# Information Retrieval

### 17 Feb 2020

#### Information Retrieval’s goals is to get actionable data from a dataset

We are primarily focusing on three facets of our data: Representation, Similarity, and Mining/Retrieval/Learning. Since data has been growing at an alarming rate, getting actionable intel to make decisions from data has become increasingly more important.

#### Distance is the key (but, similarity is king)

How we consider the measurement of distance is obviously the important/interesting question, but most of what we will do in info retrieval is hinged upon this question of distance, and all it implies for similarity.

When talking distance, we mostly come back to the question of K-Nearest-Neighbors. This concept is interesting because a neighborhood can be considered a topology, and allows us to move from discrete values to a more continuous space.

However, it should be noted that distance is only key so far as in that it gets us to similarity. Distance allows us a metric in which to sort for similarity, which is the ultimate goal of getting actionable intel.

#### Flexibility over Formalism

Our goal is to make data actionable, so we’re utilizing looser formalism or less formalism, compared to say, a SQL query or an ontology (we can think of this work as the other side of working with ontology and formalisms).

#### Online access should be fast

This is the goal with information retrieval, that any online based query/request should be as fast as possible (waiting time is BORING to the user). Pre-processing (which happens offline) is very important to this pipeline.

The assumption here is that we know how/what to extract in terms of features to our dataset (i.e: in text datasets we want to extract words as features, or edges from images, etc.)

#### Bag of Words Model

(As seen already from Computational Linguistics experience)

Utilizes vocabulary to determine two texts similarity.

In this model, we have three concerns: representation of the document, representation of the query, and the ranking function. (We will list them out here for reference, in-depth later: document and query is represented via vectors, and the ranking function is TF-IDF)

#### Boolean Model

Based on set theory logic. Feature weights are binary, meaning that words are either in the document or not in the document. The functions are logical operators (AND, OR, NOT). For instance, we can get a query like, “alexa, search for geneva AND particle, OR CERN.” Meaning that this will yield documents that contain both geneva and particle, or contain CERN. The limitation here for the boolean model is of course that it is not flexible, since it is EXACT. I.e: A document that contains particle is worth just as much as a document that only contains cat, which hardly seems fair.

Additionally, it is notable that the documents that do match the query cannot be distinguished from one another. They are all just documents that match (the extent to the match is not measured).

The advantage here is the determinism and explicit queries. However, the limitation and rigidity makes it hard to use since we will rarely get matches or get too many matches (useless).

#### Vector Space Model

The goal here is to get Free-Text Queries, meaning entire documents can act as a query. This is more intuitive when you think about an image search rather than maybe a document search, since we don’t copy and paste entire documents, but you are essentially copy pasting an entire image when you search for it.

A document becomes a vector in a representational space.

We look at cosine distance as the ranking function, and the intuition for it acting better than an euclidean distance is due to the idea of measuring projections. Meaning that you measure the distance of the composition, rather than the physical space between the points. This gets you a more accurate representation of similarity.

Term Frequency: looks at

Inverse Doc Frequency: looks at

TF-IDF is simply the multiplication of these two values. The idea here is that the first measures how frequently a term appears in the document, since frequency is valuable. However, the IDF part evaluates whether or not the word is truly discriminative of the documents. The idea is that we want words that occur frequently, but occur frequently ONLY in select documents (meaning that they are telling of those documents, not just common filler words throughout all documents).

### 21 Feb 2020

#### We think about “relevance” in terms of a continuous understanding

That is to say, between (0,1) meaning that there is no “relevant” or “irrelevant” but only “more relevant” and “less relevant”. This type of continuous understanding is important because there runs into an issue of “all or nothing” of dimensions being represented in a way that they are not orthogonal, which causes inter-term dependence at times, and this could negatively impact our understanding or representation (check the mathematics of orthogonality and orthogonal basis).

#### Probabilistic IR is concerned with predicting an answer given queries

There are two models under the PIR that we can think about:

1. Regressive Model (looks at the document level and interpolates)

2. Generative Model (works on the documents as a sum of terms, and determine relevance based on a sum of the parts)

#### In the Generative Model, there are some assumptions that differ

Namely, we’re looking at Binary Features (presence of a term), as well as making a “uniqueness assumption” meaning that two documents with the same features are indistinguishable. Additionally, terms that are not present in the query do not factor in to the computation of relevance (see paper notes, but the idea is that if we have two documents that are identical under words 2 and 3, but differ in word 1, it won’t matter if the query does not contain word 1, the two docs would still be considered the same relevance)

Terms presence will contribute to both relevance and irrelevance, and this type of calculation and factoring of information is something that can probably be more formalized from the paper notes .