Information Retrieval & Text Mining
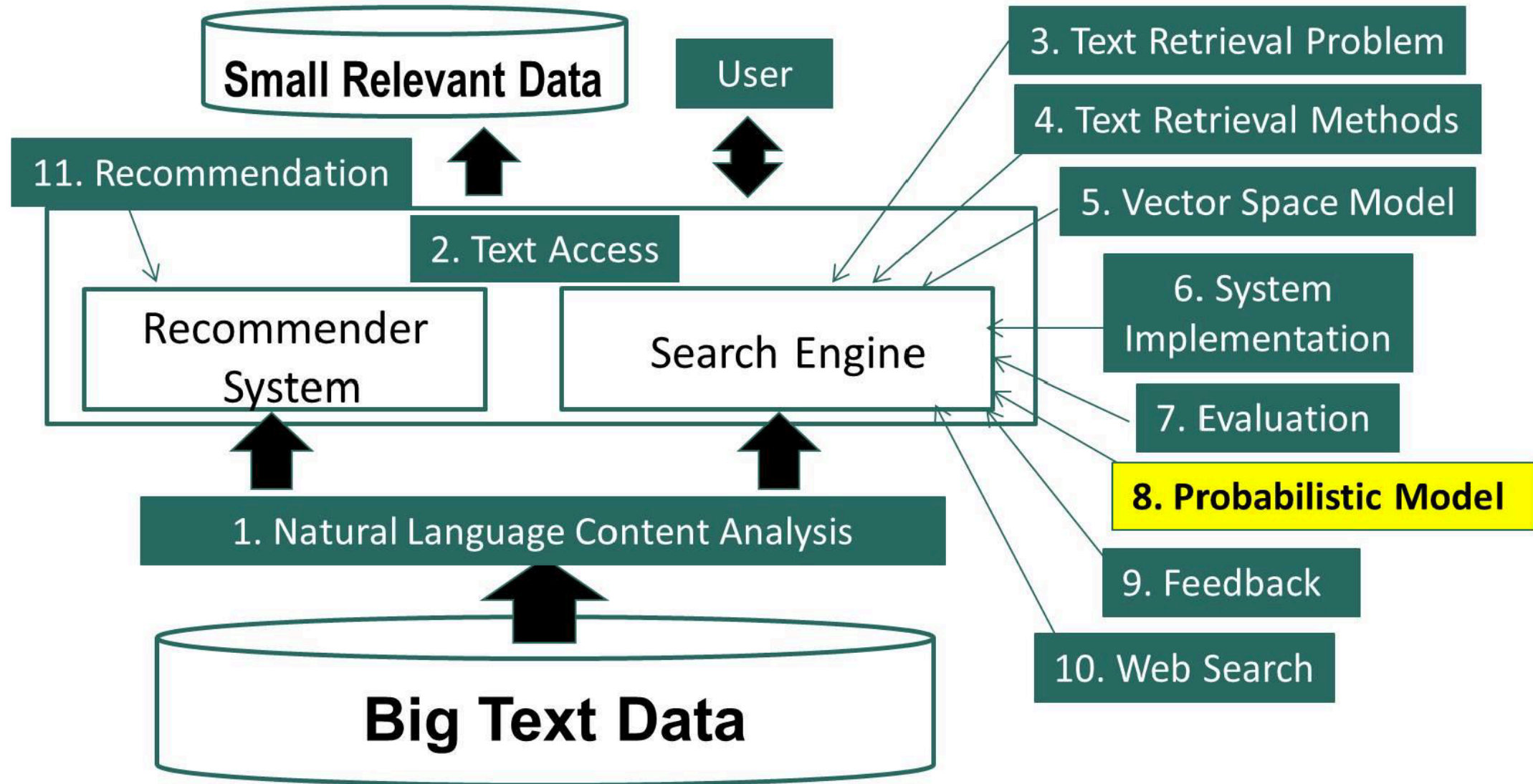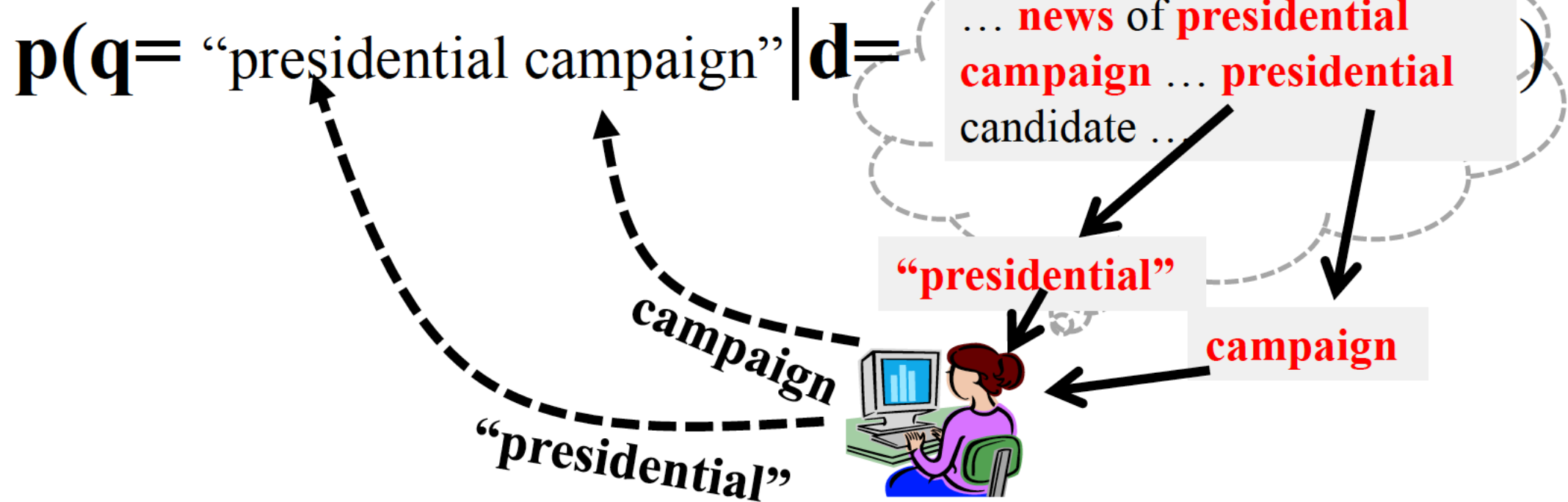
Probabilistic Retrieval Model:
Query Likelihood

Dr. Saeed Ul Hassan
Information Technology University

Probabilistic Retrieval Model: Query Likelihood

Query Generation by Sampling Words from Doc

$$p(q= \text{"presidential campaign"} | d=$$

… **news** of **presidential campaign** … **presidential** candidate …

"**presidential**"

campaign

campaign

"**presidential**"

If the user is **thinking of this doc**,
how likely would she **pose this query**?

Unigram Query Likelihood

$$p(q = \text{"presidential campaign"} | d =$$

… **news** of **presidential campaign** … **presidential** candidate …

$$= p(\text{"presidential "} | d) * p(\text{"campaign "} | d)$$

$$= \frac{c(\text{" presidential"}, d)}{|d|} * \frac{c(\text{" campaign"}, d)}{|d|}$$

Assumption:
Each query word is generated independently

Does Query Likelihood Make Sense?

$$p(q = "presidential\ campaign" \,|\, d) = \frac{c("presidential", d)}{|d|} * \frac{c("campaign", d)}{|d|}$$

p(q|d4= … news of **presidential campaign presidential** candidate … **)** $= \frac{2}{|d4|} * \frac{1}{|d4|}$

p(q|d3= … news of **presidential campaign** … **)** $= \frac{1}{|d3|} * \frac{1}{|d3|}$

p(q|d2= … news about organic food **campaign**… **)** $= \frac{0}{|d2|} * \frac{1}{|d2|} = 0$

d4> d3 > d2 as we expected

Try a Different Query?

q = "**presidential campaign** update"

$$p(q|d4= \boxed{\text{... news of \textbf{presidential campaign} ... \textbf{presidential} candidate ...}}) = \frac{2}{|d4|} * \frac{1}{|d4|} * \frac{0}{|d4|} = 0!$$

$$p(q|d3= \boxed{\text{... news of \textbf{presidential campaign} ...}}) = \frac{1}{|d3|} * \frac{1}{|d3|} * \frac{0}{|d3|} = 0!$$

$$p(q|d2= \boxed{\text{... news about organic food \textbf{campaign}...}}) = \frac{0}{|d2|} * \frac{1}{|d2|} * \frac{0}{|d2|} = 0$$

What assumption has caused this problem? How do we fix it?

Improved Model: Sampling Words from a **Doc Model**

How likely would we observe **this query** from **this doc model**?

$$p(q= \text{``presidential campaign''} | d=)$$

… **news** of **presidential campaign** … **presidential** candidate …

"presidential"

"campaign"

"presidential"

"update"

"presidential"

campaign

update

…
presidential 0.2
campaign 0.1
news 0.01
candidate 0.02
…
update 0.00001
…
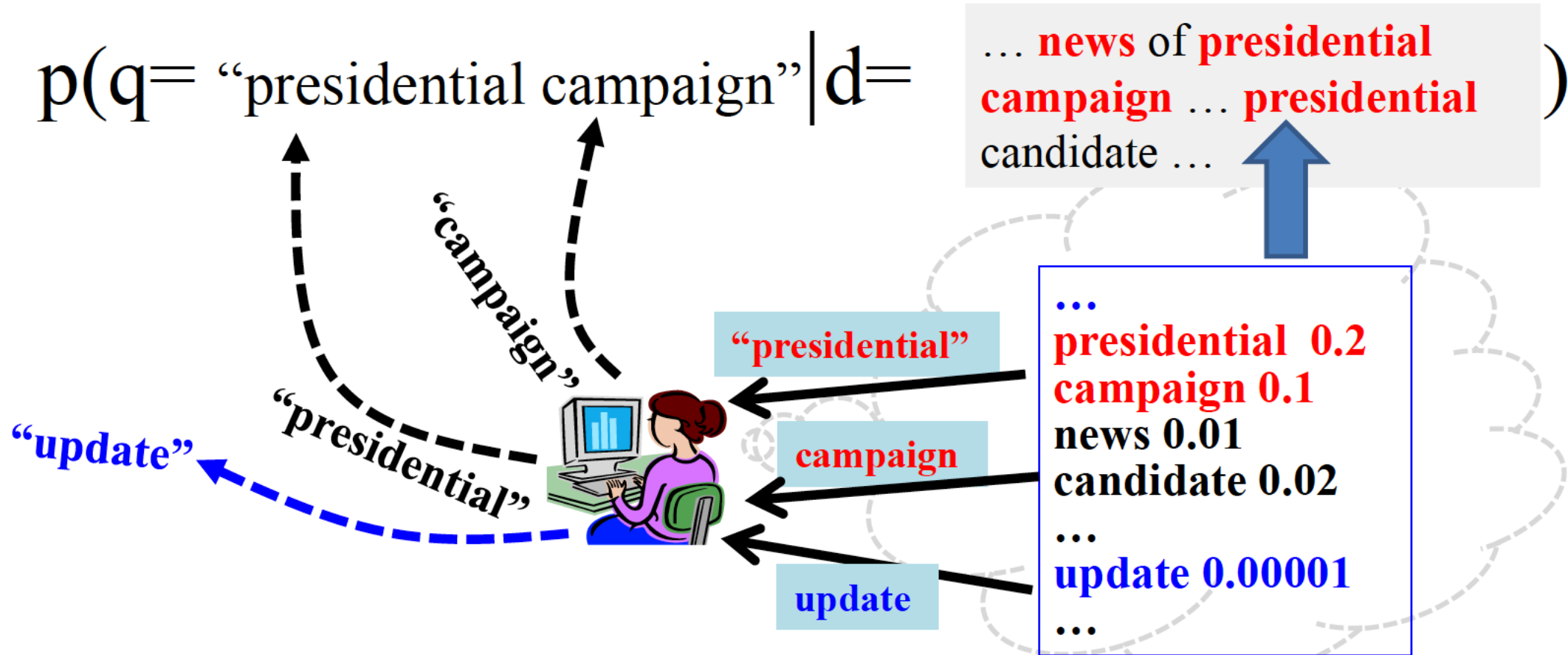
Computation of Query Likelihood

Document

d1

Text mining paper

d2

Food nutrition paper

Document LM

p(w|d1)

...
text 0.2
mining 0.1
association 0.01
clustering 0.02
...
food 0.00001
...

p(w|d2)

...
food 0.25
nutrition 0.1
healthy 0.05
diet 0.02
...

Query q =

"data mining algorithms"

p("data mining alg"|d1)
= p("data"|d1)
× p("mining"|d1)
× p("alg"|d1)

p("data mining alg"|d2)
= p("data"|d2)
× p("mining"|d2)
× p("alg"|d2)

Summary: Ranking based on Query Likelihood

$$q = w_1 w_2 \ldots w_n \qquad\qquad p(q \mid d) = p(w_1 \mid d) \times \ldots \times p(w_n \mid d)$$

$$f(q,d) = \log p(q \mid d) = \sum_{i=1}^{n} \log p(w_i \mid d) = \sum_{w \in V} c(w,q) \log p(w \mid d)$$

Document language model

Retrieval problem ➜ Estimation of $p(w_i \mid d)$

Different estimation methods ➜ different ranking functions