Information Retrieval using Cosine and Jaccard Similarity Measures in Vector Space Model

Abhishek Jain
Computer Science Department,
Bharati Vidyapeeth's College Of Engineering

Aman Jain
Computer Science Department,
Bharati Vidyapeeth's College Of Engineering

Nihal Chauhan
Computer Science Department,
Bharati Vidyapeeth's College Of Engineering

Vikrant Singh
Computer Science Department,
Bharati Vidyapeeth's College Of Engineering

Narina Thakur
HOD, Computer Science Department, Bharati Vidyapeeth's College of Engineering

ABSTRACT
With the exponential growth of documents available to us on the web, the requirement for an effective technique to retrieve the most relevant document matching a given search query has become critical. The field of Information Retrieval deals with the problem of document similarity to retrieve desired information from a large amount of data. Various models and similarity measures have been proposed to determine the extent of similarity between two objects. The objective of this paper is to summarize the entire process, looking into some of the most well-known algorithms and approaches to match a query text against a set of indexed documents.

Keywords
Weighting Measures, TF/IDF, Cosine Similarity Measure, Jaccard Similarity Measure, Information Retrieval.

1. INTRODUCTION
Retrieval of documents based on an input query is one of the basic forms of Information Retrieval. Web searches are the perfect example for this application. Many algorithms have been developed for this purpose, that take an input query and match it with the stored documents or text snippets and rank the output based on their similarity score respective to the given query.

Such algorithms rely on matching the indexed documents, which maintain the information concerning term frequencies and positions, against the individual query terms. A score is assigned to each document based on its similarity value. A query term's score with respect to a document is high for a high frequency of appearance in said document.

Different algorithms take different approaches in analysing this similarity and computing the score. One of the highly rated and used approaches is the Vector Space Model. It trumps the Boolean Model, which takes Boolean queries and matches the document with the query solely based on Boolean logic, whether the required terms are in the document or not. This often gives too few (≈ 0) or too many(1000s) documents.

The Vector Space Model along with the TF/IDF weighting scheme is explained in detail in section 2. The latest researches concerning the TF/IDF scheme are discussed in the Related Works section. In section 4, two similarity measures used with the TF/IDF scheme, i.e. Cosine and Jaccard are explained.

2. VECTOR SPACE MODEL
The Vector Space Model is a simple and the most popular model based on linear algebra allowing documents to be ranked based on their possible relevance.

This model represents text objects as vectors in an n-dimensional space, where n represents the number of terms

used to build an index to represent the documents [2]. The creation of an index requires the document to be striped and be segregated in the form of its unique terms. The document can then be further processed which reduces different forms of word into a common stem which helps increase the efficiency when matching of two documents.

The terms of a query can be weighted to account for their relative importance so as to give better results.

2.1 Weighting Scheme
The term frequency, $tf_{t,d}$ is the number of occurrences the term t in document d. The term is used in computing the document-query match scores. Since the relevance of a document should not increase linearly with the frequency of the term, it's effect is dampened by a logarithmic function. The log-frequency weighting of the term t in document d is defined as,

$$w_{t,d} = 1 + \log_{10}(tf_{t,d}) \quad (1)$$

Also, a rare term should be given higher importance than more frequent terms. For a term in a query that is rare in the collection, a document that is matched against it is more useful than other documents which are matched against more frequent terms. To account for this, document frequency is used. df_t is the document frequency of the term t, that is the number of documents in which the term t occurs. An inverse document frequency is defined for the usefulness of the term.

$$idf_t = \log_{10}(\frac{N}{df_t}) \quad (2)$$

Together a weight of the term is the product of its two weights: tf weight and idf weight. tf-idf is one of the best weighting scheme in the information retrieval.

$$w_{t,d} = (1 + \log_{10}(tf_{t,d})) \times (\log_{10}(\frac{N}{df_t}) \quad (3)$$

It increases with the increase in the number of occurrences of a term within a document and with the rarity of the query term in the collection.

2.2 Representation

With tf-idf being the best weighting scheme, the documents are then represented by the real valued vectors of tf-idf weights in an n-dimensional space, where n is the number of different terms/tokens used to index a set of documents.

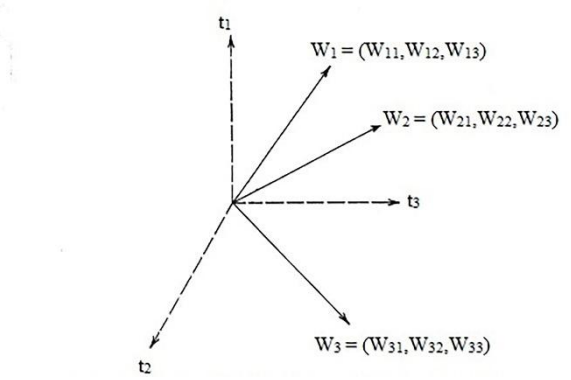

Figure 1: A document represented in a 3-dimensional term vector space

The magnitude of the vector for document d in dimension t is given by equation (4) if the term occurs in the document and by equation (5) otherwise.

$$w_{t,d} = (1 + \log_{10}(tf_{t,d})) \times (\log_{10}(\frac{N}{df_t}) \quad (4)$$

$$w_{t,d} = 0 \quad (5)$$

A documented is thus represented as:

$$W_d = (W_{d,1}, W_{d,2}, W_{d,3} \dots\dots, W_{d,N}) \quad (6)$$

With its magnitude given as:

$$|W_d| = (W_{d,1}^2 + W_{d,2}^2 + W_{d,3}^2 + \cdots + W_{d,N}^2)^{1/2e} \quad (7)$$

3. RELATED WORKS

Term weighting schemes are critical for text related information retrieval and have thus been well researched. TF/IDF is one of the most popular schemes, followed by Okapi BM25. The effectiveness of this scheme has led researchers to develop modified versions of the same. In the following text, we discuss three such modified schemes.

A new term weighting scheme, TF-ATO[6], Term Frequency with Average Term Occurrences, has been introduced, which as results have shown, is an improvement over the TF/IDF weighting scheme which is an important part of the Vector Space Model. The scheme computes the average term occurrences of terms in documents and takes a discriminative approach to remove the less significant weights from the documents. TF-ATO outperforms the TF/IDF scheme when used in combination with stop words removal and the discriminative approach suggested by the researchers.

Another modified TF-IDF scheme[7] has been proposed that combines relative TF-weighting and TF-normalization depending upon the document length. Most of the existing models employ a single term frequency normalization which does not balance well in preferring short and long documents for varying length queries. This weighting scheme employs two aspects of term frequency normalization to determine the importance of a term; one preferring short documents and one preferring long documents. These two terms are then combined using the query length information which maintains a balanced trade off in retrieving long and short documents when the ranking function faces queries of varying lengths.

M. Shirakawa et al[8] have shown that the IDF of a term is equal to the distance between the term and the empty string in the space of information distance in which the Kolmogorov complexity is approximated. Based on this finding, they have proposed a global term weighting scheme, N-gram IDF, a theoretical extension of IDF for handling words and phrases of any length. This scheme determines dominant N-grams among overlapping ones and extracts key terms of any length from texts without using any NLP techniques. It calculates the weight of all possible N-grams using two string processing techniques, i.e. maximal substring extraction and document listing. This scheme achieved competitive performance with state-of-the-art methods designed for key term extraction and web search query segmentation.

4. ALGORITHM

In VSM, the sets of documents and queries are viewed as vectors. A popular method for calculating the similarity value between the vectors with this model is the vector cosine measure. With document and queries being represented as vectors, similarity signifies the proximity between the two vectors (figure 2).

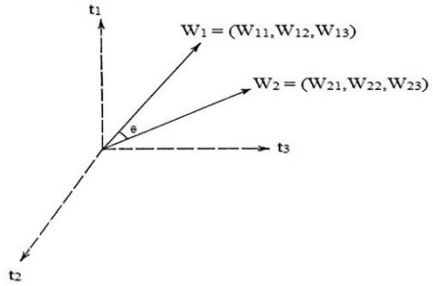

Figure 2: Illustration of angle similarity between 2 documents

Calculating this proximity through Euclidean distance does not give realistic results. Consider 2 documents d1 and d2 being completely identical. Appending the d2 with itself gives a new document d2'. Representing d1 and d2' as vectors, they are separated by a significant distance, even though they are practically identical. Thus, Euclidean distance is not a good measure in calculating the proximity between the 2 vectors.

To counter the above scenario, calculating similarity as a function of the angle made by the vectors seems the best possible option. If two vectors are close, the angle formed between them would be small and similarly if the two vectors are distant, the angle formed between them would be large. Instead of taking the angle formed by the vectors, the cosine value of the angle is considered. The cosine value varies from 1 to -1 for angles ranging from 0 to 180 degrees respectively, making it the ideal choice for these requirements. A score of 1 evaluates to the angle being 0^o, which means the document are similar. While a score of 0 evaluates to the angle being 90^o, which means the documents are entirely dissimilar.

The cosine weighting measure is implemented on length normalized vectors for making their weights comparable. Equation (8) gives the formula for Cosine Similarity.

$$similarity = \cos(\vec{d}, \vec{q}) = \frac{\vec{d} \cdot \vec{q}}{\left|\left|\vec{d}\right|\right| \left|\left|\vec{q}\right|\right|} = \frac{\sum_{i=1}^{|V|} d_i q_i}{\sqrt{\sum_{i=1}^{|V|} d_i^2} \sqrt{\sum_{i=1}^{|V|} q_i^2}} \quad (8)$$

Jaccard Similarity measure is another measure for calculating the similarity in the queries and documents. In this measure, the index starts with a minimum value of 0 (completely dissimilar) and goes to a maximum value of 1 (completely similar).

This value is calculated by the equation (9).

$$similarity = Jaccard(Q, D) = \frac{|Q \cap D|}{|Q \cup D|} \quad (9)$$

5. DISCUSSION

The Vector Space Model is a simple model based on linear algebra which was designed to overcome the limitations of the Boolean Model. One of the major advantages of VSM over the Boolean Model is that the weights assigned to the term are not binary. This allows for better matching by computing over a range of similarity values and thus eliminating the too few or too many results obtained with the Boolean Model.

On the other hand, VSM has its own limitations. The major limitation being of low sensitivity to semantics. For example, the words "car" and "automobile" won't give us a match, that is two identical phrases with one using the word 'car' and the other using the word 'automobile' will not give a match. Also, it doesn't distinguish phrases on the basis of ordering of the constituent terms. For eg. "Mary is faster than John" is indistinguishable from "John is faster than Mary" using this approach.

Cosine and Jaccard are two basic and effective similarity measures used in conjunction with the TF/IDF weighting scheme.

6. CONCLUSION

This paper gives a brief overview of a basic Information Retrieval model, VSM, with the TF/IDF weighting scheme and the Cosine and Jaccard similarity measures. This field has seen a lot of research in the past decade. This research has been focused on developing better models, some are extensions of VSM such as LSA[18] or Random Indexing[19] while others are based on entirely different principles such as the Probabilistic Relevance Model[20].

The explosive growth of information demands better information retrieval techniques. Thus, future work in this field should be focused on developing new models, weighting schemes and similarity measures that can perform effectively on large data sets utilising semantic information on the same.

7. REFERENCES

[1] "Roshdi, Akram, and Akram Roohparvar. "Review: Information Retrieval Techniques and Applications."

[2] "Salton, Gerard, and Christopher Buckley. "Term-weighting approaches in automatic text retrieval." *Information processing & management* 24.5 (1988): 513-523."

[3] "Le, Quoc V., and Tomas Mikolov. "Distributed Representations of Sentences and Documents." *ICML*. Vol. 14. 2014."

[4] "Singh, Vaibhav Kant, and Vinay Kumar Singh. "VECTOR SPACE MODEL: AN INFORMATION RETRIEVAL SYSTEM." *Int. J. Adv. Engg. Res. Studies/IV/II/Jan.-March* 141 (2015): 143."

[5] "Deshmukh, Ashwini, et al. "A Literature Survey On Latent Semantic Indexing." *International Conference on Computing*. 2012.

[6] "Ibrahim, O., and D. Landa-Silva. "Term frequency with average term occurrences for textual information retrieval." *Soft Comput* 20.8 (2016): 3045-3061."

[7] "Paik, Jiaul H. "A novel TF-IDF weighting scheme for effective ranking." *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2013."

[8] "Shirakawa, Masumi, Takahiro Hara, and Shojiro Nishio. "N-gram IDF: A Global Term Weighting Scheme Based on Information Distance." *Proceedings of the 24th International Conference on World Wide Web*. ACM, 2015."

[9] "Ghag, Kranti, and Ketan Shah. "SentiTFIDF–Sentiment Classification using Relative Term Frequency Inverse Document Frequency." *Int. J. Adv. Comput. Sci. Appl. Sci. Inf. Organ* (2014)."

[10] "Quercia, Daniele, et al. "Recommending social events from mobile phone location data." *2010 IEEE International Conference on Data Mining*. IEEE, 2010."

[11] "Nguyen, Hieu V., and Li Bai. "Cosine similarity metric learning for face verification." *Asian Conference on Computer Vision*. Springer Berlin Heidelberg, 2010."

[12] "Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. "Glove: Global Vectors for Word Representation." *EMNLP*. Vol. 14. 2014."

[13] "Steinbach, Michael, George Karypis, and Vipin Kumar. "A comparison of document clustering techniques." *KDD workshop on text mining*. Vol. 400. No. 1. 2000."

[14] "Yin, Jie, et al. "Using social media to enhance emergency situation awareness." *IEEE Intelligent Systems* 27.6 (2012): 52-59."

[15] "O'Connor, Brendan, Michel Krieger, and David Ahn. "TweetMotif: Exploratory Search and Topic Summarization for Twitter." *ICWSM*. 2010."

[16] "Choi, Seung-Seok, Sung-Hyuk Cha, and Charles C. Tappert. "A survey of binary similarity and distance measures." *Journal of Systemics, Cybernetics and Informatics* 8.1 (2010): 43-48."

[17] "S. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. In Proceedings of Text Retrieval Conference (TREC), pages 109–126, 1994."

[18] "Deerwester, Scott, et al. "Indexing by latent semantic analysis." *Journal of the American society for information science* 41.6 (1990): 391."

[19] "Sahlgren, Magnus. "An introduction to random indexing." *Methods and applications of semantic indexing workshop at the 7th international conference on terminology and knowledge engineering, TKE*. Vol. 5. 2005."

[20] "Robertson, Stephen, and Hugo Zaragoza. *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc, 2009."