

LAB 4 – Language Models for IR

Objective

In this lab, you will implement several **probabilistic unigram language models for Information Retrieval**. Each model estimates the probability distribution of terms in a document and computes the likelihood of a query using **Maximum Likelihood Estimation (MLE)** and various **smoothing techniques**.

The models to implement are as following:

1. **Unsmoothed Language Model (MLE)**
2. **Add-1 (Laplace) Smoothing**
3. **Good–Turing Smoothing** (Take the decision to use normal or approximated version)
4. **Jelinek–Mercer Smoothing ($\lambda = 0.4$)**.
5. **Dirichlet Smoothing**

The goal of this lab is to understand how different models and smoothing techniques represent documents as term distributions.

1. Prerequisites

Before starting this lab, ensure that you have successfully completed **Lab 1**, where you:

- Indexed the document collection **D₁–D₆**.
- Tokenized the text using the specified regular expression.
- Removed stop words and applied **Porter stemming**.

2. Implementation

For each document, you will build a **unigram Language Model** and compute the likelihood of a query using Maximum Likelihood Estimation and smoothings presented in course.

Test Queries

Use the following queries to test your implementation:

- q1: large language models for information retrieval and ranking**
q2: LLM for information retrieval and Ranking
q3: query Reformulation in information retrieval

q4: ranking Documents

q5: Optimizing recommendation systems with LLMs by leveraging item metadata

Expected Steps

- 1- Build vocabulary V for each document and for the Collection
- 2- Compute term frequencies for each document $tf(w,d)$ according to the course formula about Language models for IR
- 3- Compute collection frequencies for each document $cf(w,C)$ according to the course formula about Language models for IR
- 4- Compute similarity between the queries and all documents, using the formulas given in the lecture notes for each model.
- 5- Rank the Documents for each query in decreasing order.

Dirichlet μ Parameter Selection

To determine the best value of μ for a Dirichlet-smoothed language model, follow the bellow approach:

1- Compute the average document length

Let N_d be the number of words in document (d). The average document length across the collection is calculated as:

$$N_{avg} = \frac{1}{|C|} \sum_{d \in C} N_d$$

where $|C|$ = total number of documents in our collection.

2- Set the Dirichlet smoothing parameter

Set μ by replacing in the following formula:

$$\mu \approx 0.3 \times N_{avg}$$

The idea here is to ensures that the smoothing strength is proportional to the typical document size in the collection, balancing between relying on the document's own statistics and the collection statistics.