|  |  |  |  |
| --- | --- | --- | --- |
| **Factors/parameters** | **Experiment #1** | **Experiment #2** | **Experiment #3** |
| Preferred block size (*SB*) | 50, 250 (a typical page), 1000 | 50, 250 (a typical page), 1000 | 50, 250 (a typical page), 1000 |
| Maximum number of blocks (*BM*) | 50, 100 | 50, 100 | 50, 100 |
| False positive rate (*RFP*) |  |  |  |
| Number of query terms (*QT*) | 1, 2, N(5, 2) | 1, 2, N(5, 2) | 2, N(5, 2) |
| Number of words per query term (*QW*) | 1, 2, N(5, 3) | 1, 2, N(5, 3) | 2, N(5, 3) |
| Number of documents (N) | 1000, 10000 | 1000, 10000 | 1000, 10000 |
| Text corpus | {Small Documents, Large Documents } | {Small Documents, Large Documents } | {Small Documents, Large Documents } |
| Stop words | Yes, No | Yes, No | Yes, No |
| Stemming | Yes, No | Yes, No | Yes, No |
| Proximity Scoring | No | No | Yes |
| Term Weighting | No | Yes | Yes |

**Table 3** – summary of the four experiments

NOTES:

(a)

This will consist of 4224 experiments. This may be too many – in which case I will remove some of them, e.g., remove stopwords and stemming params (I would always just do both), which would get me down to 1056 experiments. If this is too many, I can of course give one or more parameters fewer choices.

(b)

Text corpus implicitly defines document size (*SD*). A text corpus will just be a set of documents. I will have the following two text corpuses:

1. A text corpus of small documents. This will not be as demanding on the proximity measure, as the document can be divided into a smaller number of blocks of size *SB*. Of course, proximity scoring may still be slow if
2. A text corpus of large documents. These will be very demanding on the proximity scoring.

I will sample N={1000, 10000} from the designated text corpus to make a database, repeats allowed.

(c)

N(5, 2) in *QT* and *QW* is a truncated normal distribution, with a minimum value of 1 and a maximum value of 8. A maximum of 8 seems reasonable as this will allow exact phrases up to 8 words long as query terms. I will construct M queries (M is just some constant that does not change) according to *QT* and *QW*.

When *QT* is 1 no proximity scoring needs to be used, so in experiment #3 I do not explore *QT*=1.

Let me know if this seems reasonable.