluation Forum
- 2006 adhoc: newspaper articles
 - Glasgow Herald
 - LA Times

Building an index

- 1 Download the collection from `http://staff.science.uva.nl/~mdr/Teaching/Cordoba2011/`
- 2 Look at the source documents from both sets (latimes and GH95). Which fields will not be indexed when we use the TREC Text format?
- 3 Create a new parameter file
- 4 Run `./bin/IndriBuildIndex param_file`

Outline

① Installing

Downloading

Building from source

② Indexing

Document types

Parameters

Building an index

Browsing an index

Browsing an index

- To quickly see what has happened after indexing, use `dumpindex(.exe)`
 - inverted lists
 - document representations
 - vocabulary
 - statistics
- Questions
 - 1 How many documents are there in this collection?
 - 2 What is the IDF of the term “Cordoba”?

Browsing an index — GUI

- Lemur comes with (Java) GUI's, both for retrieval and indexing
- First you will need to add the Lemur library to your Java path:
 - Linux (bash): `export`
`LD_LIBRARY_PATH=$LD_LIBRARY_PATH:PATH_TO_LEMUR/lib`
 - Mac OS X: `export`
`DYLD_LIBRARY_PATH=$LD_LIBRARY_PATH:PATH_TO_LEMUR/lib`
 - Windows: Control Panel → System → Advanced → Environment Variables. Add `PATH_TO_LEMUR/GUI` to the System variables.
- To browse an index:
 - Linux/Mac OS X: `java -jar`
`PATH_TO_LEMUR/share/lemur/LemurRet.jar`
 - Windows: `java -jar PATH_TO_LEMUR/GUI/LemurRet.jar`

Browsing an index — GUI

- If you get the error “`java.lang.UnsatisfiedLinkError: no lemur_jni in java.library.path`”, it means that the GUI cannot find the Lemur library, which should be in the `PATH_TO_LEMUR/lib/` or `PATH_TO_LEMUR/GUI/` directory. Fix:
 - Windows: java looks for the shared library in the current directory and in directories specified in your `PATH` environment variable
 - Linux: it looks for the library in directories in your `LD_LIBRARY_PATH` environment variable
 - Mac OS X: it looks for the library in directories in your `DYLD_LIBRARY_PATH` environment variable