Web Information Retrieval

Assignment 6

Team Name : Gamma

Members

Name	Matriculation Number
Mohammad Nizam Uddin	**216101140**
Md. Shohel Ahamad	**216203438**
MD Jakaria Nawaz	**216203442**
Shreya Chatterjee	**216100848**

1. Binary Independence Model :

1.1 Comparison to TF-IDF:

The Binary Independence Model (BIM) is the model that has traditionally been practiced with the PRP. It proposes some easy theories, which makes determining the probability function P(R|d, q) practical.
Documents and queries are both designated as binary term incidence vectors. That is, a document "d" is represented by the vector ~x = (x1, . . . , xM) where xt = 1 if term "t" is present in document "d" and "xt" = 0 if "t" is not present in "d".
Similarly, q represents by the incidence vector ~q (the distinction between q and ~q is less central since commonly q is in the form of a set of words).
Under the BIM, we model the probability P(R|d, q) that a document is relevant via the probability in terms of term incidence vectors P(R|~x,~q). Then, using Bayes rule, we have:

$$P(R=1|\vec{x},\vec{q}) = \frac{P(\vec{x}|R=1,\vec{q})P(R=1|\vec{q})}{P(\vec{x}|\vec{q})}$$

$$P(R=0|\vec{x},\vec{q}) = \frac{P(\vec{x}|R=0,\vec{q})P(R=0|\vec{q})}{P(\vec{x}|\vec{q})}$$

Here, P(~x|R = 1,~q) and P(~x|R = 0,~q) are the probability that if a relevant or non-relevant, respectively, the document is retrieved, then that document's representation is ~x. (*The Binary Independence Model. 2017*)

TF-IDF vector space model Gerard Salton and his colleagues suggested a model based on Luhn's similarity the criterion that has a stronger theoretical motivation. They considered the index representations and the

query as vectors embedded in a high dimensional Euclidean space, where each term is assigned a separate dimension. The similarity measure is usually the cosine of the angle that separates the two vectors ~d and ~q. The cosine of an angle is 0 if the vectors are orthogonal in the multidimensional space and 1 if the angle is 0 degrees. The cosine formula is given by:

$$\text{score}(\vec{d}, \vec{q}) = \frac{\sum_{k=1}^{m} d_k \cdot q_k}{\sqrt{\sum_{k=1}^{m} (d_k)^2} \cdot \sqrt{\sum_{k=1}^{m} (q_k)^2}}$$

The metaphor of angles between vectors in a multidimensional space makes it easy to explain the implications of the model to non-experts. *(Djoerd Hiemstra's home page » Publications. 2017).*

1.2 Document Relevance :

Under the assumption that relevant documents are a very small percentage of the collection, it is plausible to approximate statistics for non relevant documents by statistics from the whole collection. Under this assumption, ut (the probability of term occurrence in non relevant documents for a query) is dft/n and

$$\log[(1 - u_t)/u_t] = \log[(N - \text{df}_t)/\text{df}_t] \approx \log N/\text{df}_t$$

In other words, we can provide a theoretical justification for the most frequently used form of idf weighting. *(Probability estimates in practice. 2017)*

2 - Web Search Characteristics

2.1 Shingling

Doc A.1: “a bump on the log in the hole in the bottom of the sea”
Doc A.2: “a frog on the bump on the log in the hole in the bottom of the sea”
Doc B.1: “your mother drives you in the car”
Doc B.2: “in mother russia car drives you”

For n=1 , Shingles -
Doc A.1_Shingles -> a, bump, on, the, log, in, the, log, in, the, hole, in, the, bottom, of, the, sea
set(Doc A.1_Shingles) -> {a, bump, on, the, log, in, hole, bottom, of, sea }

Doc A.2_Shingles -> a, frog, on, the, bump, on, the, log, in, the, hole, in, the, bottom, of, the, sea
set(Doc A.2_Shingles) -> { a, frog, on, the, bump, log, in, hole, bottom, of, sea }

Doc B.1_Shingles -> your, mother, drives, you, in, the, car
set(Doc B.1_Shingles) -> { your, mother, drives, you, in, the, car }

Doc B.2_Shingles -> in, mother, russia, car, drives, you
set(Doc B.2_Shingles) -> { in, mother, russia, car, drives, you }

Jaccard coefficients J(Doc A.1, Doc A.2) = 10/11 = 0.909
Jaccard coefficients J(Doc B.1, Doc B.2) =⅝ = 0.625

For n=2, Set of Shingles -
Doc A.1_Shingles -> a-bump, bump-on, on-the, the-log, log-in, in-the, the-log, log-in, in-the, the-hole, hole-in, in-the, the-bottom, bottom-of, of-the, the-sea
set(Doc A.1_Shingles) -> { a-bump, bump-on, on-the, the-log, log-in, the-hole, hole-in, in-the, the-bottom, bottom-of, of-the, the-sea }

Doc A.2_Shingles -> a-frog, frog-on, on-the, the-bump, bump-on, on-the, the-log, log-in, in-the, the-hole, hole-in, in-the, the-bottom, bottom-of, of-the, the-sea
set(Doc A.2_Shingles) -> { a-frog, frog-on, on-the, the-bump, bump-on, the-log, log-in, in-the, the-hole, hole-in, the-bottom, bottom-of, of-the, the-sea }

Doc B.1_Shingles -> your-mother, mother-drives, drives-you, you-in, in-the, the-car

set(Doc B.1_Shingles) -> { your-mother, mother-drives, drives-you, you-in, in-the, the-car }

Doc B.2_Shingles -> in-mother, mother-russia, russia-car, car-drives, drives-you
set(Doc B.2_Shingles) -> { in-mother, mother-russia, russia-car, car-drives, drives-you }

Jaccard coefficients J(Doc A.1, Doc A.2) = 11/15 = 0.73
Jaccard coefficients J(Doc B.1, Doc B.2) = 1/10 = 0.1

For n=3, Set of Shingles -
Doc A.1_Shingles -> a-bump-on, bump-on-the, on-the-log, the-log-in, log-in-the, in-the-hole, the-hole-in, hole-in-the, in-the-bottom, the-bottom-of, bottom-of-the, of-the-sea
set(Doc A.1_Shingles) -> { a-bump-on, bump-on-the, on-the-log, the-log-in, log-in-the, in-the-hole, the-hole-in, hole-in-the, in-the-bottom, the-bottom-of, bottom-of-the, of-the-sea }

Doc A.2_Shingles -> a-frog-on, frog-on-the, on-the-bump, the-bump-on, bump-on-the, on-the-log, the-log-in, log-in-the, in-the-hole, the-hole-in, hole-in-the, in-the-bottom, the-bottom-of, bottom-of-the, of-the-sea
set(Doc A.2_Shingles) -> { a-frog-on, frog-on-the, on-the-bump, the-bump-on, bump-on-the, on-the-log, the-log-in, log-in-the, in-the-hole, the-hole-in, hole-in-the, in-the-bottom, the-bottom-of, bottom-of-the, of-the-sea }

Doc B.1_Shingles -> your-mother-drives, mother-drives-you, drives-you-in, you-in-the, in-the-car
set(Doc B.1_Shingles) -> { your-mother-drives, mother-drives-you, drives-you-in, you-in-the, in-the-car }

Doc B.2_Shingles -> in-mother-russia, mother-russia-car, russia-car-drives, car-drives-you
set(Doc B.2_Shingles) -> { in-mother-russia, mother-russia-car, russia-car-drives, car-drives-you }

Jaccard coefficients J(Doc A.1, Doc A.2) = 11/16 = 0.6875
Jaccard coefficients J(Doc B.1, Doc B.2) = 0/9 =0

2.2 Implementation

In shingles.py file.

2.3 Hashing for Duplicate Detection

There are so many duplicate content. If some content is exactly duplicate including images, syntax style etc. then it can be detected using Hashing but for near duplicate contents only hashing is not possible.
Like there are two similar documents one with images and another without images this can't be detected with hashing. Also, for near duplicate if different synonyms and metaphors are used to tell the same thing it will also not be possible to detect with hashing. There is an example given from wikipedia about Michael Jackson that is near duplicate but can't be detect using hassing.
Slide 37 at lecture06-websearchchar.pdf

2.4 Choice of Prior

References :

Djoerd Hiemstra's home page » Publications. 2017. *Djoerd Hiemstra's home page » Publications*. [ONLINE] Available at: http://wwwhome.cs.utwente.nl/~hiemstra/publications/. [Accessed 26 June 2017].

Probability estimates in practice. 2017. *Probability estimates in practice*. [ONLINE] Available at: https://nlp.stanford.edu/IR-book/html/htmledition/probability-estimates-in-practice-1.html. [Accessed 26 June 2017].

The Binary Independence Model. 2017. *The Binary Independence Model*. [ONLINE] Available at: https://nlp.stanford.edu/IR-book/html/htmledition/the-binary-independence-model-1.html. [Accessed 26 June 2017].