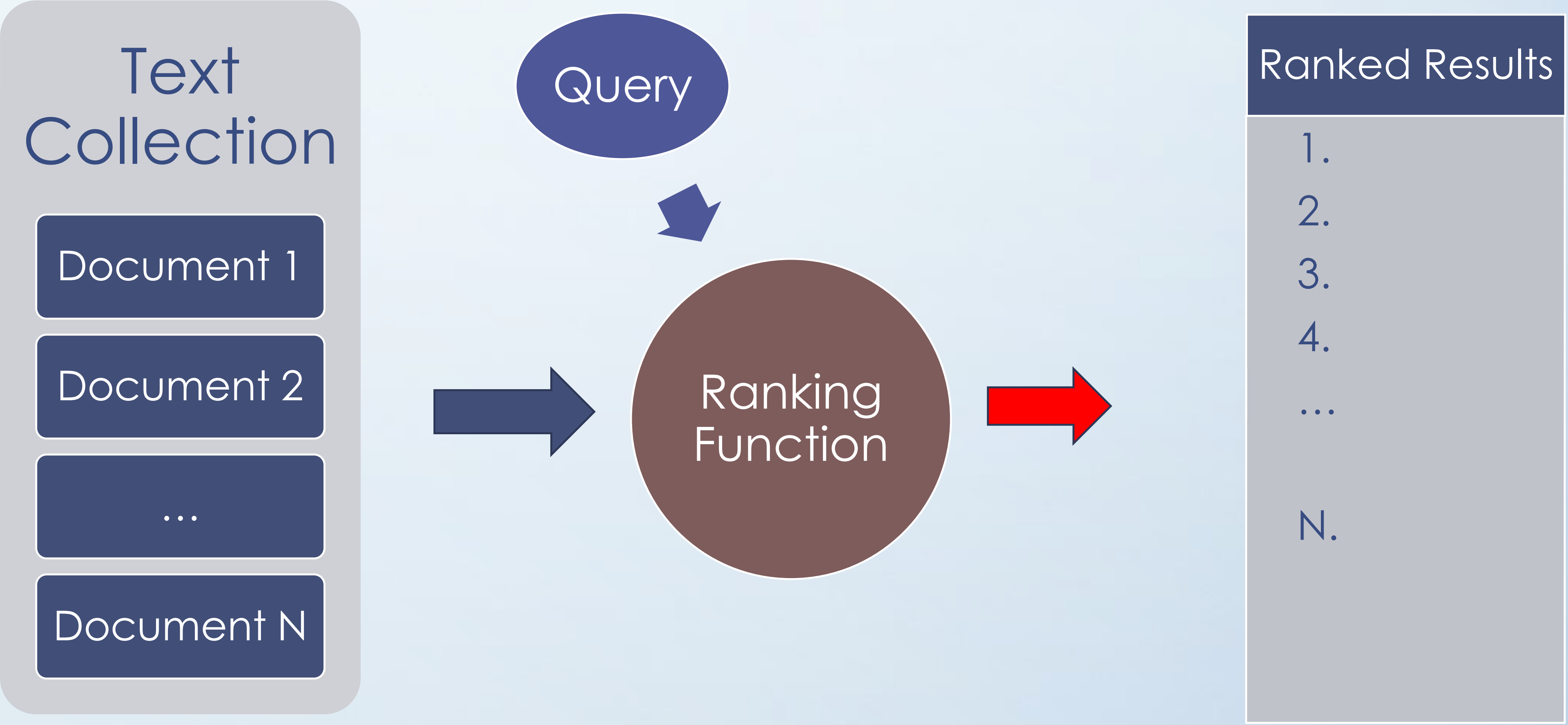
Ranking Segmented documents using Data Fusion

by Hamed Rezanejad

Outline

- Description of the problem
- Motivation/Importance
- Methodology
- Experimental results
- Demo
- Conclusion/future work

Description



Text Collection

Document 1

Document 2

…

Document N

Query

Ranking Function

Ranked Results

1.

2.

3.

4.

…

N.

Description

- **Order** of retrieved documents is very important
- Generally, **Size** of documents differs compare to each other.
- Each document has **different segments** discussing **different issues**
- Using these segments can help us to have **better order** of retrieved documents

Motivation/Importance

- Passage Retrieval
  - ✓ Unit of retrieval is blocks of text from the stored document

    - ➢ Current IR systems are used for indexing a great **variety of documents.**
    - ➢ For **big size documents**, standard ranking is not of value.
    - ➢ Tracking topics in **information feeds**, is a case that standard ranking has nothing to do.

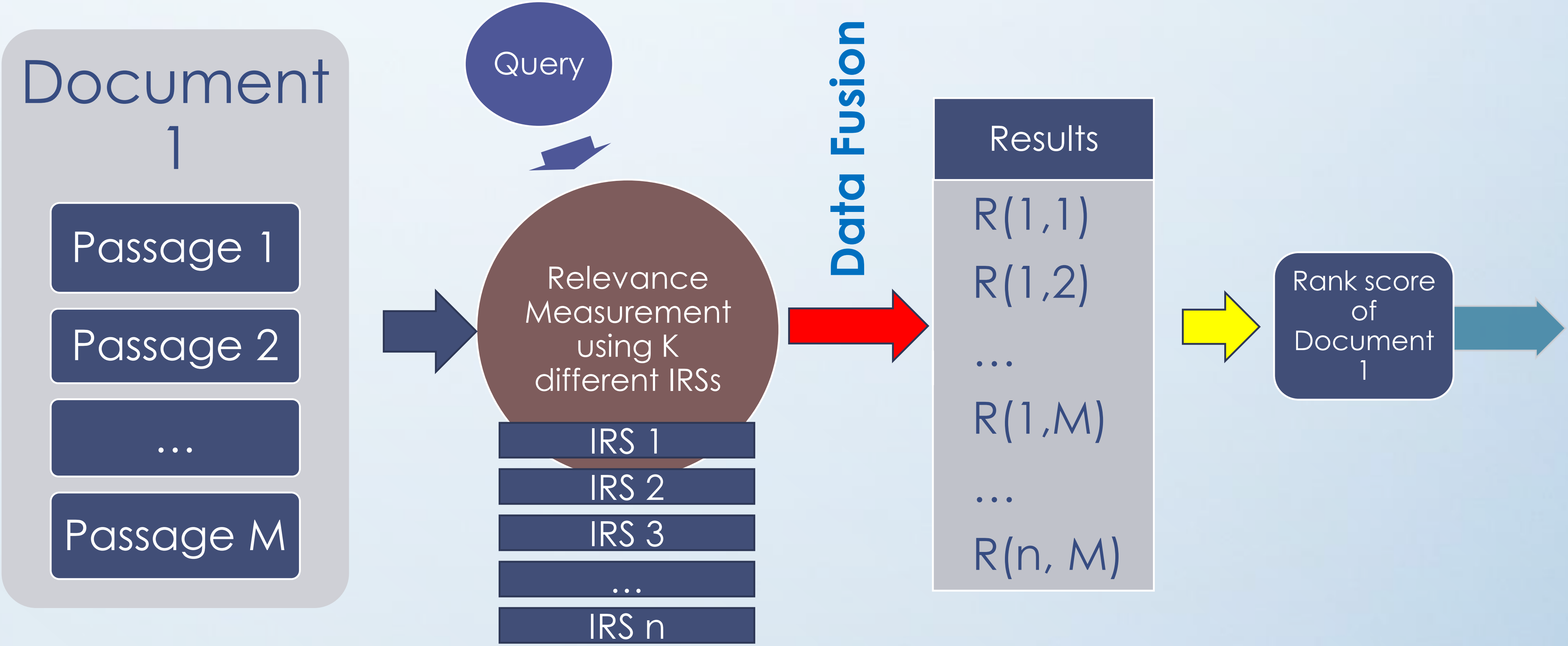
Motivation/Importance

- Data Fusion
  - ✓ Accepts two or more ranked lists and **merges** these lists into a single ranked list

Aim of data fusion:

1. Providing **a better effectiveness** than all systems used for data fusion.
2. Grouping existing search services under **one umbrella**.

Methodology



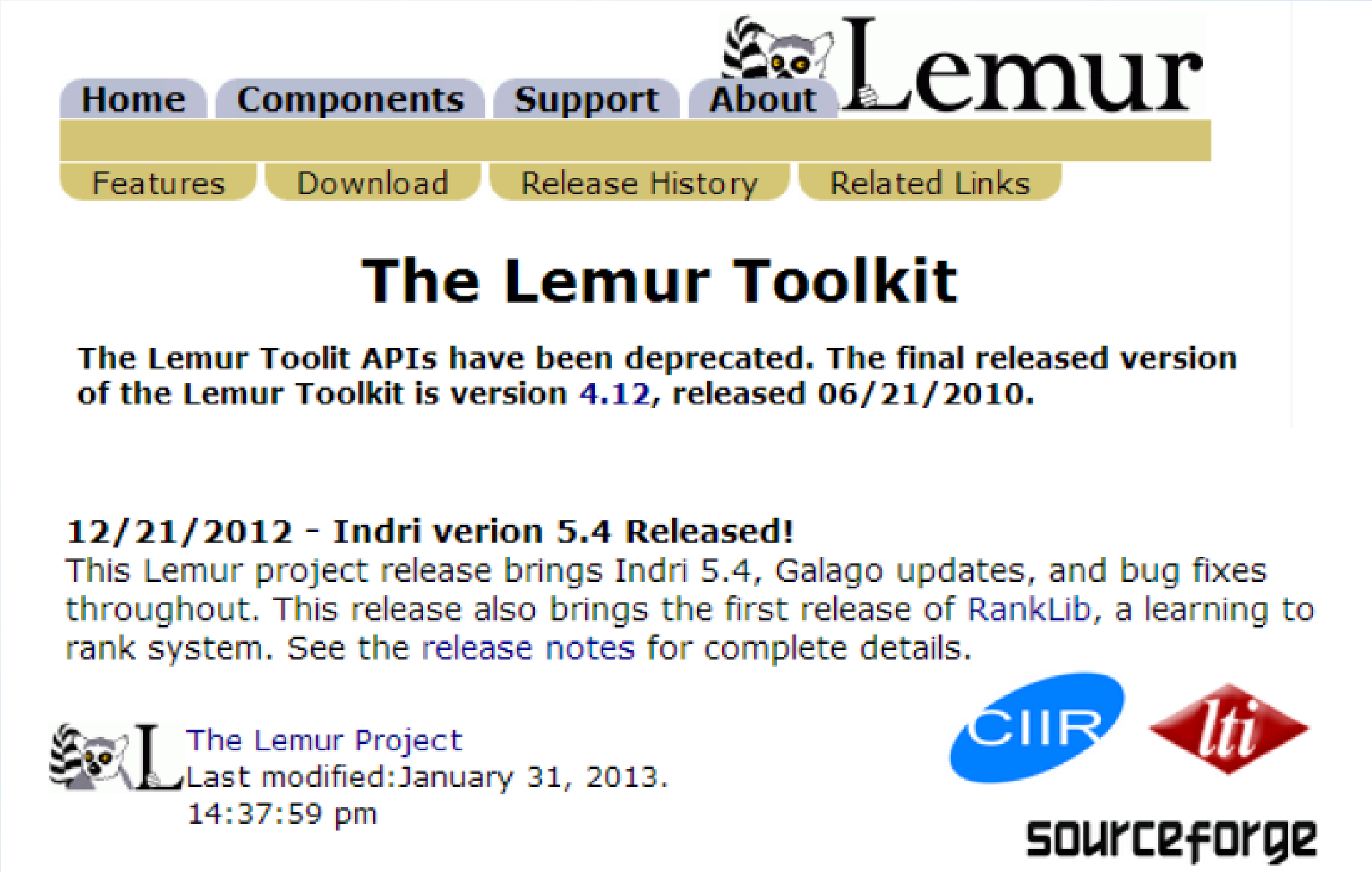
Methodology

Document	# Passages	Ranks of passages	Final rank
1	2	1, 3	1.58
2	3	2, 6, 7	4.033
3	2	9, 10	6.49
4	4	4, 5, 8, 11	5.39

$$\text{Final Rank} = \frac{\sum \log(rank)}{\log(\#passages)}$$

Experimental Results

- I have used **Indri** from **Lemur Project**
- The project's first product was the Lemur Toolkit, a collection of software tools and search engines designed to support research on using statistical language models for information retrieval tasks.
- Later the project added the Indri search engine for large-scale search
- I have used **TREC vol. 4** as dataset.



The Lemur Toolkit

The Lemur Toolit APIs have been deprecated. The final released version of the Lemur Toolkit is version 4.12, released 06/21/2010.

12/21/2012 - Indri verion 5.4 Released!
This Lemur project release brings Indri 5.4, Galago updates, and bug fixes throughout. This release also brings the first release of RankLib, a learning to rank system. See the release notes for complete details.

The Lemur Project
Last modified:January 31, 2013.
14:37:59 pm

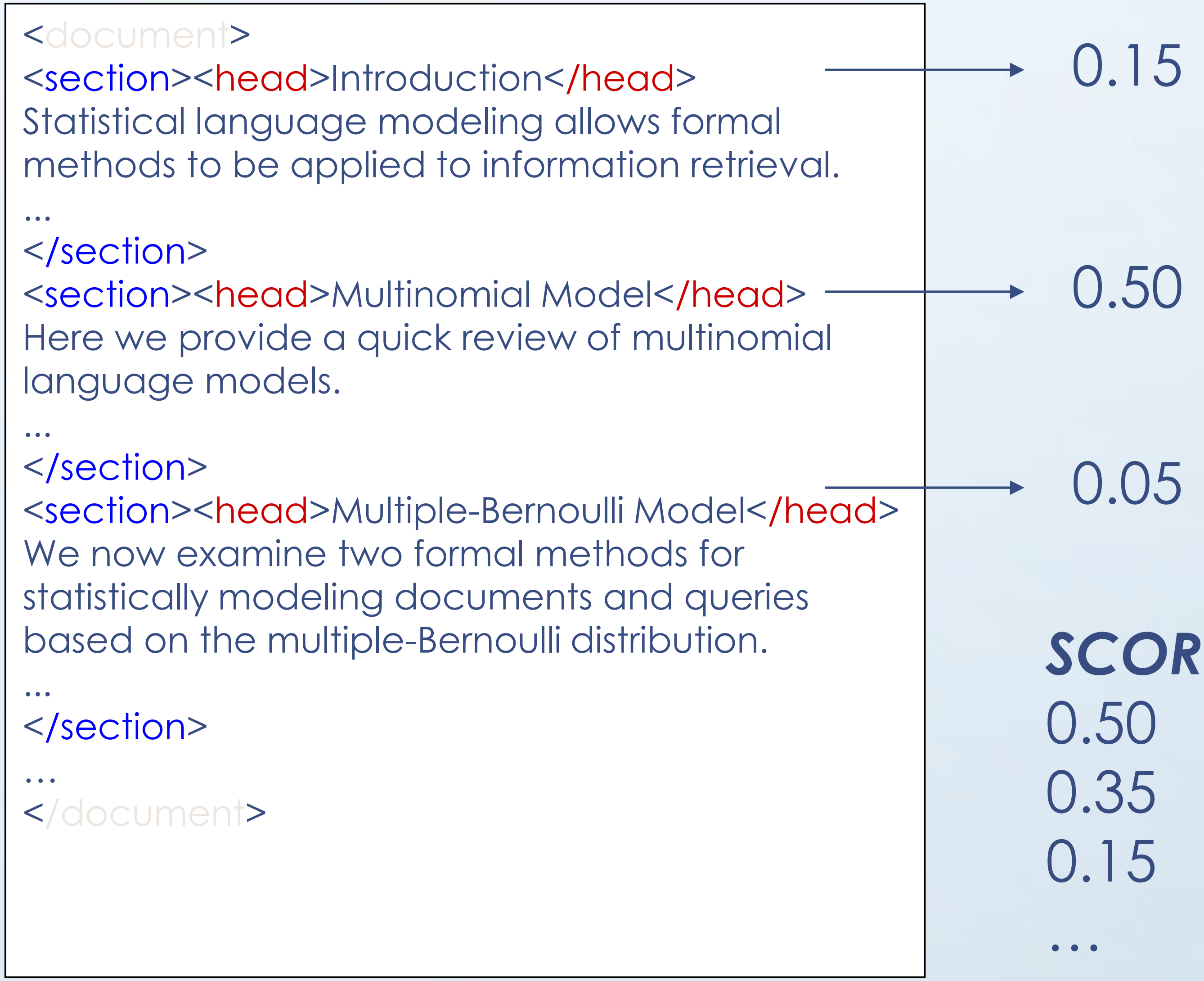
Experimental Results

- Indri provides the **QueryEnvironment** and **IndexEnvrionment** classes, which can be used from C++, Java, C# or PHP

- **QueryEnvironment** allows you to run queries and retrieve a ranked list of results.

- **IndexEnvironment** understands many different file types.
  - TREC formatted documents, HTML documents, text documents, and PDF files , …

Demo & Future Works



```
<document>
<section><head>Introduction</head>
Statistical language modeling allows formal
methods to be applied to information retrieval.
…
</section>
<section><head>Multinomial Model</head>
Here we provide a quick review of multinomial
language models.
…
</section>
<section><head>Multiple-Bernoulli Model</head>
We now examine two formal methods for
statistically modeling documents and queries
based on the multiple-Bernoulli distribution.
…
</section>
…
</document>
```

0.15

0.50

0.05

1. Treat each *section extent* as a "document"
2. Score each "document" according to query
3. Return a ranked list of *extents.*

SCORE	***DOCID***	***BEGIN***	***END***
0.50	IR-352	51	205
0.35	IR-352	405	548
0.15	IR-352	0	50
…	…	…	…