Information Retrieval & Text Mining

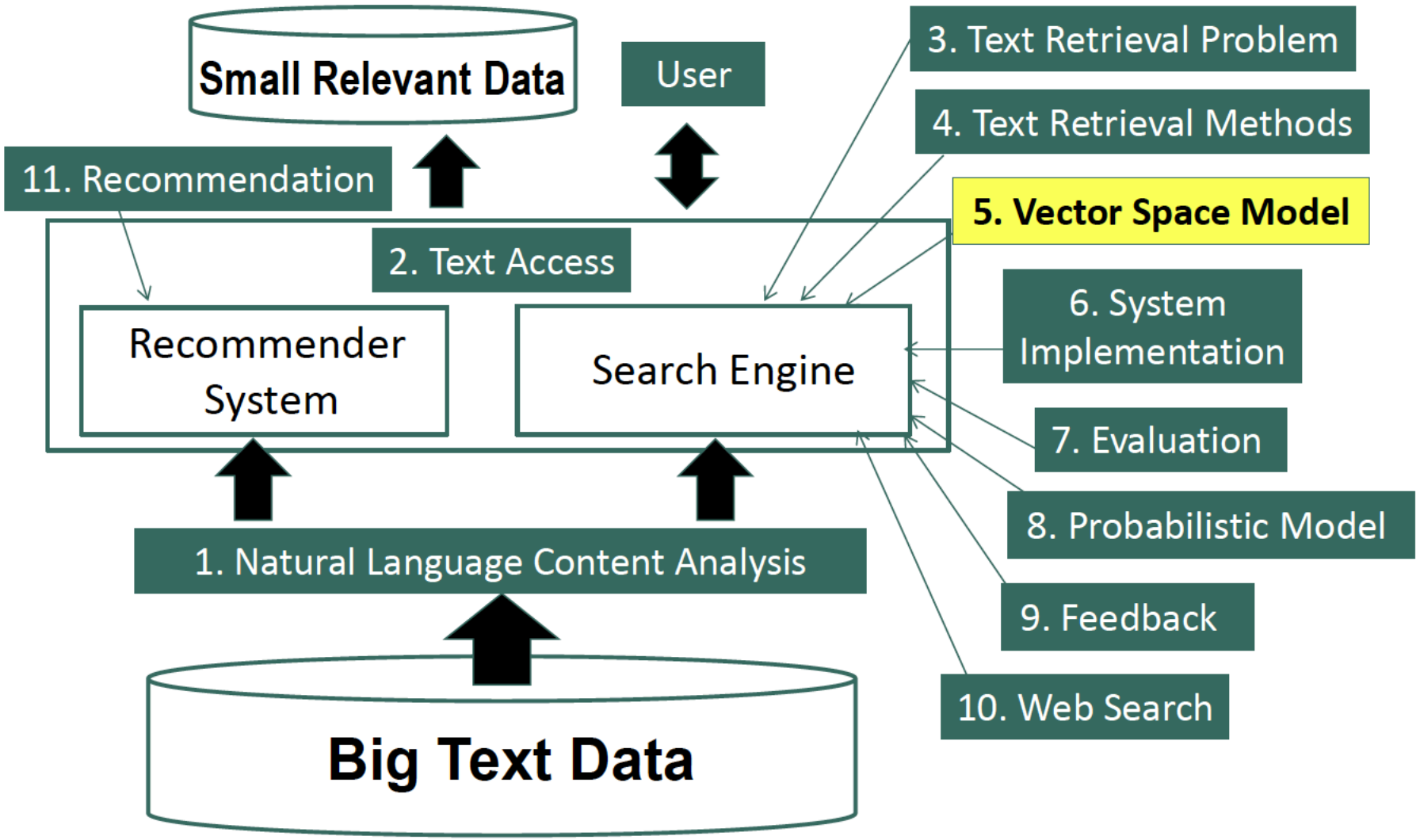
Vector Space Model

Simplest Instantiation

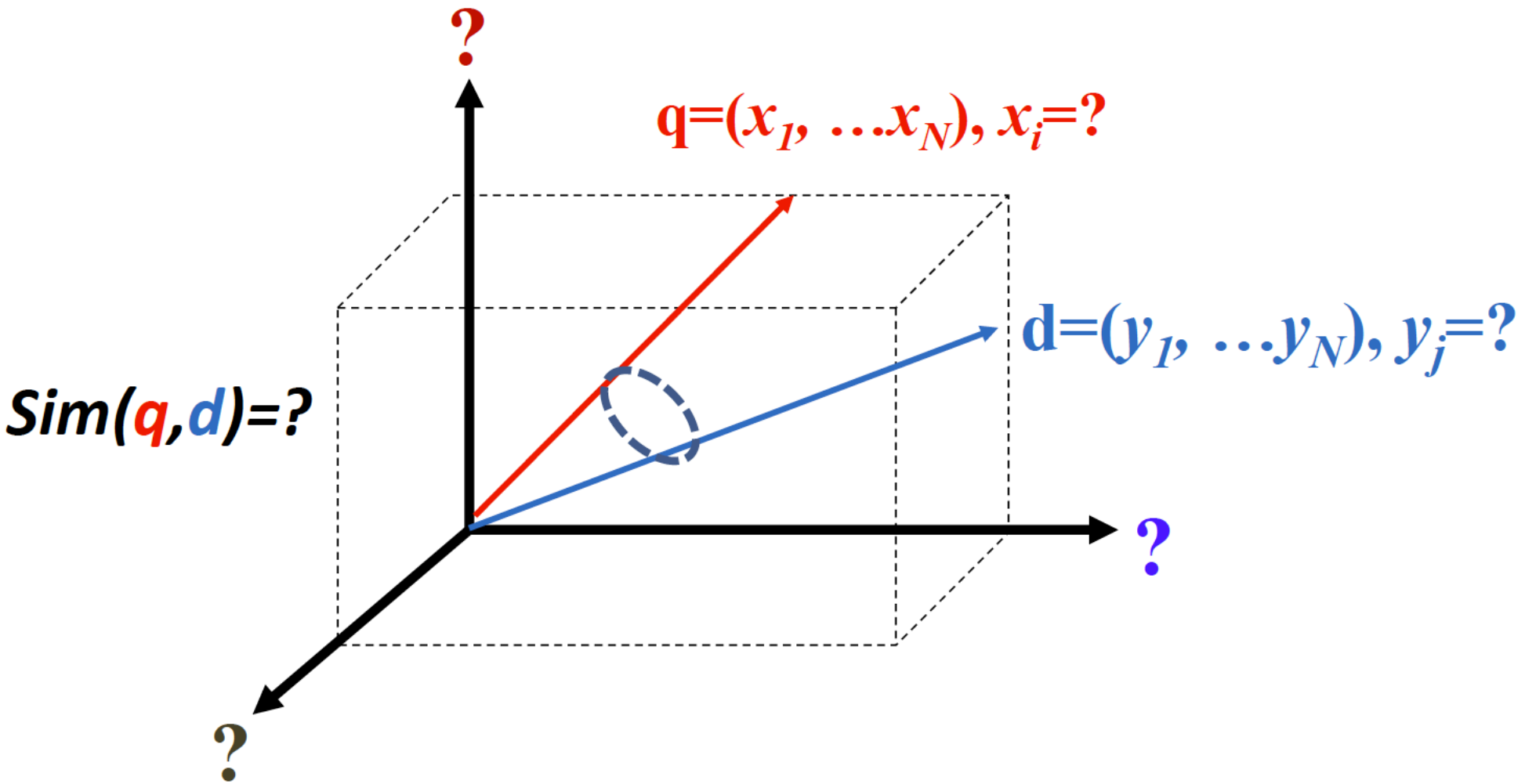
Dr. Saeed Ul Hassan

Information Technology University

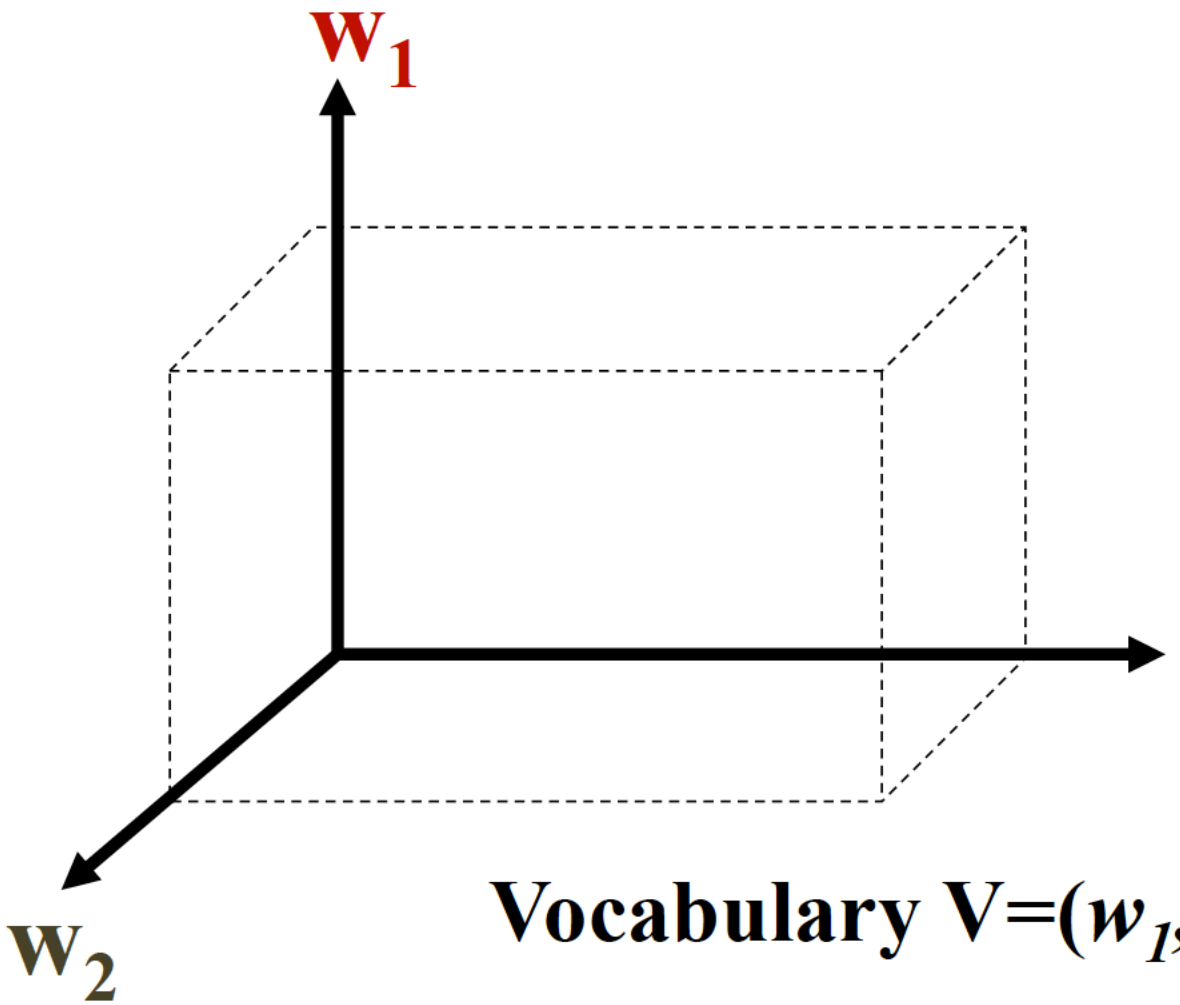
Course Schedule



What VSM Doesn't Say

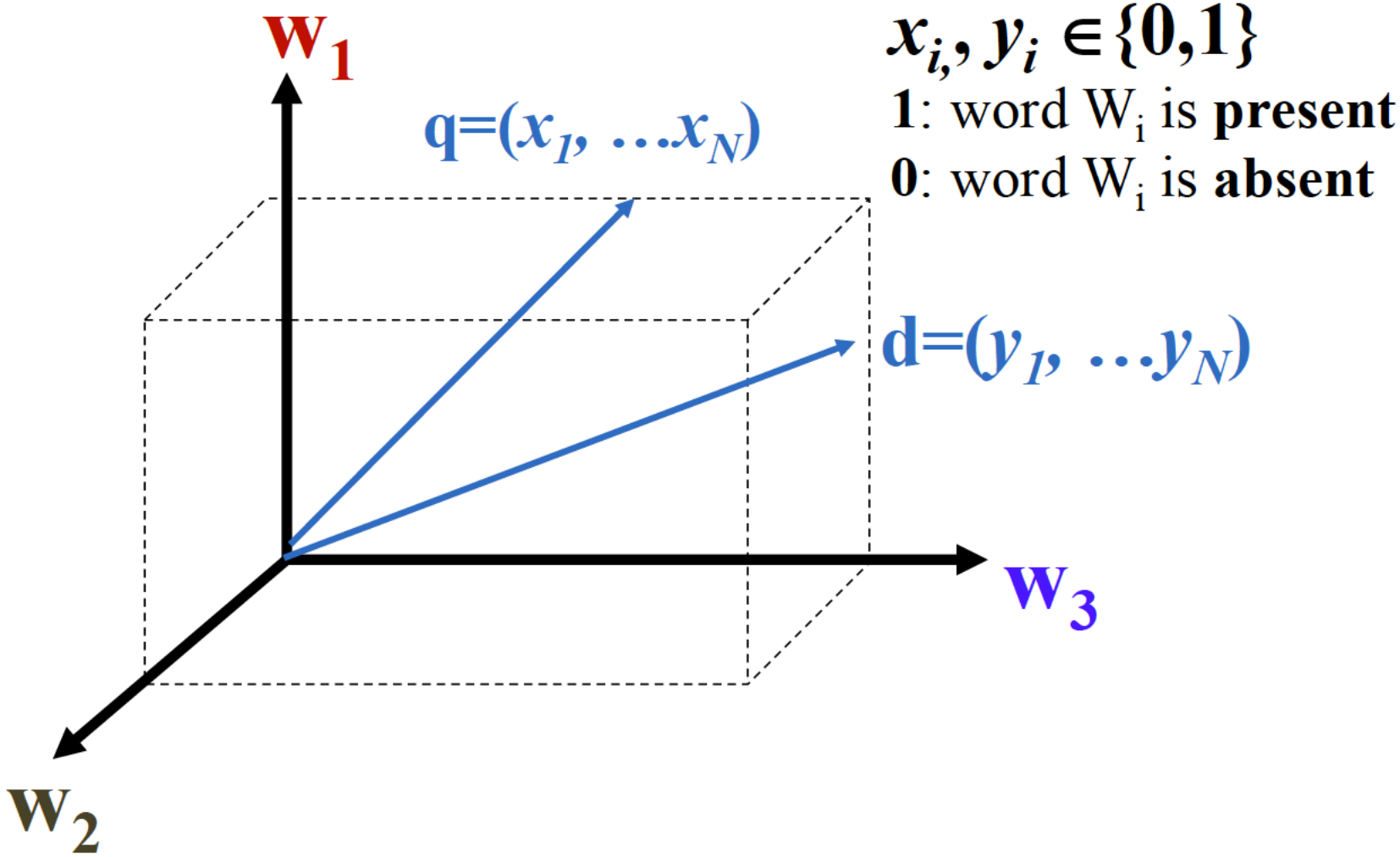


Dimension Instantiation: Bag of Words (BOW)



Vocabulary V=(w_1, …w_N)

Vector Placement: Bit Vector



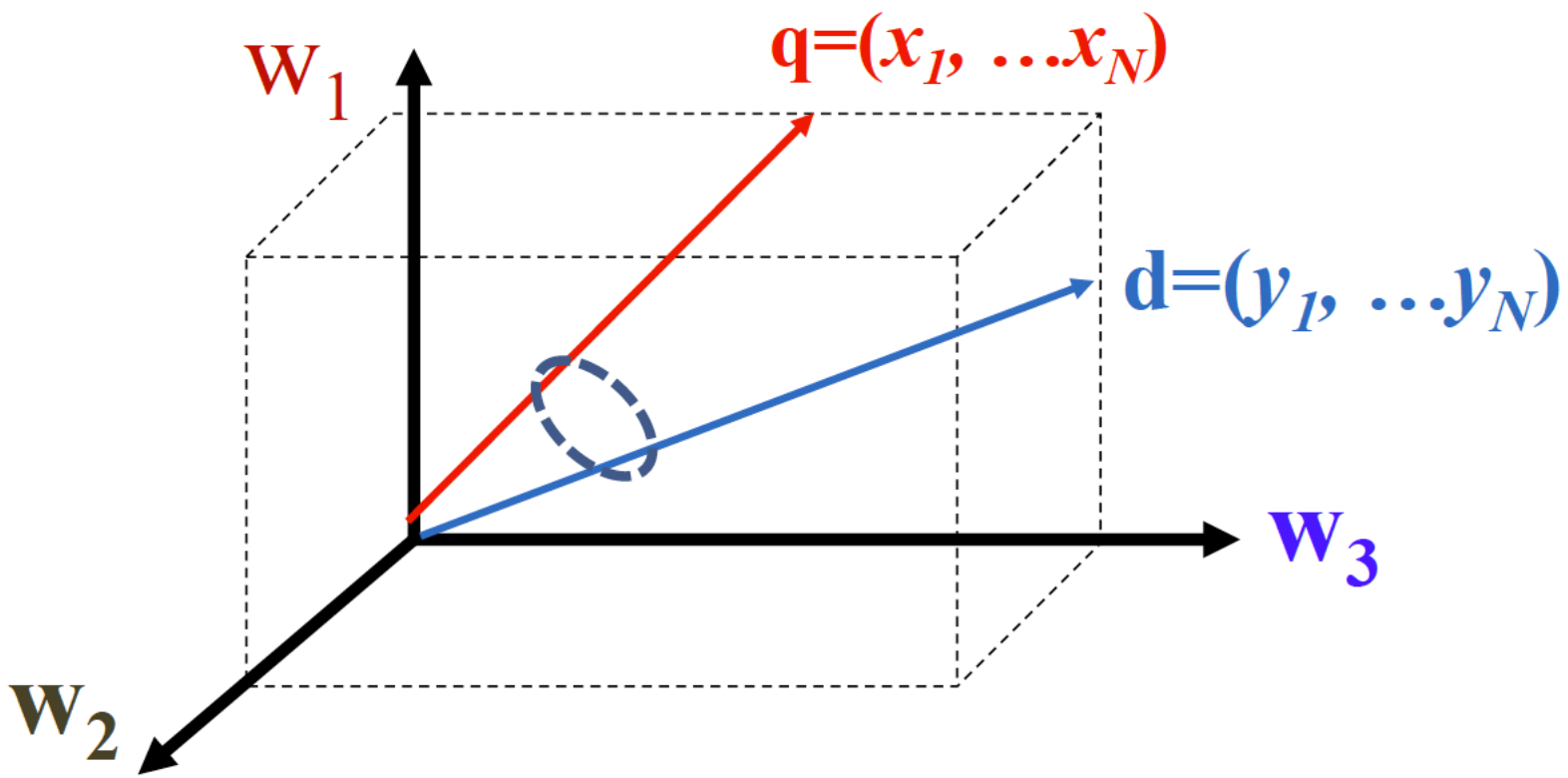
$x_i, y_i \in \{0,1\}$

1: word W_i is **present**

0: word W_i is **absent**

Similarity Instantiation: Dot Product

$$Sim(q,d)=q.d= x_1 y_1 + \ldots + x_N y_N = \sum_{i=1}^{N} x_i y_i$$



Simplest VSM= Bit-Vector + Dot-Product + BOW

$q=(x_1, \ldots x_N)$

$d=(y_1, \ldots y_N)$

$x_i, y_i \in \{0,1\}$

1: word W_i is **present**

0: word W_i is **absent**

$$Sim(q,d)=q.d= x_1 y_1 + \ldots + x_N y_N = \sum_{i=1}^{N} x_i y_i$$

What does this ranking function intuitively capture?
Is this a good ranking function?

An Example: How Would You Rank These Documents?

Query = "news about presidential campaign"

		Ideal Ranking?
d1	… news about …	d4 +
d2	… news about organic food campaign…	d3 +
d3	… news of presidential campaign …	
d4	… news of presidential campaign … … presidential candidate …	d1 -
d5	… news of organic food campaign… campaign…campaign…campaign…	d2 - d5 -

Ranking Using the Simplest VSM

Query = "news about presidential campaign"

d1 … **news about** …

d3 … **news** of **presidential campaign** …

V= {news, about, presidential, campaign, food …. }

q= (1, 1, 1, 1, 0, …)

d1= (1, 1, 0, 0, 0, …)

f(q,d1)= 1*1+1*1 +1* 0+1*0+0*0 + …= 2

d3= (1, 0, 1, 1, 0, …)

f(q,d3)= 1*1+1*0 +1*1 +1*1+0*0+ … = 3

Is the Simplest VSM Effective?

Query = "news about presidential campaign"

d1	… **news about** …	f(q,d1)=2
d2	… **news about** organic food **campaign**…	f(q,d2)=3
d3	… **news** of **presidential campaign** …	f(q,d3)=3
d4	… **news** of **presidential campaign** … … **presidential** candidate …	f(q,d4)=3
d5	… **news** of organic food **campaign**… **campaign**…**campaign**…**campaign**…	f(q,d5)=2

Summary

- VSM instantiation: dimension, vector placement, similarity

- Simplest VSM
  - Dimension = word
  - Vector = 0-1 bit vector (word presence/absence)
  - Similarity = dot product
  - f(q,d) = number of **distinct** query words matched in d