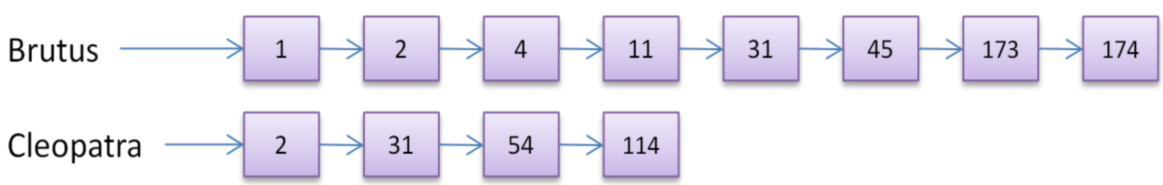
BOOLEAN RETRIEVAL

- Most simple retrieval and relies on the use of Boolean operators.
- The term in a query are linked together with AND, OR and NOT.
- Terms weights are set to 1 if the terms are occurred in the documents.

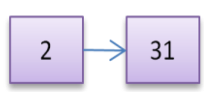
INTERSECTION ALGORITHM TO COMPUTE BOOLEAN QUERY

```
INTERSECT (p1, p2)
answer ← ( )
while p1! = NIL and p2! = NIL
    do if docID (p1) – docID (p2)
    then ADD (answer, docID (p1))
        p1 ← next (p1)
        p2 ← next (p2)
    else if docID (p1) < docID (p2)
    then p1 ← next (p1)
    else p2 ← next (p2)
return answer
```

EXAMPLE



Intersection (Brutus AND Cleopatra) →



EXAMPLE

d1 = English tutorial and fast track

d2 = Book on semantic analysis

d3 = Learning latent semantic indexing

d4 = Advance in structure and semantic indexing

d5 = Analysis of latent structures

Query → "advance AND structure AND NOT analysis"

Terms	d1	d2	d3	d4	d5
English	1	0	0	0	0
Tutorial	1	0	0	0	0
Fast	1	0	0	0	0
Track	1	0	0	0	0
Book	0	1	0	0	0
Semantic	0	1	1	1	0
Analysis	0	1	0	0	1
Learning	0	0	1	0	0
Structure	0	0	0	1	1
Indexing	0	0	1	1	0
Latent	0	0	1	0	1
Advance	0	0	0	1	0

Fig: Term Document Matrix

Solution:

Query Terms	d1	d2	d3	d4	d5
Advance	0	0	0	1	0
Structure	0	0	0	1	1
	0	0	0	1	0
NOT Analysis	1	0	1	1	0
	0	0	0	1	0

LIMITATION OF BOOLEAN RETRIEVAL

- Very rigid → AND means all & OR means any.
- Difficult to control the number of documents retrieved, i.e. all matched documents will be returned.
- Incapable to rank the output [i.e. all matched documents logically satisfy the query].
- Using many Boolean operators make the query complex to formulate.
- Good for specific user having good knowledge on Boolean operation.
- Not good for majority of the users.

RANK RETRIEVAL

- System decides which documents best satisfy the query.
- Vector space model.

VECTOR SPACE MODEL (VSM)

- A vector space model is a mathematical structure formed by a collection of vectors.
- A point in the space represents a vector.
- The set of all n-tuples (x1, x2,……, xn) of n real numbers is known as n-space where n being a positive integer.
- All the documents are represented by a point in a space of n dimension by n term co-ordinate.
- Queries are treated like documents.
- Documents are ranked by closeness to the query.
- Closeness is determined by a similarity score calculation.

MAJOR PROPERTIES OF VSM

- Ranking of documents according to similarity value.
- Documents can be retrieved even if they don't contain some query keyword.

COSINE SIMILARITY

- Scores the similarity between two document vectors.
- The similarity between the two vectors is defined by the angle between them.
- If the two vectors are exactly similar then the angle between the two vectors are zero and thus cosine equal to 1, representing the perfect match.
- If the two vectors are perfectly dissimilar, then the angle between the vectors is perfect 90° and the cosine equal to 0, represents the perfect dissimilar.

POINTS IN A PLANE

- Points in a two dimension XY plane is defined by a pair of co-ordinates.

DOT PRODUCT

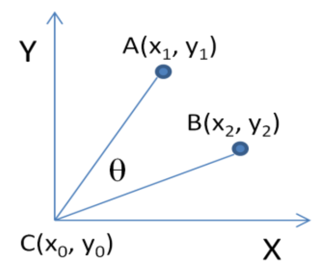
- Dot product is an algebraic operation that takes two co-ordinates vector and returns a single number obtained by multiplying corresponding entries and adding up those products.
- A.B = $x_1x_2 + y_1y_2$
- If A and B are in 3D, A.B = $x_1x_2 + y_1y_2 + z_1z_2$
- In general, if A = $(a_1, a_2, \ldots\ldots, a_n)$ and B = $(b_1, b_2, \ldots\ldots, b_n)$, then, A.B = $\sum_{i=1}^{n} a_i \; . b_i$

EUCLIDEAN DISTANCE

- Euclidean distance is the distance between two points, one being the origin point.
- i.e. $d_{AC} = \sqrt{(x_1 - x_0)^2 + (y_1 - y_0)^2} = \sqrt{x_1{}^2 + y_1{}^2}$

  $d_{BC} = \sqrt{(x_2 - x_0)^2 + (y_2 - y_0)^2} = \sqrt{x_2{}^2 + y_2{}^2}$

REPRESENTING DOCUMENT VECTOR



- A vector is a quality with direction and magnitude.
- The head and angle of the arrow indicates the direction of the vector.
- Magnitude is defined by Euclidean distance.

DOCUMENT LENGTH NORMALIZATION

- To normalize A.B, the dot product, it is divided by the Euclidean distances of A and B,

  i.e. $\frac{A.B}{|A||B|}$
- The ratio defines the cosine angle between the vectors, with values between 0 and 1.
- This ratio is used as a similarity measure between any two vectors representing documents, queries denoted by sim (A, B)

  i.e. sim (A, B) = cosine θ

  $= \frac{A.B}{|A||B|}$

  $= \frac{x_1 x_2 + y_1 y_2}{\sqrt{x_1{}^2 + y_1{}^2}\sqrt{x_2{}^2 + y_2{}^2}}$

QUERIES OF VECTORS

- By viewing a query as a "bag of words", it is able to treat as a very short document.

  Score (q, d) = $\frac{\vec{v}(q) \cdot \vec{v}(d)}{|\vec{v}(q)||\vec{v}(d)|}$

- A document may have a high score for a query even if it does not contain all query terms.

LINEAR ALGEBRA APPROACH TO TERM VECTOR

- Example:

  DOC 1 → Linear (3 times), algebra (1 times), approach (3 times)

  DOC 2 → Linear (1 times), algebra (2 times), approach (4 times)

  DOC 3 → Linear (2 times), algebra (3 times), approach (0 times)

  Query → Approach

Term	DOC 1	DOC 2	DOC 3	Query
Linear	3	1	2	0
Algebra	1	2	3	0
Approach	3	4	0	1
Co-ordinate	(3, 1, 3)	(1, 2, 4)	(2, 3, 0)	(0, 0, 1)
Magnitude (L_d)	$\sqrt{19}$	$\sqrt{21}$	$\sqrt{13}$	$\sqrt{1}$

A = term – document $\begin{bmatrix} 3 & 1 & 2 \\ 1 & 2 & 3 \\ 3 & 4 & 0 \end{bmatrix}$

q = query matrix = $\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$ and $q^T = [0 \quad 0 \quad 1]$

Normalize A, $\begin{bmatrix} \frac{3}{\sqrt{19}} & \frac{1}{\sqrt{21}} & \frac{2}{\sqrt{13}} \\ \frac{1}{\sqrt{19}} & \frac{2}{\sqrt{21}} & \frac{3}{\sqrt{13}} \\ \frac{3}{\sqrt{19}} & \frac{4}{\sqrt{21}} & \frac{0}{\sqrt{13}} \end{bmatrix}$

Now, $q^TA = [\frac{3}{\sqrt{19}} \quad \frac{4}{\sqrt{21}} \quad \frac{0}{\sqrt{13}}]$

= (0.68, 0.87, 0)

i.e. sim (q, DOC 1) = 0.68; sim (q, DOC 2) = 0.87; sim (q, DOC 3) = 0

WEIGHTING

- Weight of a term is a value given to the term.
- Value is the dependent factor of its occurrence in the document.
- Weight of a term is a basic element for the document ranking.
- Weighting mechanism:

(1) Term Frequency

- Term frequency is a measure of how often a term is found in a collection of documents.
- A reasonable scoring mechanism is computed a score for each query terms that matches with the document terms.
- Count the frequency of the terms that matches between the query terms and the document terms list.
- Denoted by $tf_{t,d}$.

(2) Inverse Document Frequency

- Term frequency suffers from a critical problem that all terms are considered equally important.
- In fact, certain terms have little or no selective power in determining relevance.
- For example: a collection of documents of the "Noodle" industry is likely to have the term "Noodle" in almost every document.
- Terms which appear very few in numbers may have higher probability of being relevant.
- So, we have to scale down the term weights of term with high collection frequency.
- Collection frequency is the total number of occurrence of a term in the collection.
- Document frequency is the number of documents in the collection that contain a term t.

Words	c.f	df
Book	10200	8532
Pen	10198	4502

$idf_t = \log \frac{N}{df_t}$

- For example:

Terms (t)	df_t	idf_t
Computer	1054	0.152
Monitor	508	0.470
Keyboard	475	0.500
Device	1247	0.080
Optical	1500	0

N = Total number of documents = 1500

- It is seen that the term having the highest df has the lowest idf and vice-versa.

TF – IDF WEIGHTING

- Terms are weighted according to a given weighting model which may include local weight, global weight or both.
- Local weights are functions of how many times each term appear in a document.
- Global weights are functions of how many times each term appears in the entire collection.
- The tf – idf weight for a term t in a document d is given by, $tf - idf_{t,d} = tf_{t,d} \times idf_t$, which is
  - Highest when t occurs within a small number of documents.
  - Lowers when the term t occurs fewer times in a document.
  - Lowest when the terms t occurs in virtually all documents.

ALGORITHM (VECTOR SPACE MODEL FOR DOCUMENT RANKING)

- A term document matrix 'A' is constructed.
- Weight for each element of the matrix is defined, $a_{ij} = L_{ij} \times G_i \times N_j$

  where, L_{ij} = local weight of a term i in document j($tf_{i,j}$)

  G_i = global weight (idf_i)

  N_j = Normalization function = 1/l; l = Euclidean distance of document j
- Query matrix Q is defined.
- A X Q^T is computed.
- Obtained result shows the rank of the document.