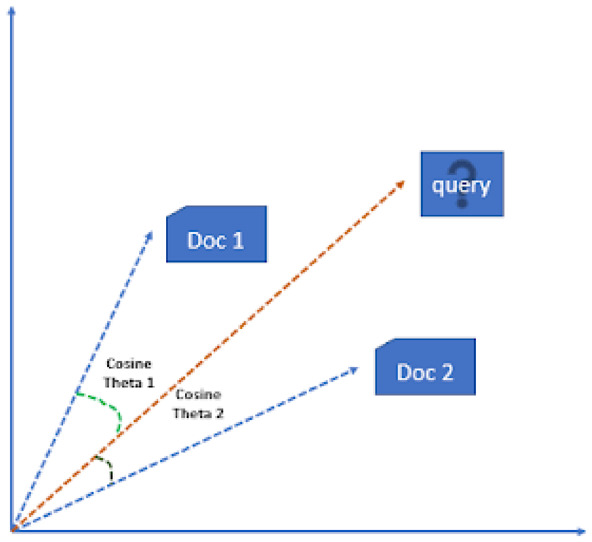
Document Similarity

Natural Language Processing



Agenda

- What is Document Similarity
- Methods to measure Document Similarity
- Cosine Similarity Method

Goal

- Given a set of documents and search term(s)/query we need to retrieve relevant documents that are **similar to the search query**.

Match Relevant Documents

- a measure of similarity that can be used to
  - compare documents or
  - provide a ranking of documents
- with respect to a given vector of query words.

Cosine similarity measure

- Cosine similarity documents are represented as term vectors.
- The similarity of two documents corresponds to the correlation between the vectors.
- This is quantified as the cosine of the angle between vectors - known as cosine similarity.
- Cosine similarity is one of the most popular similarity measure applied to text documents

Cosine Similarity Method

- Vector Space Model
- TF , TDF …
- Cosine Similarity Method Calculation

Vector Space Model (VSM)

- Vector Space Model (VSM) is a way of representing documents through the words that they contain
- It is a standard technique in Information Retrieval

Steps - Cosine Similarity

- Step 1 : Term frequency (TF)
- Step 2 : Inverse Document Frequency(IDF)
- Step 3 : TF * IDF
- Step 4 : Vector Space Model Cosine Similarity

Example

- Document 1: The game of life is a game of everlasting learning
- Document 2: The unexamined life is not worth living
- Document 3: Never stop learning

Step 1 : Term frequency (TF)

- The term frequency $tf_{t,d}$ of term *t* in document *d* is defined as the number of times that *t* occurs in *d*.

Step 1 : Term frequency (TF)

Document1	the	game	of	life	is	a	everlasting	learning
Term Frequency	1	2	2	1	1	1	1	1

Document2	the	unexamined	life	is	not	worth	living
Term Frequency	1	1	1	1	1	1	1

Document3	never	stop	learning
Term Frequency	1	1	1

- Document 1: The game of life is a game of everlasting learning
- Document 2: The unexamined life is not worth living
- Document 3: Never stop learning

Normalized TF

Document1	the	game	of	life	is	a	everlasting	learning
Normalized TF	0.1	0.2	0.2	0.1	0.1	0.1	0.1	0.1

Document2	the	unexamined	life	is	not	worth	living
Normalized TF	0.142857	0.142857	0.142857	0.142857	0.142857	0.142857	0.142857

Document3	never	stop	learning
Normalized TF	0.333333	0.333333	0.333333

Step2:Inverse Document Frequency(IDF)

- The main purpose of doing a search is to find out **relevant documents** matching the query.
- In the first step all terms are considered equally important.
- Certain terms that occur too frequently have little power in determining the relevance.
- We need a way to **weigh down** the effects of too frequently occurring terms.
- Also the terms that occur less in the document can be more relevant.
- We need a way to **weigh up** the effects of less frequently occurring terms.
- Logarithms helps to solve this problem.

idf - Inverse Document Frequency

- "inverse document frequency"
- measures how common a word is among all documents in bloblist.
- More common a word is, the lower its idf.
- We take the ratio of the total number of documents to the number of documents containing word, then take the log of that.
- Add 1 to the divisor to prevent division by zero.

- IDF(**game**) = 1 + $\log_e$(Total Number Of Documents / Number Of Documents with term **game** in it)
- There are 3 documents in all
  - Document1, Document2, Document3
- The term game appears in Document1
- IDF(**game**) = 1 + $\log_e(3 / 1)$ = 1 + 1.098726209 = 2.098726209

Terms	IDF
the	1.405507153
game	2.098726209
of	2.098726209
life	1.405507153
is	1.405507153
a	2.098726209
everlasting	2.098726209
learning	1.405507153
unexamined	2.098726209
not	2.098726209
worth	2.098726209
living	2.098726209
never	2.098726209
stop	2.098726209

Step 3 : TF * IDF

- to find out relevant documents for the query: life learning
- For each term in the query multiply its normalized term frequency with its IDF on each document.
- In Document1 for the term life the normalized term frequency is 0.1 and its IDF is 1.405507153.
- Multiplying them together we get 0.140550715 (0.1 * 1.405507153).

Step 3 : TF * IDF

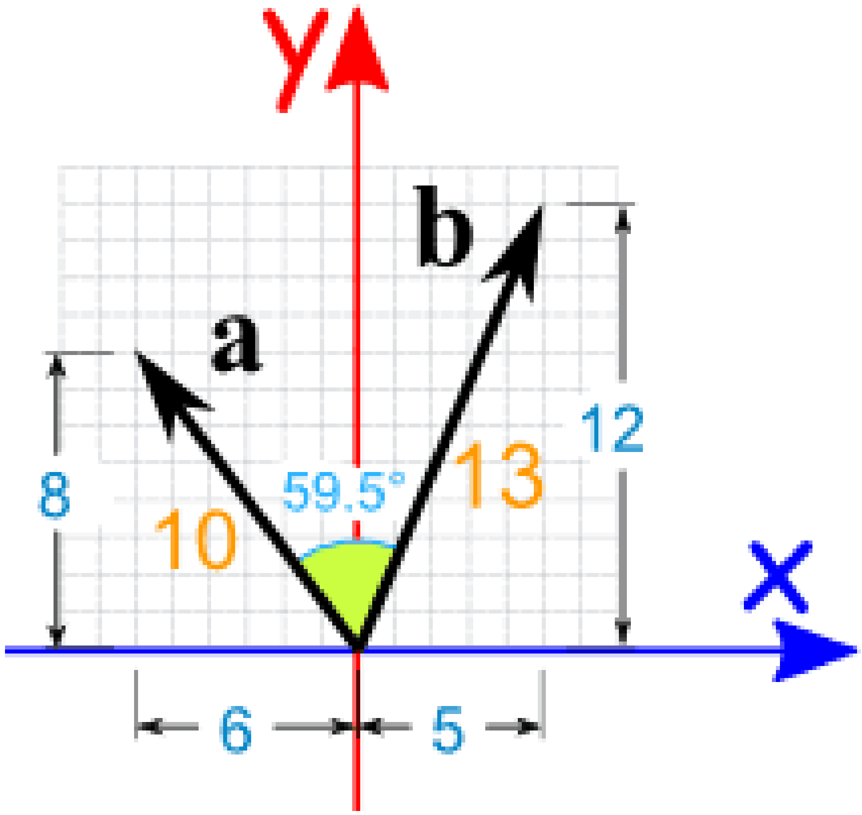
	Document1	Document2	Document3
life	0.140550715	0.200786736	0
learning	0.140550715	0	0.468502384

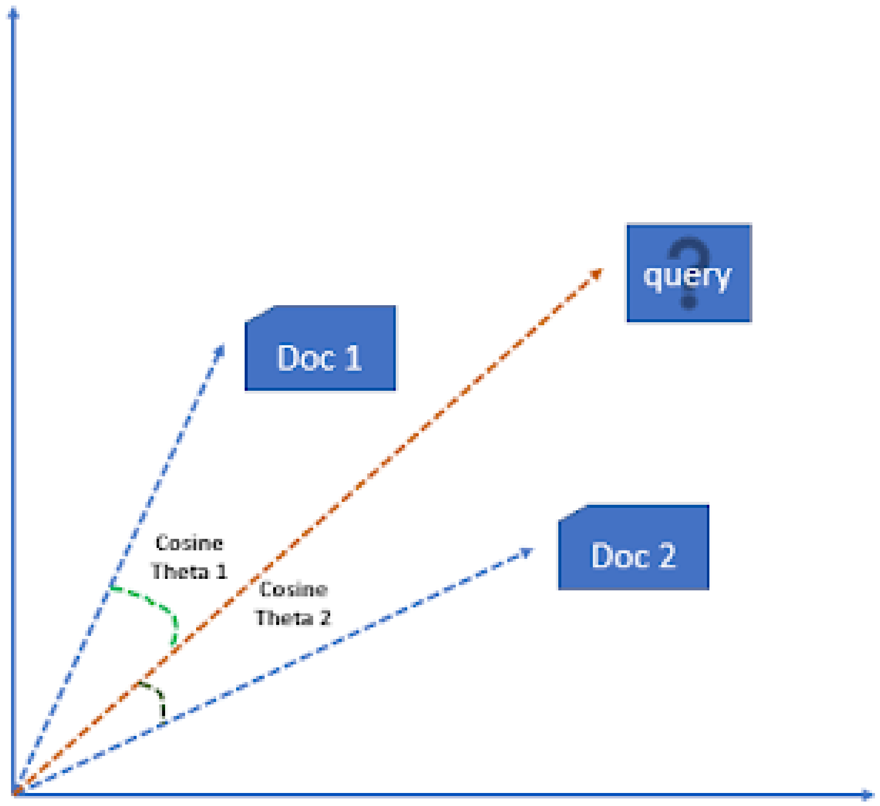
Step 4:Vector Space Model

- The representation of a set of documents as vectors in a common vector space is known as the *vector space model*
- It is fundamental to a host of information retrieval operations ranging from
  - scoring documents on a query,
  - document classification and
  - document clustering.

Step 4:Vector Space Model Cosine Similarity

- From each document we derive a vector.
- The set of documents in a collection then is viewed as a set of vectors in a vector space.
- Each term will have its own axis.


y
x
a
b
8
10
59.5°
13
12
6
5


query
Doc 1
Doc 2
Cosine
Theta 1
Cosine
Theta 2

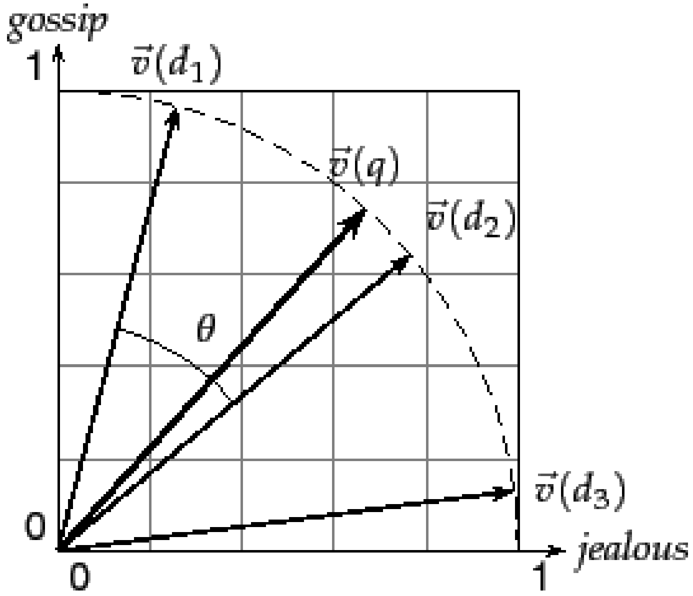
Overview

The Vector Space Model (VSM) is a way of representing documents through the words that they contain

It is a standard technique in Information Retrieval

The VSM allows decisions to be made about which documents are similar to each other and to keyword queries

Step 4:Vector Space Model Cosine Similarity



Cosine Similarity

$$\cos\theta = \frac{\mathbf{d_2} \cdot \mathbf{q}}{\|\mathbf{d_2}\| \, \|\mathbf{q}\|}$$

Ranking documents

A user enters a query

The query is compared to all documents using a similarity measure

The user is shown the documents in decreasing order of similarity to the query term