## Your Name: Wei Wang

**Your Andrew ID: ww5**

**Homework 2**

# Collaboration and Originality

Your report must include answers to the following questions:

1. Did you receive help of any kind from anyone in developing your software for this assignment (Yes or No)? It is not necessary to describe discussions with the instructor or TAs.

If you answered Yes, provide the name(s) of anyone who provided help, and describe the type of help that you received.

No

1. Did you give help of any kind to anyone in developing their software for this assignment (Yes or No)?

If you answered Yes, provide the name(s) of anyone that you helped, and describe the type of help that you provided.

No

1. Are you the author of every line of source code submitted for this assignment (Yes or No)? It is not necessary to mention software provided by the instructor.

Yes

If you answered No:

* 1. identify the software that you did not write,
  2. explain where it came from, and
  3. explain why you used it.

1. Are you the author of every word of your report (Yes or No)?

Yes

If you answered No:

* 1. identify the text that you did not write,
  2. explain where it came from, and
  3. explain why you used it.

## Your Name: Wei Wang

**Your Andrew ID: ww5**

**Homework 2**

# Experiment 1: Baselines

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Ranked**  **Boolean** | **BM25**  **BOW** | **Indri**  **BOW** |
| **P@10** | 0.0400 | 0.3700 | 0.4900 |
| **P@20** | 0.0800 | 0.3550 | 0.4250 |
| **P@30** | 0.0867 | 0.3400 | 0.3833 |
| **MAP** | 0.0079 | 0.0614 | 0.0973 |

# Experiment 2: BM25 Parameter Adjustment

## k1

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **k1** | | | | | | | |
| 1.2 | 1.5 | 2 | 0 | 4 | 1 | 0.5 | 12 |
| **P@10** | 0.3700 | 0.3700 | 0.3600 | 0.2000 | 0.3300 | 0.3800 | 0.4500 | 0.2600 |
| **P@20** | 0.3550 | 0.3500 | 0.3300 | 0.2250 | 0.3000 | 0.3650 | 0.4100 | 0.2150 |
| **P@30** | 0.3400 | 0.3267 | 0.3067 | 0.2367 | 0.3000 | 0.3367 | 0.3700 | 0.1933 |
| **MAP** | 0.0614 | 0.0598 | 0.0572 | 0.0540 | 0.0463 | 0.0620 | 0.0676 | 0.0291 |

.

## b

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **b** | | | | | | | |
| 0.75 | 0 | 0.1 | 0.3 | 0.4 | 0.5 | 0.8 | 0.9 |
| **P@10** | 0.3700 | 0.3500 | 0.4800 | 0.5000 | 0.5000 | 0.4800 | 0.3500 | 0.2700 |
| **P@20** | 0.3550 | 0.3800 | 0.4400 | 0.4500 | 0.4350 | 0.4150 | 0.3350 | 0.2900 |
| **P@30** | 0.3400 | 0.3567 | 0.4167 | 0.4233 | 0.4000 | 0.3800 | 0.3233 | 0.3000 |
| **MAP** | 0.0614 | 0.0865 | 0.1063 | 0.0945 | 0.0858 | 0.0800 | 0.0583 | 0.0520 |

## Parameters

During the process of choosing k1, b, the parameters shows the following tendency. When either k1 or b equal to 0, the precision is quite low comparing to other k1 and b values. Since the range of k1 >=0, b>0 and b<1, so I test the marginal value of both k1 and b to see how the result goes. Also the default value of k1 and b is 1.2 and 0.75, I made sure that the tried parameter is distributed around the standard value. So I can compare the lower and higher value around baseline to see its impact.

For k1, I choose from 0 to 12, and evenly distributed. For b, I choose 0 to 1 and evenly distributed.

## Discussion

When k1 is 0, the document frequency has no effect. The query term frequency will dominate the impact. As a result, the document with longer length will rank high. The repeated term in the query will also dominate the result. Therefore, the precision looks relatively low. Similarly, the larger the k1 is, the smaller the impact of documents term frequency to the result (the term frequency weight can be smaller). In the experiment result, the precision is relatively high at k=0.5, this might be caused by given a proper term frequency weight. Also, when we want to put attention on those rare words, we need to keep k1 in a relatively low value, since when k1 approaches 0, rare words and repeated query terms will dominate.

When b is 0, document length is being ignored. As a result, the documents term frequency will impact the result. Therefore, the long document with bigger vocabulary will be more likely to be retrieved. My best result of b is around 0.1-0.4, which is lower than 0.75. This is because lower b ignore the documents length, which is good, documents length sometimes will provide not effective information. Once b is larger than 0.5, the precision looks decreasing again.

# Experiment 3: Indri Parameter Adjustment

## µ

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **µ** | | | | | | | |
| 2500 | 2000 | 1000 | 3000 | 4000 | 0 | 2400 | 2600 |
| **P@10** | 0.4900 | 0.4600 | 0.4400 | 0.4700 | 0.4700 | 0.2700 | 0.4800 | 0.4800 |
| **P@20** | 0.4250 | 0.4300 | 0.4100 | 0.4400 | 0.4350 | 0.2450 | 0.4150 | 0.4250 |
| **P@30** | 0.3833 | 0.3933 | 0.3767 | 0.3933 | 0.4000 | 0.2600 | 0.3900 | 0.3833 |
| **MAP** | 0.0973 | 0.0936 | 0.0796 | 0.0962 | 0.0973 | 0.0492 | 0.0967 | 0.0968 |

## 𝜆

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **𝜆** | | | | | | | |
| 0.4 | 0 | 0.1 | 0.2 | 0.3 | 0.5 | 0.8 | 1 |
| **P@10** | 0.4900 | 0.4900 | 0.4800 | 0.4800 | 0.4800 | 0.4600 | 0.3700 | 0.0000 |
| **P@20** | 0.4250 | 0.4200 | 0.4100 | 0.4250 | 0.4250 | 0.4100 | 0.3350 | 0.0000 |
| **P@30** | 0.3833 | 0.4100 | 0.4067 | 0.3967 | 0.3967 | 0.3833 | 0.3333 | 0.0033 |
| **MAP** | 0.0973 | 0.0994 | 0.0997 | 0.0990 | 0.0982 | 0.0945 | 0.0765 | 0.0002 |

## Parameters

For lambda, when it reaches 1, not good, when lambda = 0, result looks fine since the map value is low, so I keep pushing lambda to a relatively small value that approaches 0;

For mu, when it is 0, the result is not good. When the value floating around 2500, it looks like either direction is decreasing precisions, the best values are around 2500 and 2600.

## Discussion

For mu, I think it is more important to short document. Since mu will help to smoothing parameters, for a short document, the smoothing can have greater impact on longer one.

For lambda, if the query is short, the lambda should be small, if the query is long, lambda should be large. In our case, the query is the relatively short, so the smaller lambda is performing well, since the smaller the lambda is, the more weight is putting on the document term frequency.

# Experiment 4: Different representations

## Example Query

#AND ( #WSUM(0.1 indiana.url 0.1 indiana.title 0.2 indiana.keywords 0.7 indiana.body) #WSUM(0.1 child.url 0.1 child.title 0.2 child.keywords 0.7 child.body) #WSUM(0.1 support.url 0.1 support.title 0.2 support.keywords 0.7 support.body))

## Results for the Query Set

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Indri**  **BOW**  **(body)** | **0.20 url**  **0.00 keywords**  **0.30 title**  **0.50 body** | **0.10 url**  **0.20 keywords**  **0.10 title**  **0.60 body** | **0.00 url**  **0.10 keywords**  **0.00 title**  **0.90 body** | **0.00 url**  **0.00 keywords**  **0.00 title**  **0.10 body** | **0.10 url**  **0.00 keywords**  **0.10 title**  **0.80 body** |
| **P@10** | 0.4900 | 0.4700 | 0.4600 | 0.4700 | 0.4900 | 0.4600 |
| **P@20** | 0.4250 | 0.4150 | 0.4100 | 0.4200 | 0.4250 | 0.4200 |
| **P@30** | 0.3833 | 0.3700 | 0.3833 | 0.3833 | 0.3833 | 0.3833 |
| **MAP** | 0.0973 | 0.0890 | 0.0936 | 0.0967 | 0.0973 | 0.0965 |

## Weights

At first I distribute the weight among different field evenly, but putting 0.5 at body field, since mostly the words are appear in the body. Also the title is playing some important roles. Later when adjusting weights, I found that the higher the body weight is, the higher the map and p@10 is, therefore in the following process I keep increasing the weight of body. After putting all weights on body, the map reach same affect as IndriBOW. So I choose to give some weights to url and title. They are both important words, the result turns out to be good.

## Discussion

The p@30 is stay at a stable status, around 0.3833, what varies more is p@10 and p@20, as well as the map. The query has a good performance when adding most of the weights onto body. Also, when other fields are giving weights, the map can be improve relatively.

# Experiment 5: Sequential dependency models

## Example Query

#wand(0.6 #and ( indiana child support ) 0.1 #and ( #near/1( child support ) #near/1( indiana child ) ) 0.3 #and ( #window/8( child support ) #window/8( indiana child ) ) )

## Results for the Query Set

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Indri**  **BOW**  **(body)** | **0.5 AND**  **0.25 NEAR**  **0.25 WINDOW** | **0.60 AND**  **0.10 NEAR**  **0.30 WINDOW** | **0.70 AND**  **0.00 NEAR**  **0.30 WINDOW** | **0.80 AND**  **0.00 NEAR**  **0.20 WINDOW** | **0.90 AND**  **0.00 NEAR**  **0.10 WINDOW** |
| **P@10** | 0.4900 | 0.5000 | 0.5000 | 0.5100 | 0.5000 | 0.5000 |
| **P@20** | 0.4250 | 0.4150 | 0.4200 | 0.4400 | 0.4100 | 0.4200 |
| **P@30** | 0.3833 | 0.4000 | 0.4033 | 0.4267 | 0.4067 | 0.4167 |
| **MAP** | 0.0973 | 0.0920 | 0.0968 | 0.1017 | 0.1002 | 0.0980 |

## Weights

First given relatively even distributed weight to three operators, put 0.5 on AND, and I found that as the weight of AND goes up, the precision becomes quite good, and the map is also increasing, so I increase the weight of AND. Also, when AND has most of the weight, by increasing the WINDOW weight, the result will be better too. So at last two columns the results are both quite good, when AND has more than 0.8 of weight.

## Discussion

The reason why precision of adding too much weight to WINDOW or NEAR is not high is because they both constraint the appearance of terms within the document. Sometimes this constraints might lost some retrieval of documents. The AND operator can calculate smoothing scores, and at the same time don’t have a certain constraints on the terms distances. Therefore, AND with more weight can have a better result.