

67-300 SEARCH ENGINES

LANGUAGE MODEL

LECTURER: JOAO PALOTTI (JPALOTTI@ANDREW.CMU.EDU)

27TH MARCH 2016

LECTURE GOALS

- ▶ Missing notes on floating representation in a computer
- ▶ Language Model
- ▶ Discussion on Homework
- ▶ Implementation Part (IPython notebook) - If we have time

FLOATING PRECISION AND LOGARITHM

LOG TRANSFORMATION

- ▶ Floating-point numbers are represented in base 2 fraction.
 - ▶ $0.125 \Rightarrow 1/10 + 2/100 + 5/1000$ (human representation)
 - ▶ $0.125 \Rightarrow 0/2 + 0/4 + 1/8$ (computer representation)
- ▶ Not precise, best approximation with 53 bits for precision
- ▶ How good is the human representation/precision for $1/3$?
 - ▶ 0.3?
 - ▶ 0.333?
 - ▶ 0.33333333?

PRECISION LIMITED

```
[In [1]: 1./3
Out[1]: 0.3333333333333333

[In [2]: print "%.100f" % (1./3)
0.3333333333333333331482961625624739099293947219848632812500000000000000

[In [3]: 1/3. > 0.33333333333333332
Out[3]: True

[In [4]: 1/3. > 0.3333333333333333332
Out[4]: False

[In [5]: 1/3. == 0.33333333333333333
Out[5]: True

[In [6]: 1/3. == 0.33333333333333332
Out[6]: False

[In [7]: 1/3. == 0.333333333333333332
Out[7]: True
```

PRECISION LIMITED

```
[In [1]: 0.1
```

```
Out[1]: 0.1
```

```
[In [2]: 0.1 + 0.1
```

```
Out[2]: 0.2
```

```
[In [3]: 0.1 + 0.1 + 0.1
```

```
Out[3]: 0.30000000000000004
```

```
[In [4]: 0.3 == 0.1 + 0.1 + 0.1
```

```
Out[4]: False
```

PRECISION LIMITED

$$RSV = \log \prod_{x_i=1; q_i=1} \frac{p_i \times (1 - r_i)}{r_i \times (1 - p_i)} = \sum_{x_i=1; q_i=1} \log \frac{p_i \times (1 - r_i)}{r_i \times (1 - p_i)}$$

```
[In [33]: 1e-10 * 1e-10
```

```
Out[33]: 1.00000000000000000001e-20
```

CHANGING FROM PRODUCT TO SUM ALLEVIATE
PRECISION PROBLEMS, AMONG OTHER
ADVANTAGES.

```
[In [34]: 1e-10 * 1e-10 > 1e-20
```

```
Out[34]: True
```

```
[In [35]: math.log(1e-10) + math.log(1e-10) > math.log(1e-20)
```

```
Out[35]: False
```

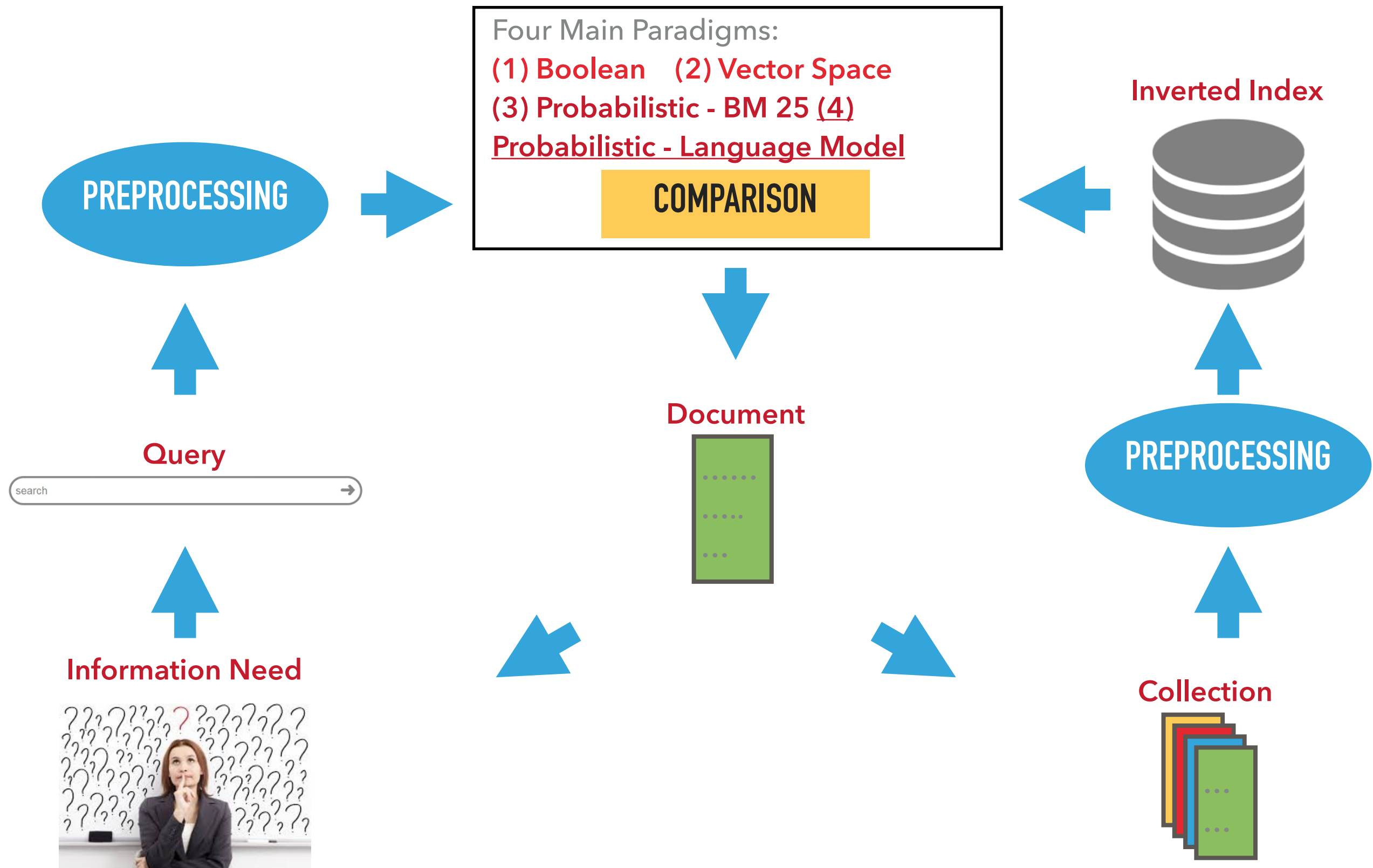
```
[In [36]: math.log(1e-10) + math.log(1e-10) < math.log(1e-20)
```

```
Out[36]: False
```

```
[In [37]: math.log(1e-10) + math.log(1e-10) == math.log(1e-20)
```

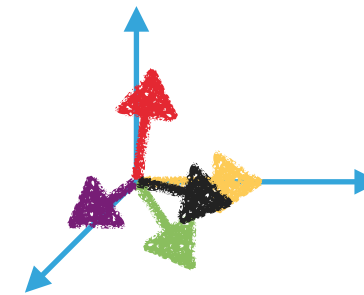
```
Out[37]: True
```

CLASS 5 - LANGUAGE MODEL



RECAP: BIM

FRAMEWORKS RECAP



- ▶ VSM: strong geometric motivation

- ▶ Probabilistic framework:

$$P(R_{d,q} = 1|d, q)$$

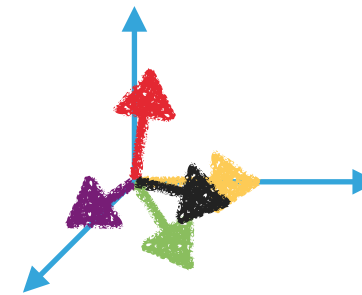
- ▶ Binary Independence Model

document	relevant (R=1)	nonrelevant(R=0)
term present $x_i = 1$	p_i	r_i
term absent $x_i=0$	$1-p_i$	$1-r_i$

$$O(R|q, x) = \frac{P(R = 1|q, x)}{P(R = 0|q, x)} = \sum_i^{|V|} \boxed{\frac{p_i}{r_i}} \times \boxed{\frac{(1 - r_i)}{(1 - p_i)}}$$

FRAMEWORKS RECAP

- ▶ VSM: strong geometric motivation
- ▶ Probabilistic framework:
- ▶ Binary Independence Model



$$P(R_{d,q} = 1|d, q)$$

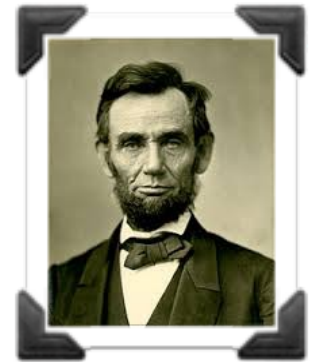
document	relevant (R=1)	nonrelevant(R=0)
term present $x_i = 1$	p_i	r_i
term absent $x_i=0$	$1-p_i$	$1-r_i$

$$O(R|q, x) = \frac{P(R = 1|q, x)}{P(R = 0|q, x)} = \sum_i^{|V|} \boxed{\frac{p_i}{r_i}} \times \boxed{\frac{(1 - r_i)}{(1 - p_i)}}$$

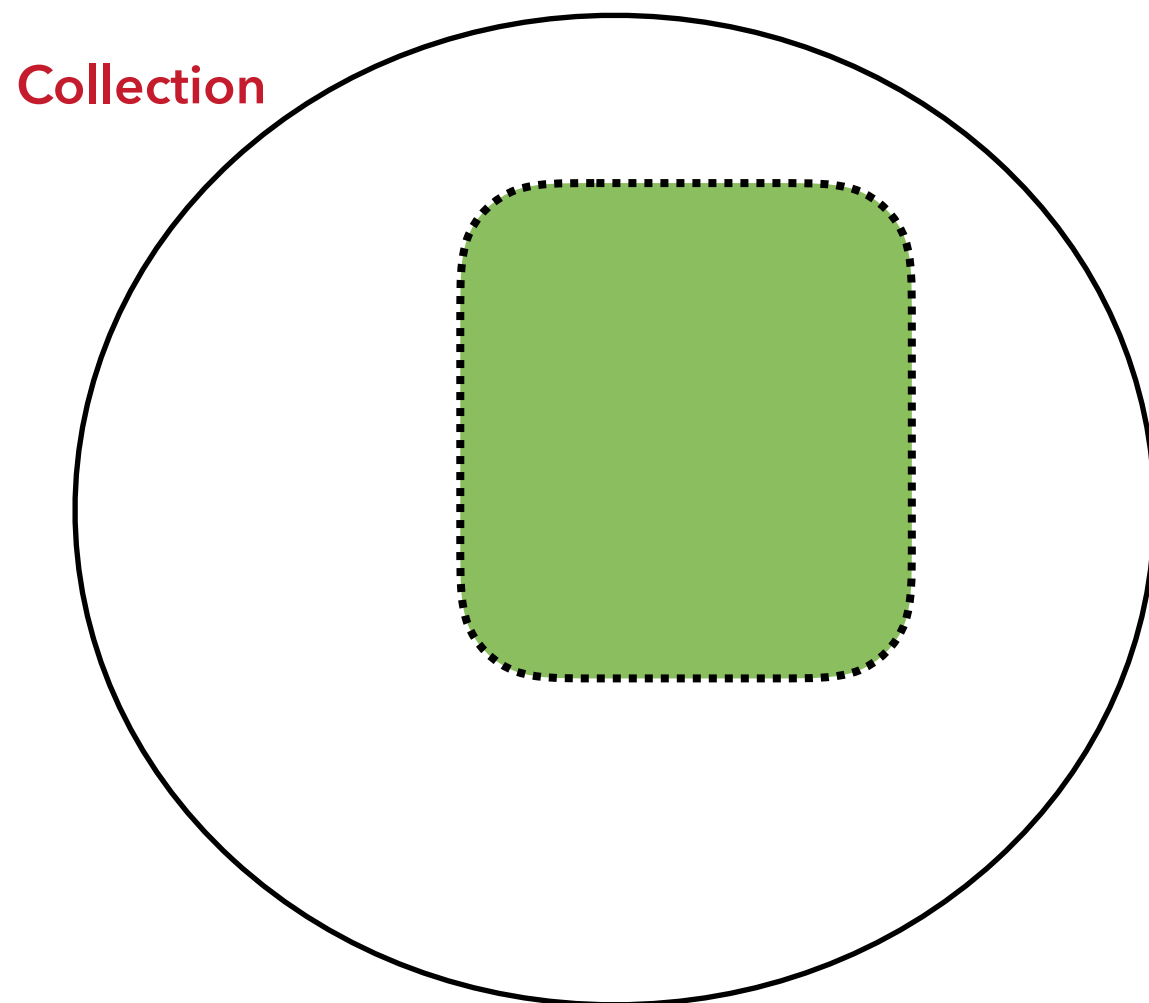
Term i is in this document.
How much certainty are we that this is a relevant doc?

Term i is NOT in this document.
How much certainty are we that this is a relevant doc?

BIM AND LINCOLN

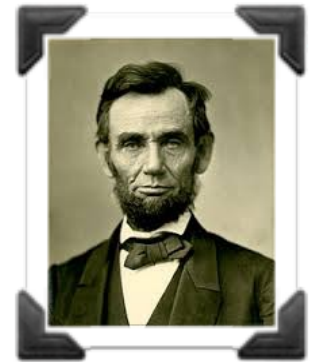


- ▶ Query: "lincoln"

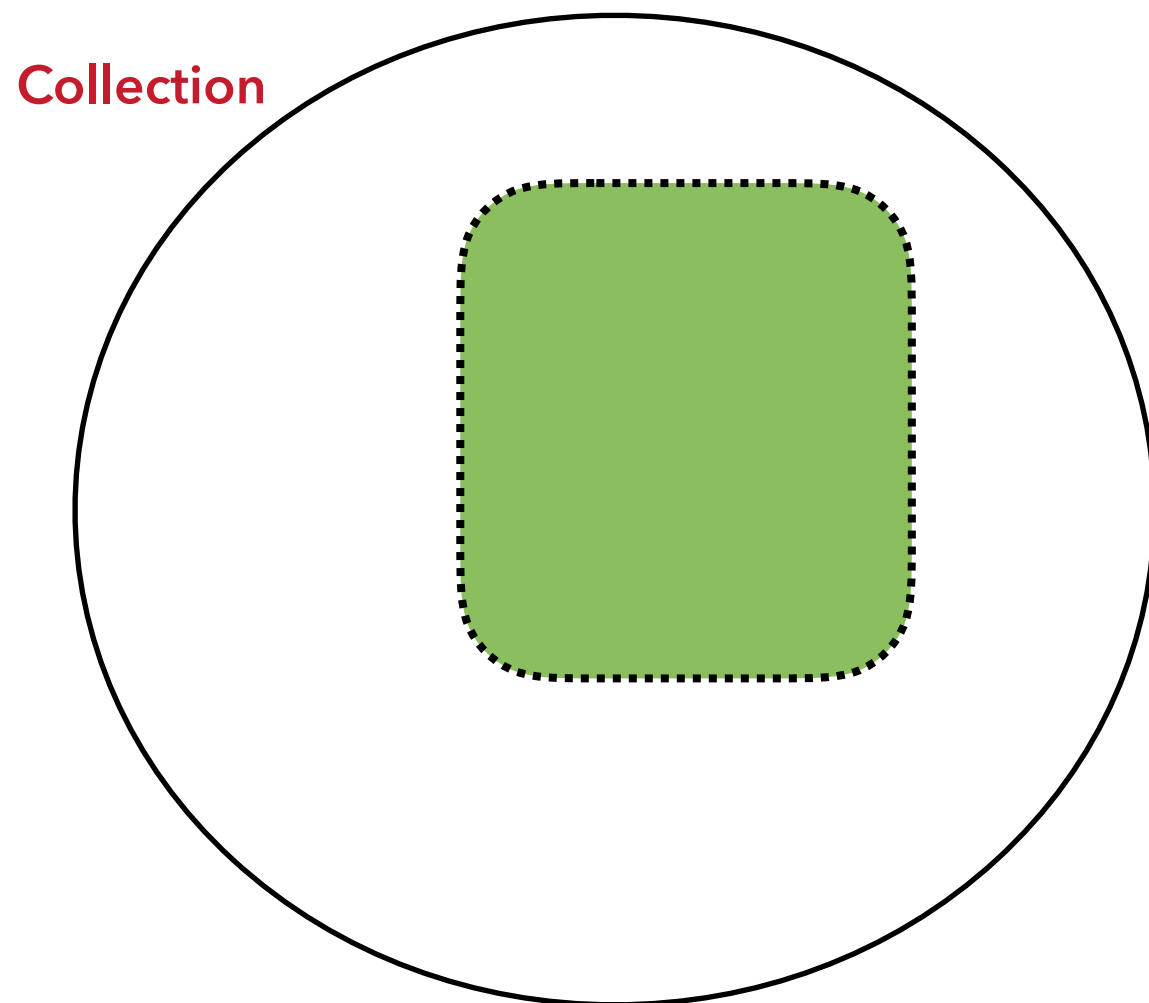


- ▶ Retrieve all documents that contain "lincoln"
- ▶ No values for p_i . Only sort docs by IDF only
- ▶ All document have the same score!

BIM AND LINCOLN



- ▶ Query: "lincoln"



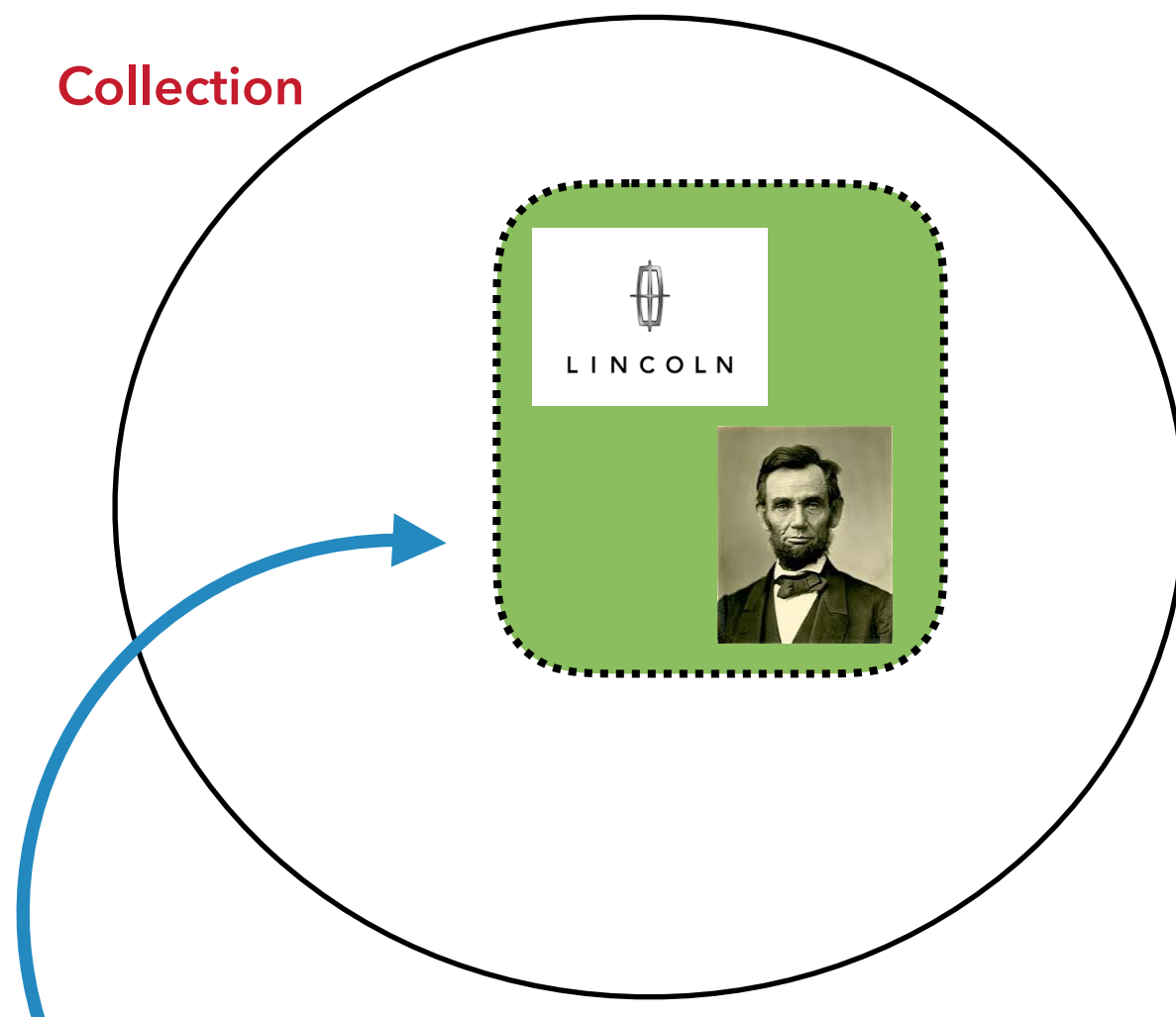
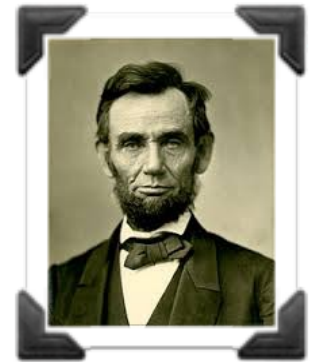
- ▶ Retrieve all documents that contain "lincoln"
- ▶ No values for p_i . Only sort docs by IDF only
- ▶ All document have the same score!

Can you tell me why?

Is it the same output of a Boolean search??

BIM AND LINCOLN

- ▶ Query: "lincoln"



Not all documents are relevant!

- ▶ Retrieve all documents that contain "lincoln"
- ▶ No values for p_i . Only sort docs by IDF only
- ▶ All document have the same score!

Can you tell me why?

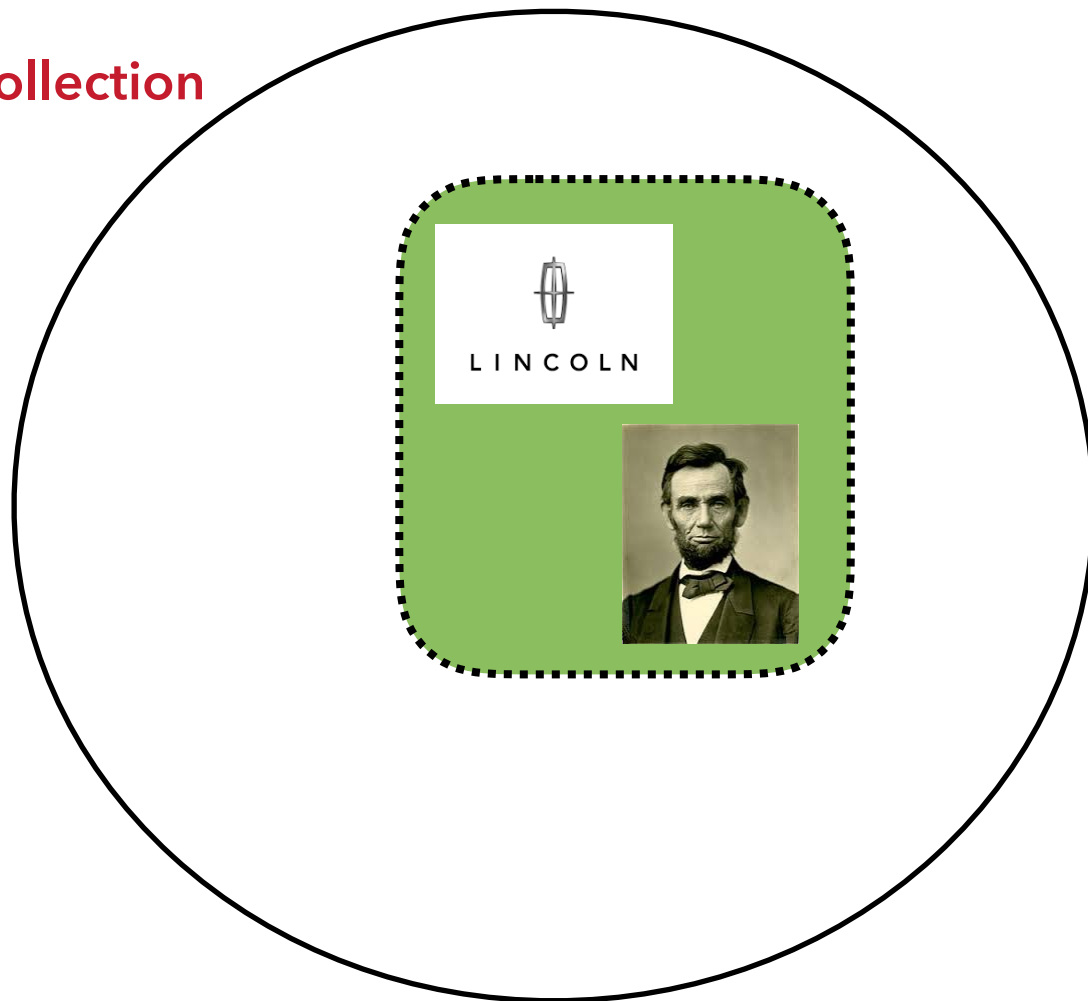
Is it the same output of a Boolean search??

BIM AND LINCOLN

- ▶ Query: "lincoln"

- ▶ User reads some documents and state that he/she wants documents like D_i and D_j

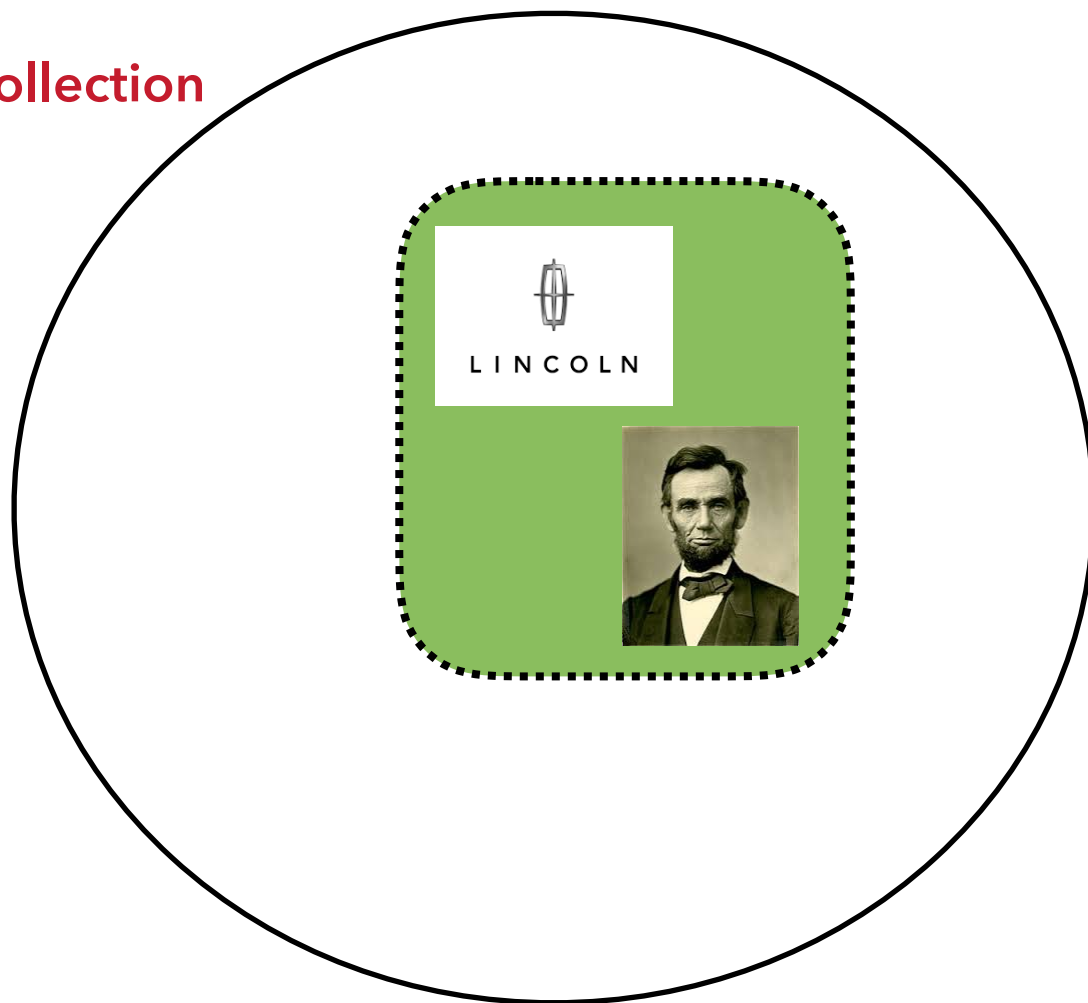
Collection



BIM AND LINCOLN

- ▶ Query: "lincoln"

Collection

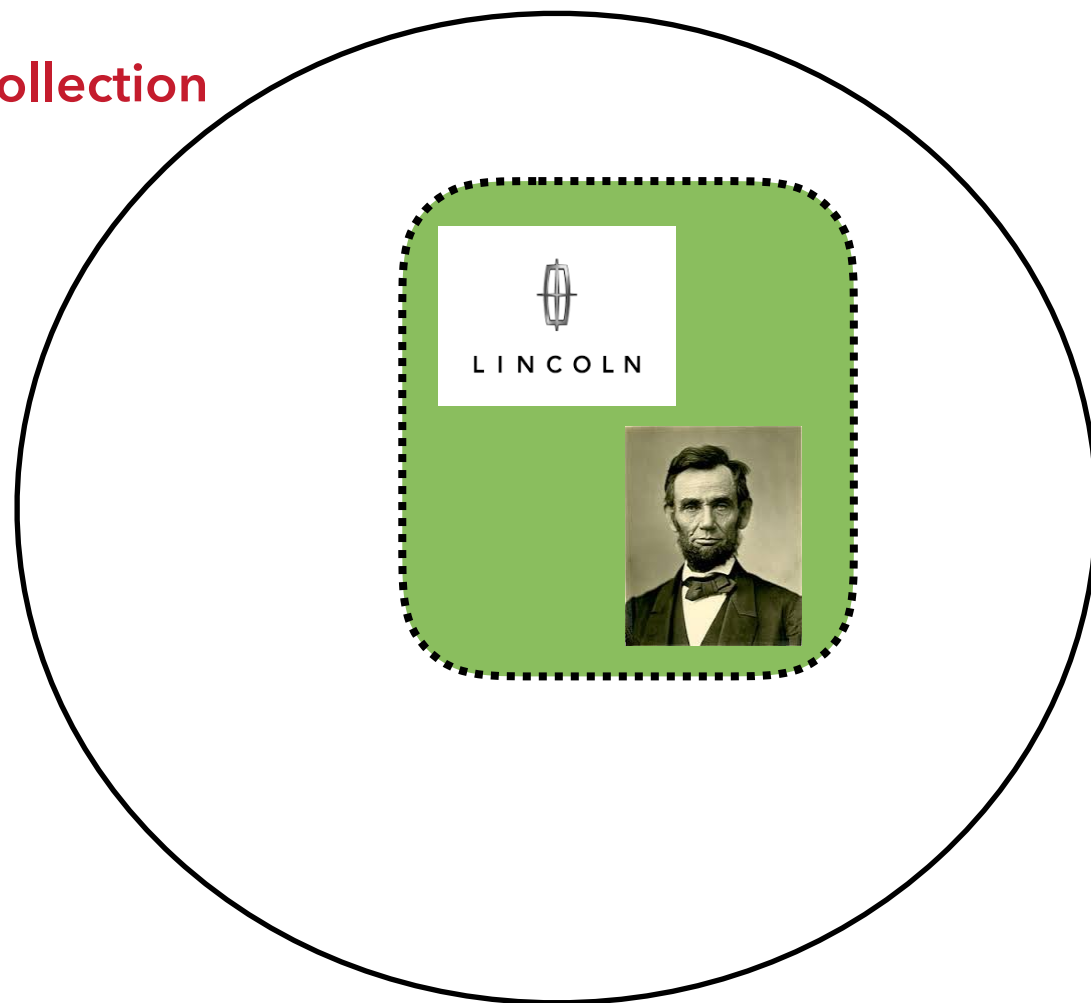


- ▶ User reads some documents and state that he/she wants documents like D_i and D_j
- ▶ Algorithm inspects terms in document D_i, D_j
 - ▶ Terms from the relevant documents: life, bio, gettysburgh...

BIM AND LINCOLN

- ▶ Query: "lincoln"

Collection



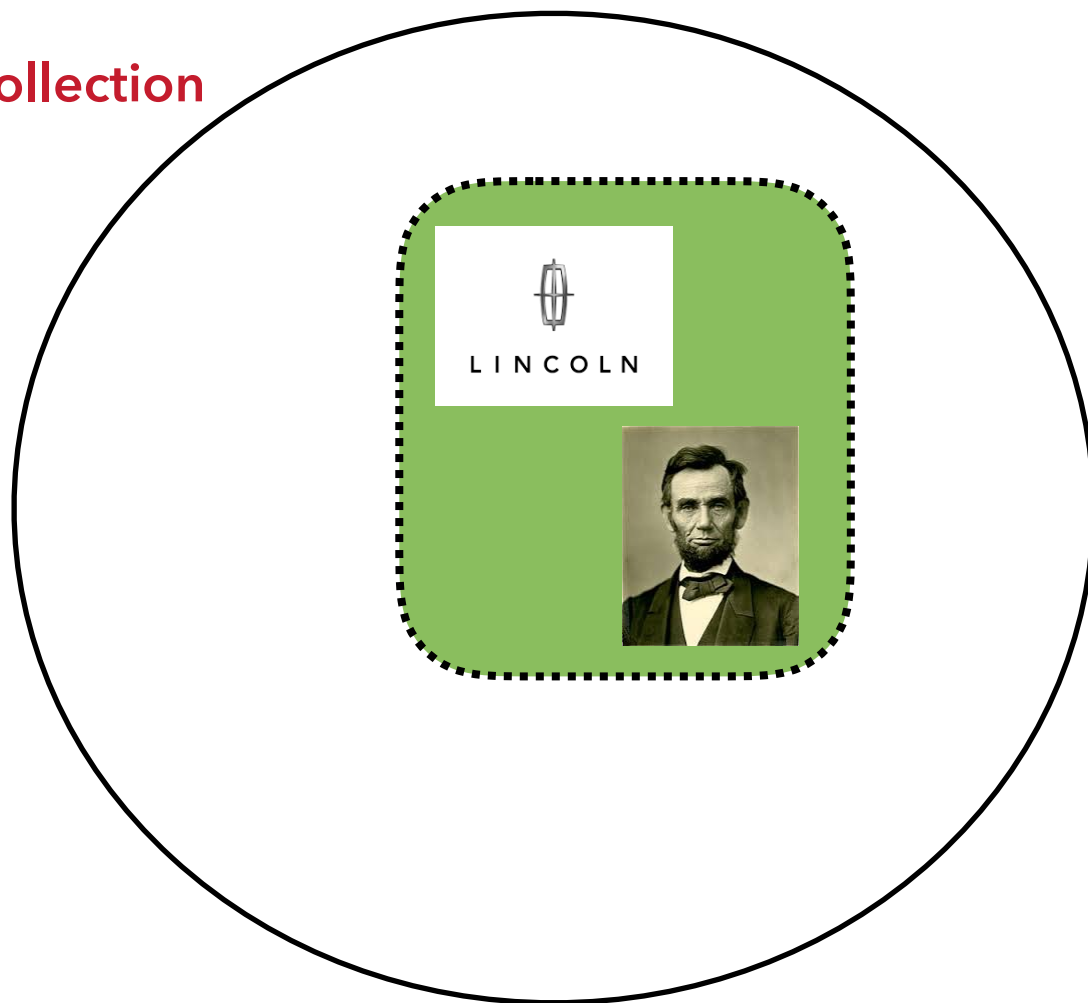
- ▶ User reads some documents and state that he/she wants documents like D_i and D_j
- ▶ Algorithm inspects terms in document D_i, D_j
 - ▶ Terms from the relevant documents: life, bio, gettysburgh...
- ▶ Algorithm inspects terms in all other documents
 - ▶ Terms from the non relevant documents: car, automobile...

BIM AND LINCOLN

- ▶ Query: "lincoln"

- ▶ For every term \underline{t} , what is the likelihood of \underline{t} being present in relevant docs Vs. being present in non relevant docs.

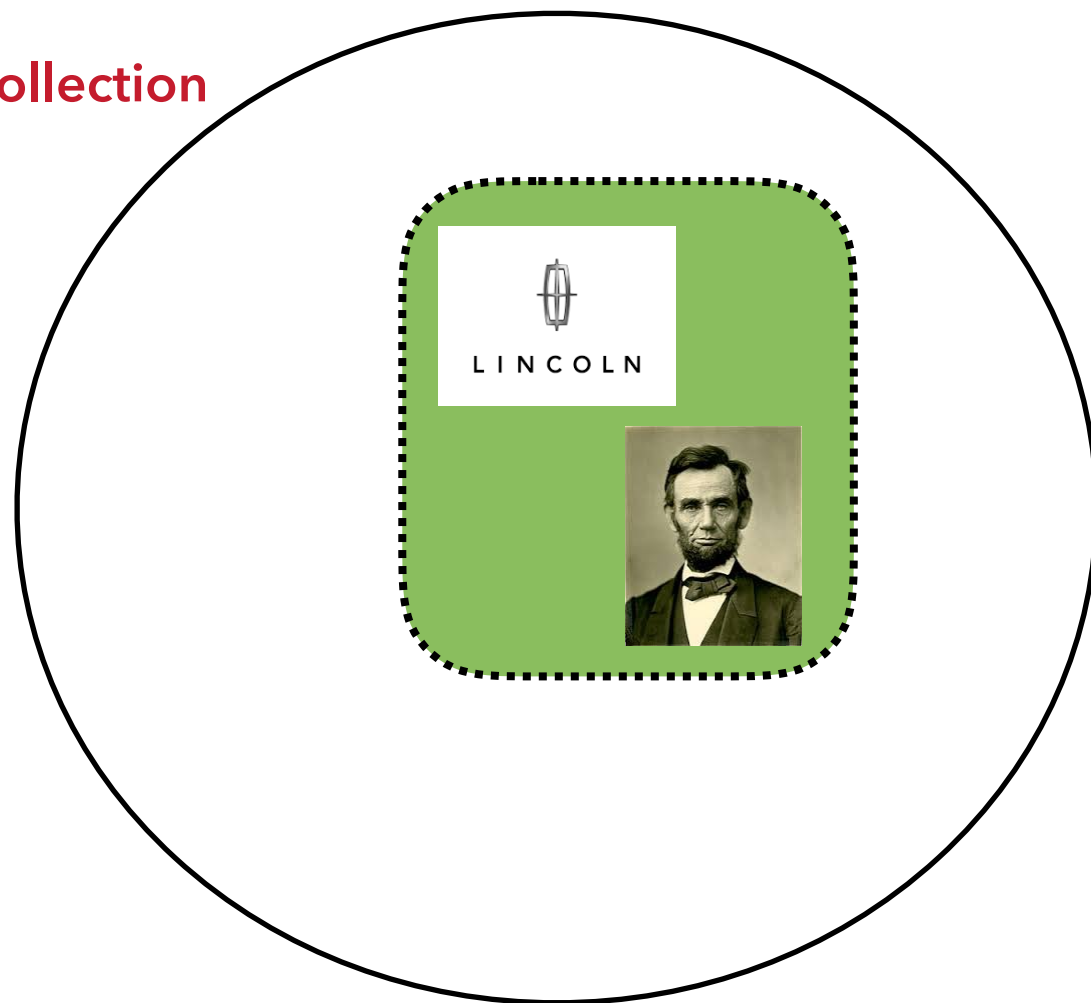
Collection



BIM AND LINCOLN

- ▶ Query: "lincoln"

Collection



- ▶ For every term \underline{t} , what is the likelihood of \underline{t} being present in relevant docs Vs. being present in non relevant docs.
- ▶ Term: "biography"
 - ▶ What is $P(R = 1 \mid \text{"biography"})$?
 - ▶ What is $P(R = 0 \mid \text{"biography"})$?
- ▶ Term: "industry"
 - ▶ What is $P(R = 1 \mid \text{"industry"})$?
 - ▶ What is $P(R = 0 \mid \text{"industry"})$?

BM25

$$P(R_{d,q} = 1|d, q)$$

- ▶ Empirical way to instantiate the probabilistic framework
- ▶ Removed binary assumption from BIM

$$RSV_d = \sum_{t \in q} \log \left[\frac{N}{df_t} \right] \cdot \frac{(k_1 + 1)tf_{td}}{k_1((1 - b) + b \times (L_d/L_{ave})) + tf_{td}} \cdot \frac{(k_3 + 1)tf_{tq}}{k_3 + tf_{tq}}$$

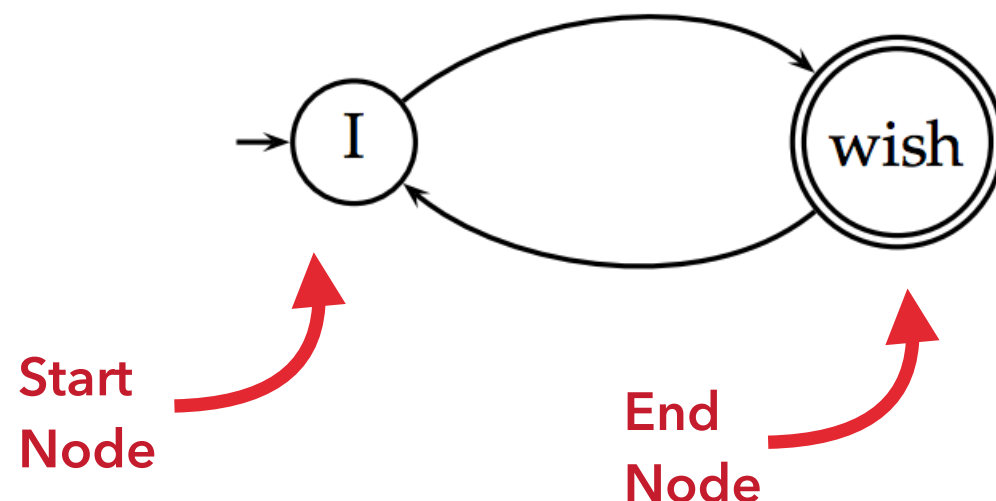
Diagram illustrating the BM25 formula components:

- IDF** (Inverse Document Frequency) is represented by the term $\log \left[\frac{N}{df_t} \right]$.
- Document Length** is represented by the term $k_1((1 - b) + b \times (L_d/L_{ave}))$ in the denominator of the first fraction.
- TF** (Term Frequency) is represented by the term tf_{td} in the numerator of the first fraction and tf_{tq} in the numerator of the second fraction.

LANGUAGE MODELS

LANGUAGE MODELS

- ▶ Statistical natural language processing approach.
- ▶ Generative model:
 - ▶ Let M_d be the language model define by this finite automaton:



I wish

I wish I wish

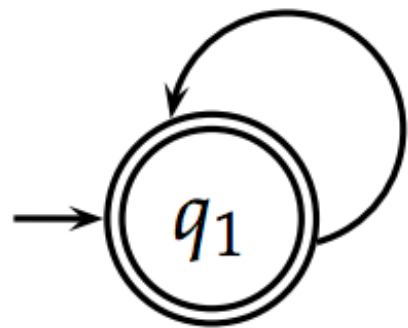
I wish I wish I wish

I wish I wish I wish.....

I wish I?

GENERATIVE MODEL

- ▶ How about this other M_d :



$$P(\text{STOP}|q_1) = 0.2$$

the	0.2
a	0.1
frog	0.01
toad	0.01
said	0.03
likes	0.02
that	0.04
...	...

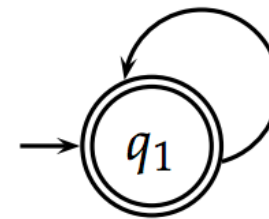
Probabilities sum to 1

$$\sum_{t \in L} P(t) = 1$$

- ▶ $P(\text{frog said that toad likes frog})$

GENERATIVE MODEL

- ▶ How about this other M_d :



$$P(\text{STOP}|q_1) = 0.2$$

the	0.2
a	0.1
frog	0.01
toad	0.01
said	0.03
likes	0.02
that	0.04
...	...

- ▶ We can calculate the probability of seen the sequence: "frog said that toad likes frog".
 - ▶ $P(\text{frog said that toad likes frog})$: 0.00000000000001573
 - ▶ $P(\text{Today is Monday})$: 0.000001
 - ▶ $P(\text{The capital of Qatar is Doha})$: 0.00000012345

GENERAL WAY TO CALCULATE THE PROBABILITY OF A SEQUENCE

- ▶ Chain rule tell us that we need to calculate the following:

$$P(w_1 w_2 w_3 \dots w_N) = ?$$

GENERAL WAY TO CALCULATE THE PROBABILITY OF A SEQUENCE

- ▶ Chain rule tell us that we need to calculate the following:

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1 w_2 \dots w_{n-1})$$

GENERAL WAY TO CALCULATE THE PROBABILITY OF A SEQUENCE

- ▶ Chain rule tell us that we need to calculate the following:

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1 w_2 \dots w_{n-1})$$

Conditional dependences



GENERAL WAY TO CALCULATE THE PROBABILITY OF A SEQUENCE

- ▶ Chain rule tell us that we need to calculate the following:

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1 w_2 \dots w_{n-1})$$

Conditional dependences

- ▶ How can we get rid of these conditional dependences?

GENERAL WAY TO CALCULATE THE PROBABILITY OF A SEQUENCE

- ▶ Chain rule tell us that we need to calculate the following:

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2) \dots P(w_n|w_1 w_2 \dots w_{n-1})$$

Conditional dependences

- ▶ How can we get rid of these conditional dependences?
 - ▶ Again: assuming some degree of independence for terms in a text

Example: $P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2)P(w_3) \dots P(w_n)$

PROBABILITY OF GENERATING A TEXT

- ▶ Unigram Language model:

No Conditional dependences at all

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2)P(w_3) \dots P(w_n)$$

- ▶ Bigram Language model:

Dependence restrict the last term

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_2) \dots P(w_n|w_{n-1})$$

- ▶ Trigram Language model:

Dependence restrict the last two terms

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_2w_1) \dots P(w_n|w_{n-1}w_{n-2})$$

- ▶ N-gram Language model:

Dependence restrict the last n-1 terms

$$P(w_1 w_2 w_3 \dots w_N) = P(w_1)P(w_2|w_1)P(w_3|w_2w_1) \dots P(w_n|w_{n-1}w_{n-2} \dots w_{n-N+1})$$

Which one to use?

IN INFORMATION RETRIEVAL...

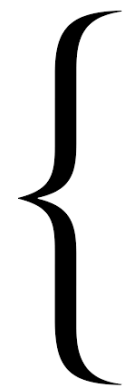
- ▶ Most of the time we use the Unigram Language Model. Why?

Simple enough and powerful enough for this task

IN INFORMATION RETRIEVAL...

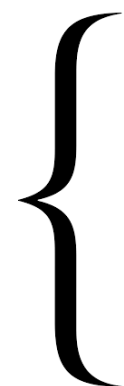
- Most of the time we use the Unigram Language Model. Why?

LM ϕ_1



qatar 0.01
location 0.002
south 0.003
arab 0.0009
...
nutrition 0.00002
food 0.00000001

LM ϕ_2

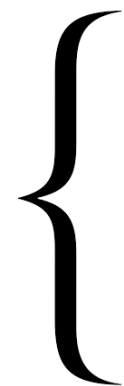


qatar 0.00000003
location 0.0001
south 0.00005
arab 0.003
...
nutrition 0.001
food 0.01

IN INFORMATION RETRIEVAL...

- Most of the time we use the Unigram Language Model. Why?

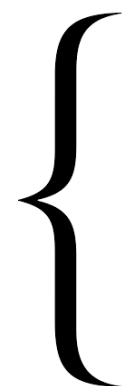
LM ϕ_1



qatar 0.01
location 0.002
south 0.003
arab 0.0009
...
nutrition 0.00002
food 0.00000001

Word distribution for a document about the location of qatar

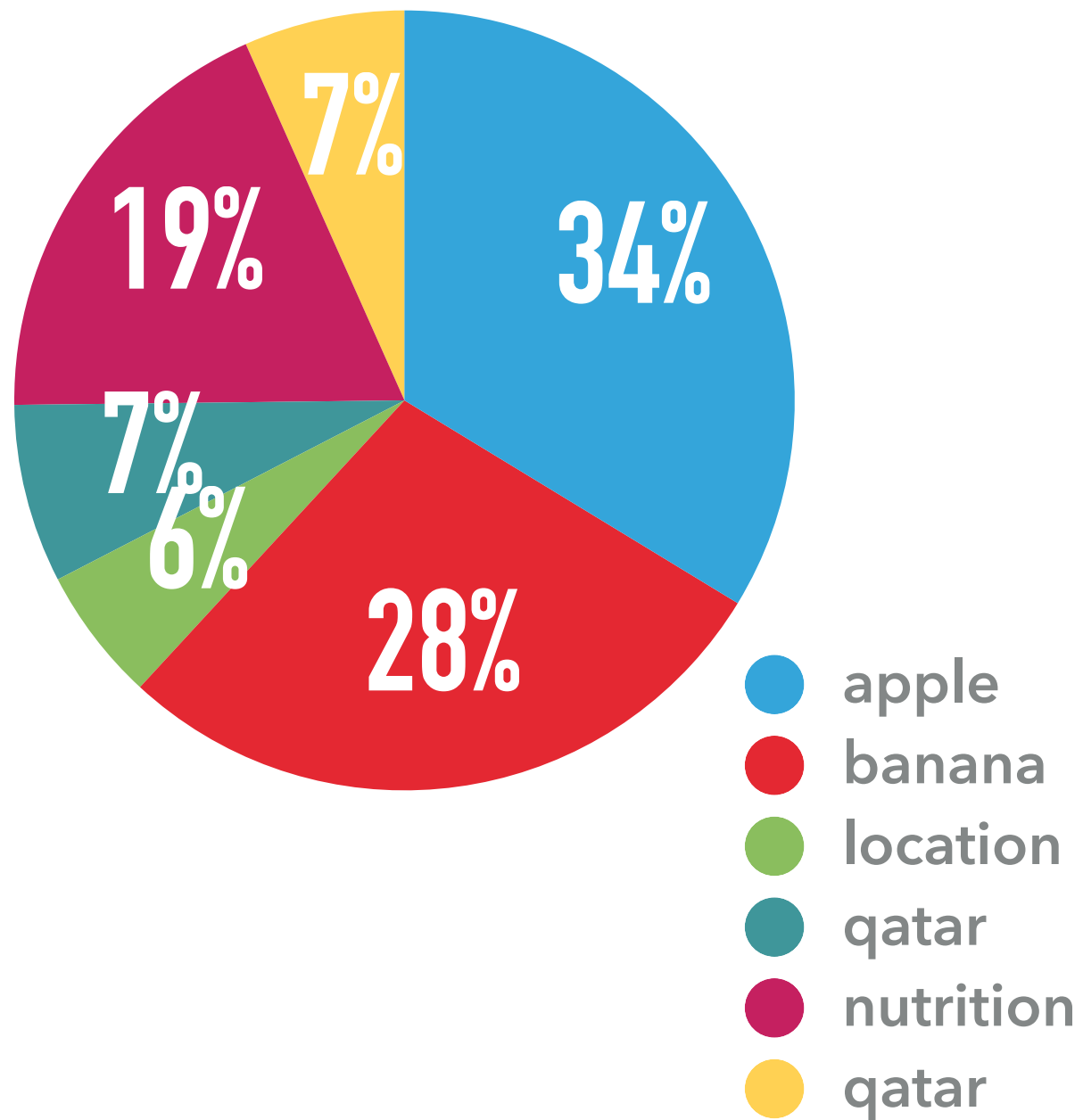
LM ϕ_2



qatar 0.00000003
location 0.0001
south 0.00005
arab 0.003
...
nutrition 0.001
food 0.01

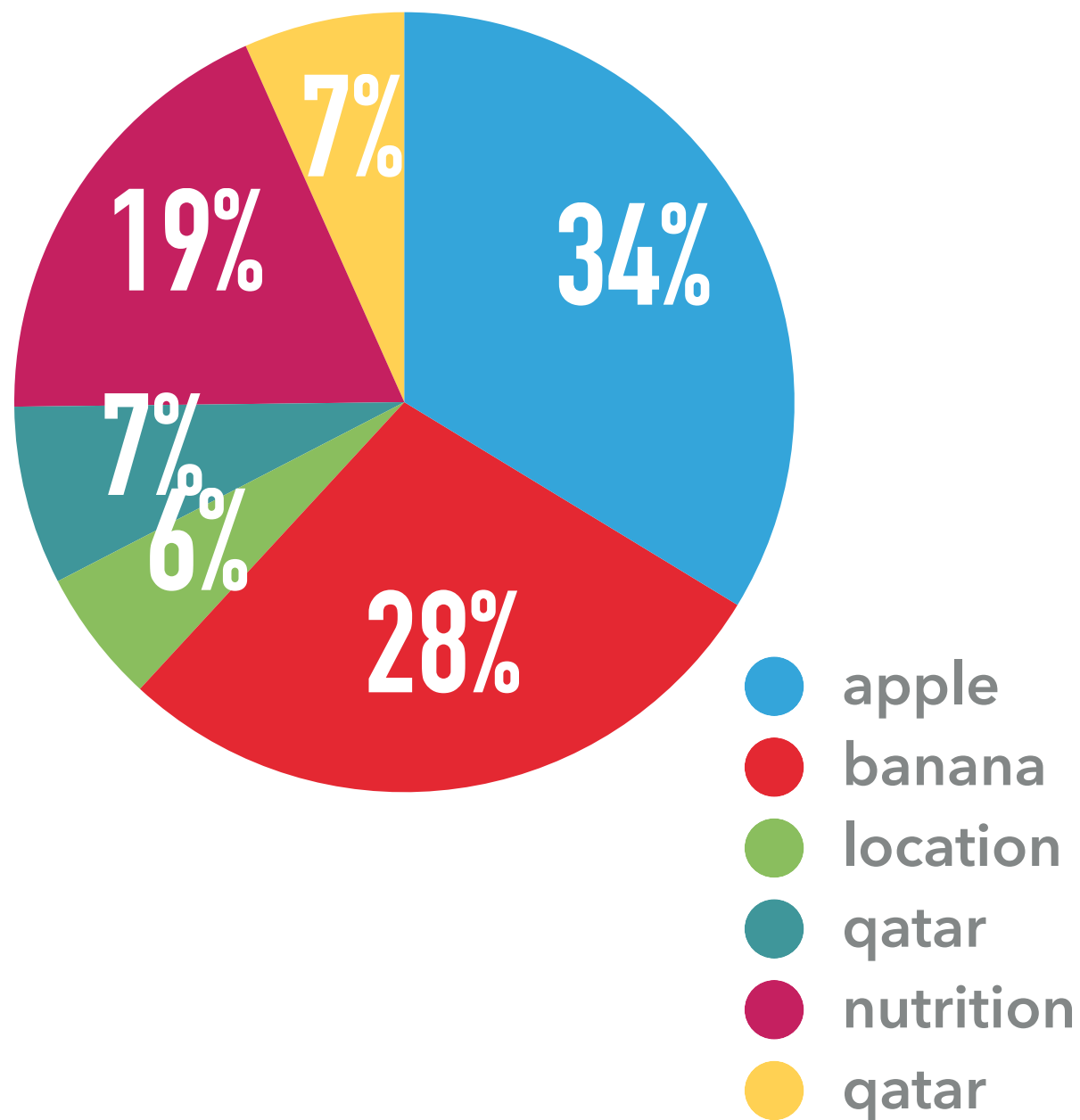
Word distribution for a document about health food

GENERATING TEXT IN UNIGRAM LM



► Throw a dice...

GENERATING TEXT IN UNIGRAM LM



- ▶ Throw a dice...
 - ▶ Generates word W_1
- ▶ Throw another dice...
 - ▶ Generates word W_2
- ▶ Throw a dice again...
 - ▶ Generates word W_3

FROM TEXT TO PROBABILITIES...

Qatar is a sovereign country located in Western **Asia**, occupying the small **Qatar Peninsula** on the northeastern coast of the **Arabian Peninsula**. Its sole land border is with Saudi **Arabia** to the south, with the rest of its territory surrounded by the **Persian Gulf**. A strait in the **Persian Gulf** separates **Qatar** from the nearby island country of Bahrain, as well as sharing maritime borders with the United **Arab** Emirates and Iran.



► $P(\text{"qatar"}) = 3/79$

► $P(\text{"."}) = 3/79$

► $P(\text{"gulf"}) = 2/79$

► $P(\text{"arab"}) = 3/79$



► $P(\text{"asia"}) = 1/79$

► $P(\text{"land"}) = 1/79$

► $P(\text{"persian"}) = 2/79$

► $P(\text{"peninsula"}) = 2/79$



FROM PROBABILITIES TO TEXT...

▶ $P(\text{"qatar"}) = 3/79$

▶ $P(\text{"."}) = 3/79$

▶ $P(\text{"gulf"}) = 2/79$

▶ $P(\text{"arab"}) = 3/79$

▶ $P(\text{"asia"}) = 1/79$

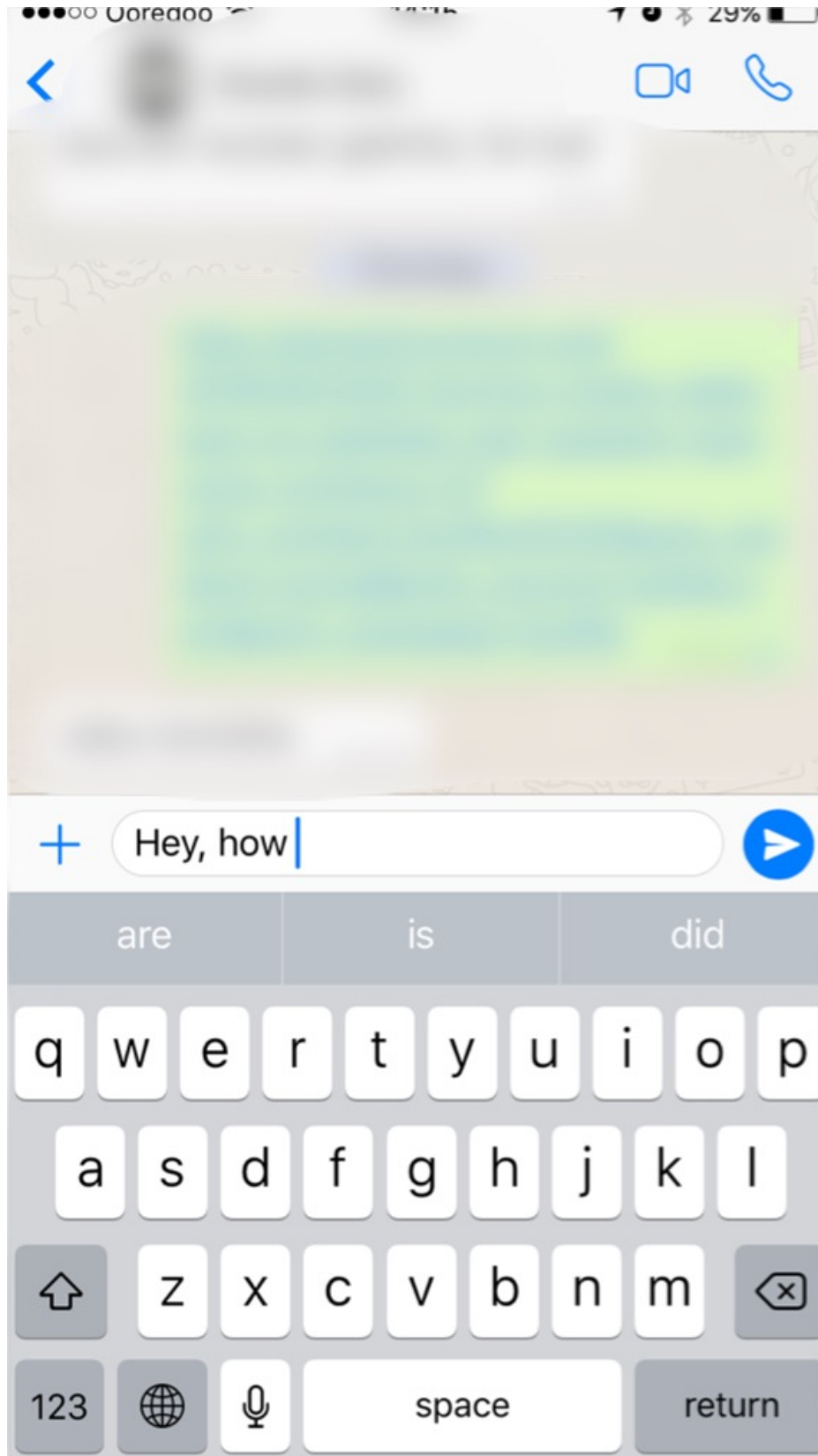
▶ $P(\text{"land"}) = 1/79$

▶ $P(\text{"persian"}) = 2/79$

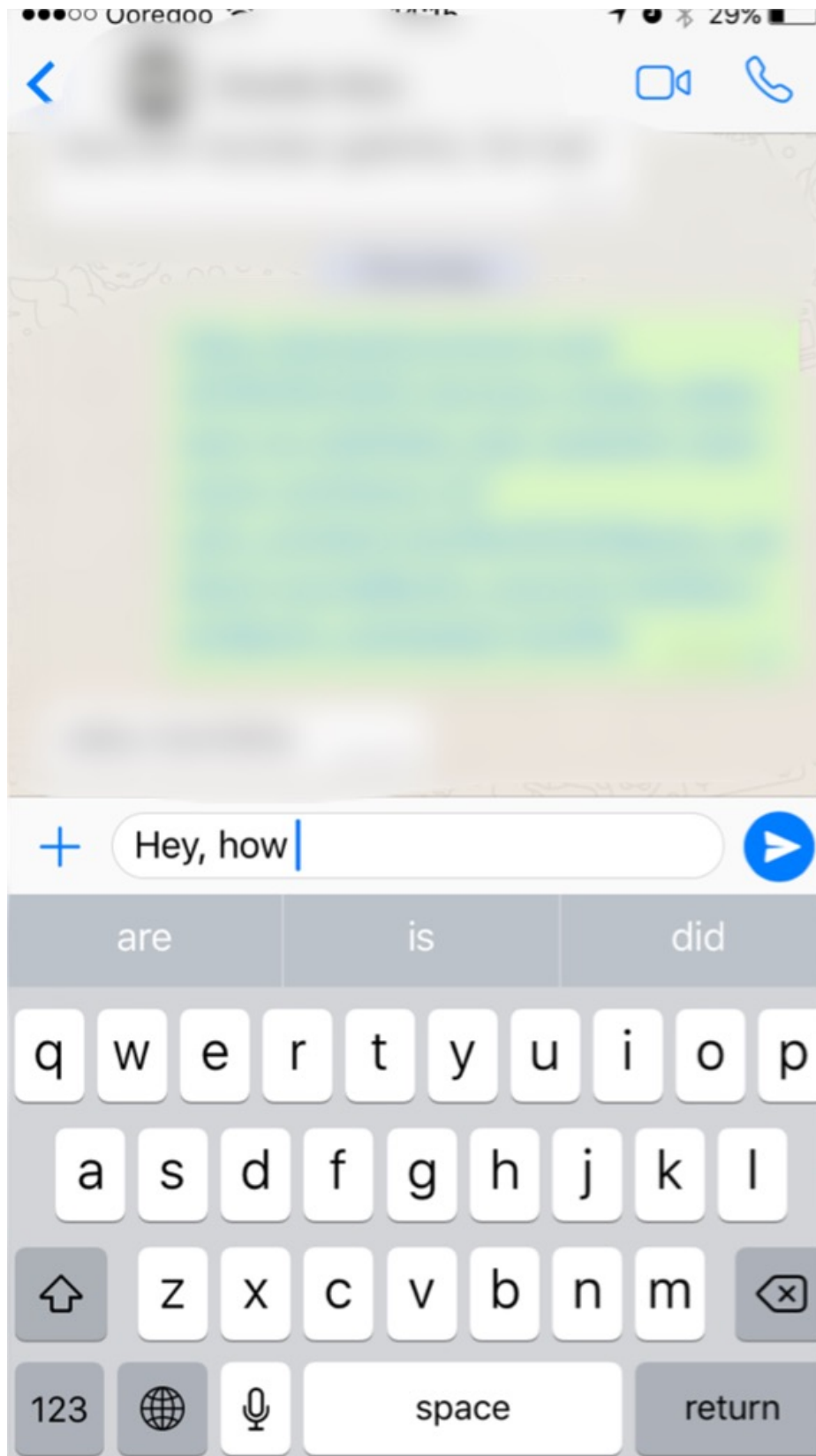
▶ $P(\text{"peninsula"}) = 2/79$



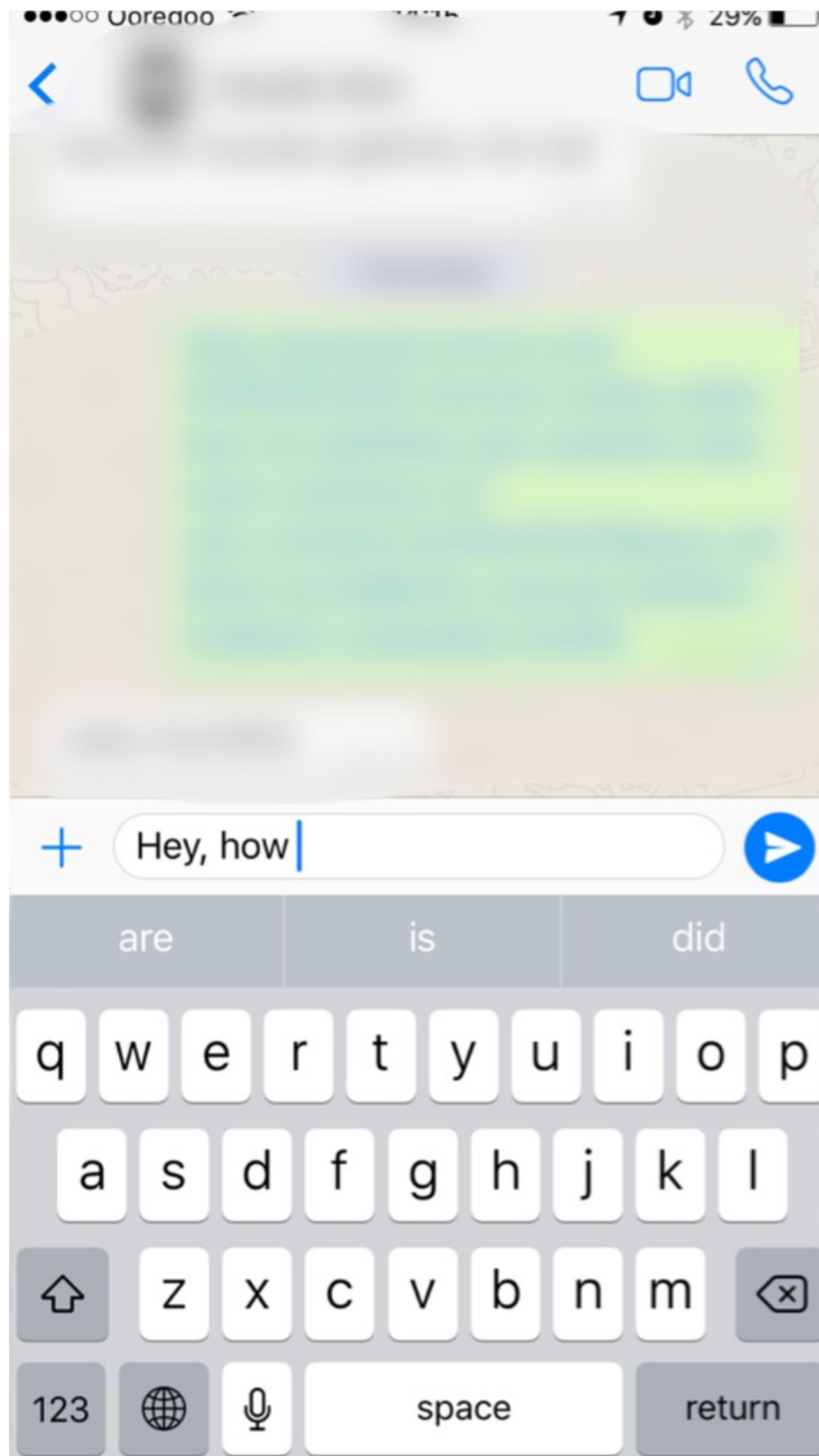
Gulf United as surrounded . well **Qatar** is **Persian** a of nearby **Peninsula** rest coast sovereign **Arabian** country south located in Western **Asia**, occupying **Persian** the small **Qatar** on the northeastern of the **Peninsula** . Its sole land border is Emirates with **Arabia** to the . Saudi , with the of its territory by the A strait in the **Gulf** **Qatar** from the island country , as separates Iran maritime Bahrain borders with the **Arab** sharing and



- ▶ How can we generate these suggestions?



- ▶ How can we generate these suggestions?
- ▶ E.g. Top 3 term T that maximize:
 - ▶ $P(T \mid \text{"Hey, how"})$
 - ▶ $P(\text{"are"} \mid \text{"Hey, how"}) > P(\text{"house"} \mid \text{"Hey, how"})$
 - ▶ $P(\text{"are"} \mid \text{"Hey, how"}) > P(\text{"do"} \mid \text{"Hey, how"})$

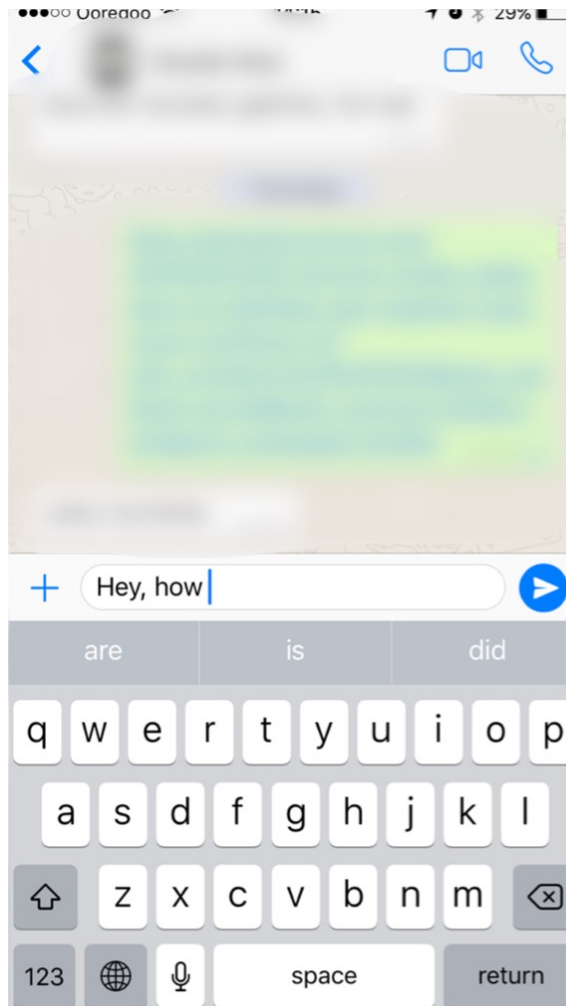


- ▶ How can we generate these suggestions?
- ▶ E.g. Top 3 term T that maximize:
 - ▶ $P(T \mid \text{"Hey, how"})$
 - ▶ $P(\text{"are"} \mid \text{"Hey, how"}) > P(\text{"house"} \mid \text{"Hey, how"})$
 - ▶ $P(\text{"are"} \mid \text{"Hey, how"}) > P(\text{"do"} \mid \text{"Hey, how"})$

Unigrams are not enough
for this other task...

GENERATING TEXT IN BIGRAM LM

- ▶ $P(\text{"qatar"} \mid \langle \text{bos} \rangle) = 0.00001$
- ▶ $P(\text{"you"} \mid \langle \text{bos} \rangle) = 0.01$
- ▶ Throw a dice for the first word:
- ▶ Generates word W_1



GENERATING TEXT IN BIGRAM LM

▶ $P(\text{"qatar"} \mid \langle \text{bos} \rangle) = 0.000001$

▶ $P(\text{"you"} \mid \langle \text{bos} \rangle) = 0.01$

...

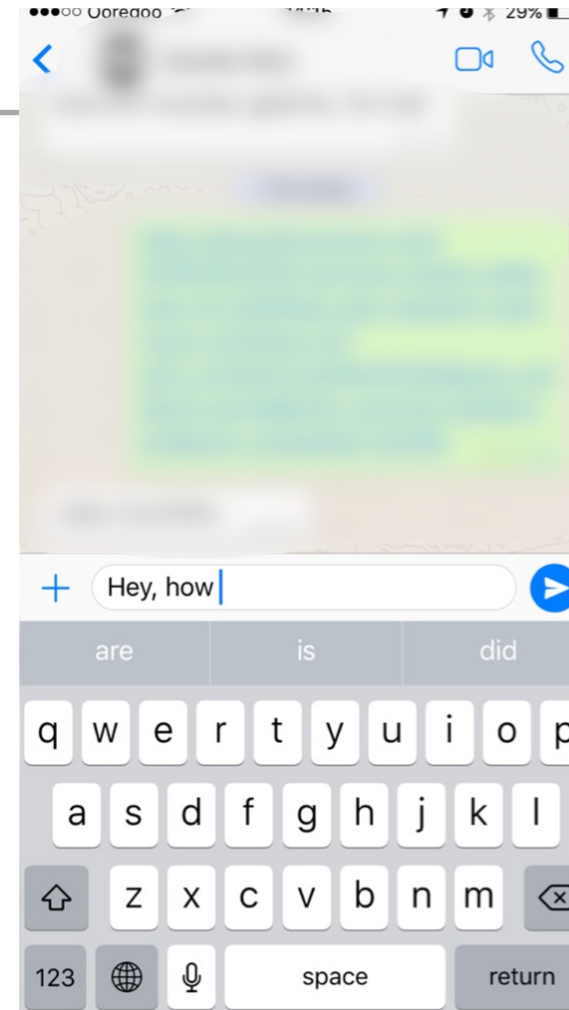
▶ $P(\text{"are"} \mid \text{"you"}) = 0.02$

▶ $P(\text{"is"} \mid \text{"you"}) = 0.000001$

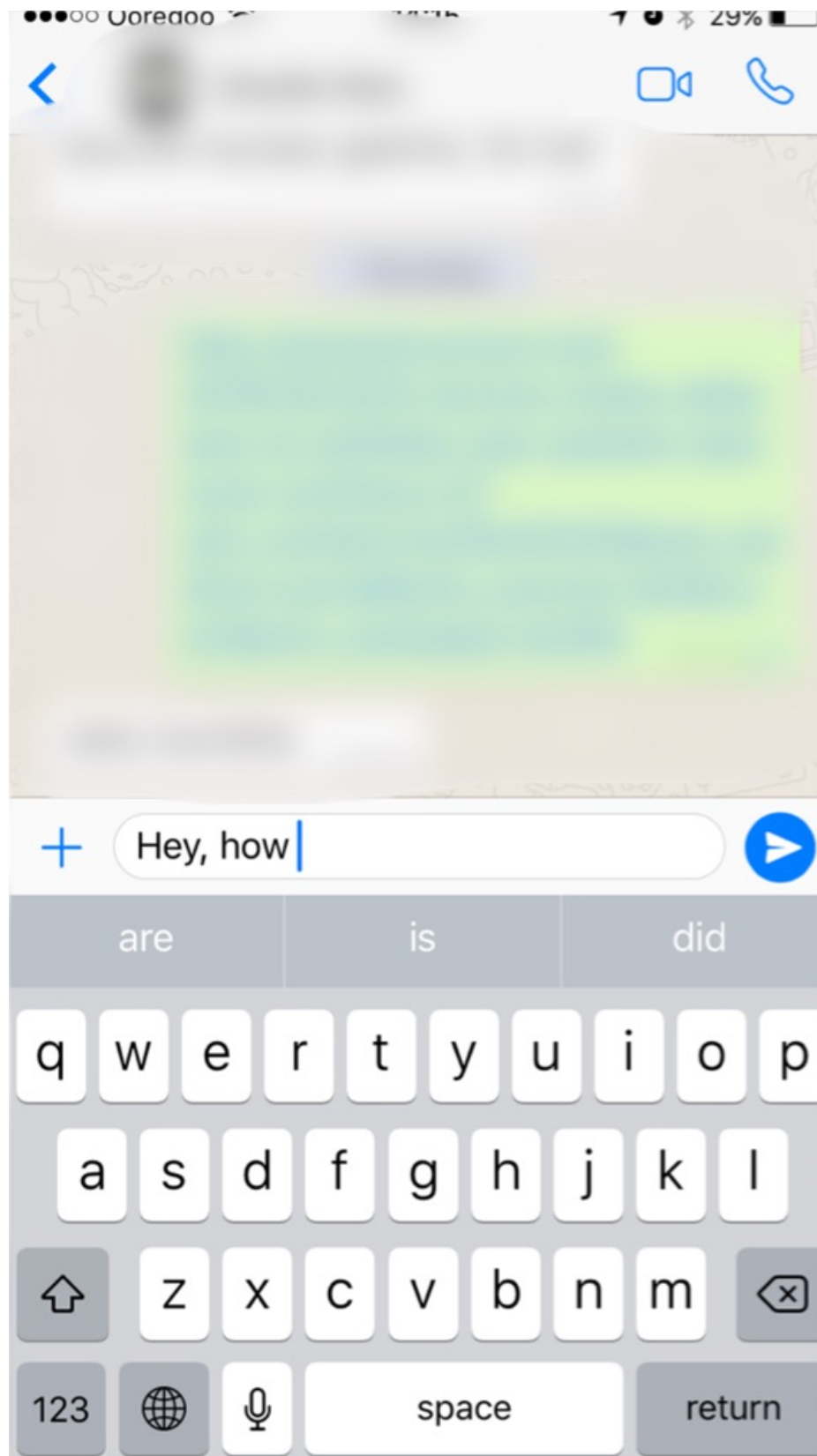
▶ $P(\text{"were"} \mid \text{"you"}) = 0.01$

▶ $P(\text{"am"} \mid \text{"you"}) = 0.00001$

▶ $P(\text{"think"} \mid \text{"you"}) = 0.003$



- ▶ Given the first word, throw a dice for the second word:
 - ▶ Generates word W_2 given W_1



- ▶ E.g. Top 3 term T that maximize:
 - ▶ $P(T \mid \text{"Hey, how"})$
 - ▶ $P(\text{"are"} \mid \text{"Hey, how"}) > P(\text{"house"} \mid \text{"Hey, how"})$
 - ▶ $P(\text{"are"} \mid \text{"Hey, how"}) > P(\text{"do"} \mid \text{"Hey, how"})$

Is the max a good function
to be implemented here?
Why?

FUNNY COMPARISON

GENERATED FROM LANGUAGE
MODELS OF THE NEW YORK TIMES

► Unigram:

Months the my and issue of year foreign new exchange's september were recession exchange new endorsed a q acquire to six executives.

► Bigram:

Last December through the way to preserve the Hudson corporation N.B.E.C. Taylor would seem to complete the major central planners one point five percent of U.S.E. has already told M.X. corporation of living on information such as more frequently fishing to keep her

► Trigram:

They also point to ninety nine point six billion dollars from two hundred four oh six three percent of the rates of interest stores as Mexico and Brazil on market conditions.

<https://pdos.csail.mit.edu/archive/scigen/>

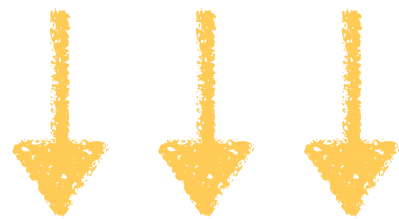
HOW CAN WE START GENERATING TEXT?

Qatar is a sovereign country located in Western **Asia**, occupying the small **Qatar Peninsula** on the northeastern coast of the **Arabian Peninsula**. Its sole land border is with Saudi **Arabia** to the south, with the rest of its territory surrounded by the **Persian Gulf**. A strait in the **Persian Gulf** separates **Qatar** from the nearby island country of Bahrain, as well as sharing maritime borders with the United **Arab** Emirates and Iran.

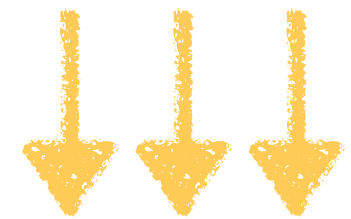
- | | | |
|------------------------------|----------------------------------|----------------------------------|
| ▶ $P(\text{"qatar"}) = 3/79$ | ▶ $P(\text{"asia"}) = 1/79$ | ▶ $P(\text{"brazil"}) = 0/79$ |
| ▶ $P(\text{"."}) = 3/79$ | ▶ $P(\text{"land"}) = 1/79$ | ▶ $P(\text{"food"}) = 0/79$ |
| ▶ $P(\text{"gulf"}) = 2/79$ | ▶ $P(\text{"persian"}) = 2/79$ | ▶ $P(\text{"continent"}) = 0/79$ |
| ▶ $P(\text{"arab"}) = 3/79$ | ▶ $P(\text{"peninsula"}) = 2/79$ | ▶ $P(\text{"house"}) = 0/79$ |

HOW CAN WE START GENERATING TEXT?

Qatar is a sovereign country located in Western **Asia**, occupying the small **Qatar Peninsula** on the northeastern coast of the **Arabian Peninsula**. Its sole land border is with Saudi **Arabia** to the south, with the rest of its territory surrounded by the **Persian Gulf**. A strait in the **Persian Gulf** separates **Qatar** from the nearby island country of Bahrain, as well as sharing maritime borders with the United **Arab** Emirates and Iran.



MAXIMUM LIKELIHOOD ESTIMATION



- | | | |
|------------------------------|----------------------------------|----------------------------------|
| ▶ $P(\text{"qatar"}) = 3/79$ | ▶ $P(\text{"asia"}) = 1/79$ | ▶ $P(\text{"brazil"}) = 0/79$ |
| ▶ $P(\text{"."}) = 3/79$ | ▶ $P(\text{"land"}) = 1/79$ | ▶ $P(\text{"food"}) = 0/79$ |
| ▶ $P(\text{"gulf"}) = 2/79$ | ▶ $P(\text{"persian"}) = 2/79$ | ▶ $P(\text{"continent"}) = 0/79$ |
| ▶ $P(\text{"arab"}) = 3/79$ | ▶ $P(\text{"peninsula"}) = 2/79$ | ▶ $P(\text{"house"}) = 0/79$ |

Bag of words again - assume independence between any two tokens

LANGUAGE MODELS FOR IR

Documents

Query

LM ϕ_1 {

qatar 0.01
location 0.002
south 0.003
arab 0.0009
...
nutrition 0.00002
food 0.00000001

LM ϕ_2 {

qatar 0.00000003
location 0.0001
south 0.00005
arab 0.003
...
nutrition 0.001
food 0.01

"capital arabic countries"

LANGUAGE MODELS FOR IR

Documents

LM ϕ_1 {

qatar 0.01
location 0.002
south 0.003
arab 0.0009
...
nutrition 0.00002
food 0.00000001

LM ϕ_2 {

qatar 0.00000003
location 0.0001
south 0.00005
arab 0.003
...
nutrition 0.001
food 0.01

Query

"capital arabic countries"

Retrieval question:

**WHAT IS THE MOST
LIKELY DOCUMENT THAT
GENERATED THIS QUERY?**

LANGUAGE MODELS FOR IR

Documents

LM ϕ_1 {

qatar 0.01
location 0.002
south 0.003
arab 0.0009
...
nutrition 0.00002
food 0.00000001

LM ϕ_2 {

qatar 0.000000003
location 0.0001
south 0.00005
arab 0.003
...
nutrition 0.001
food 0.01

Query

"capital arabic countries"

Retrieval question:

**WHAT IS THE MOST
LIKELY DOCUMENT THAT
GENERATED THIS QUERY?**

$$P(Q|\phi_1) > P(Q|\phi_2)$$

or

$$P(Q|\phi_2) > P(Q|\phi_1)$$

THEORETIC REASONING BASED ON PRP

$$P(d|q)$$

THEORETIC REASONING BASED ON PRP

USING BAYES' THEOREM AGAIN...

$$P(d|q) = \frac{P(q|d)P(d)}{P(q)}$$

THEORETIC REASONING BASED ON PRP

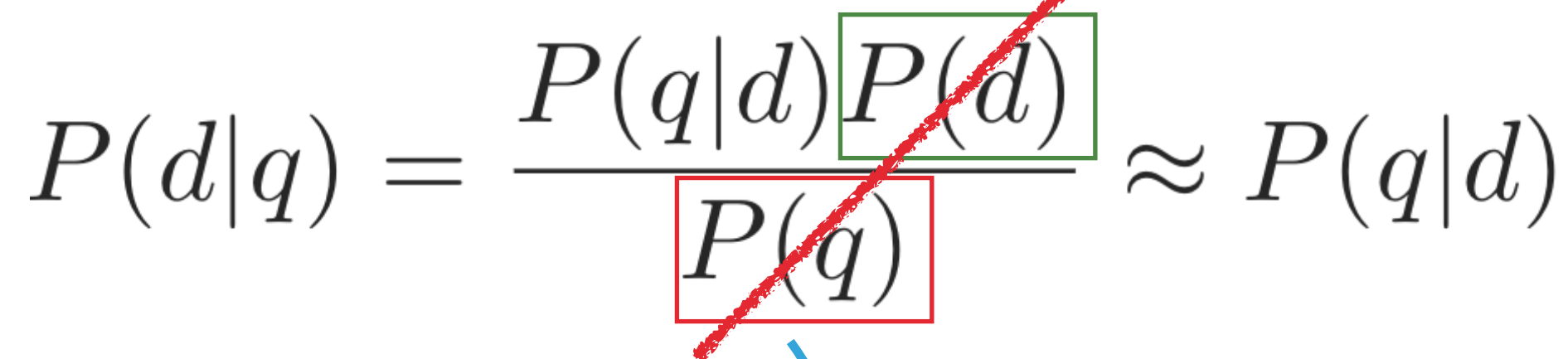
It can be a constant...
or it can be a proxy for
document popularity,
document credibility,
document readability...

$$P(d|q) = \frac{P(q|d)P(d)}{P(q)}$$

It is a constant for every
document in the collection.

THEORETIC REASONING BASED ON PRP

It can be a constant...
or it can be a proxy for
document popularity,
document credibility,
document readability...

$$P(d|q) = \frac{P(q|d)P(d)}{P(q)} \approx P(q|d)$$


It is a constant for every
document in the collection.

THEORETIC REASONING BASED ON PRP

It can be a constant...
or it can be a proxy for
document popularity,
document credibility,
document readability...

$$P(d|q) = \frac{P(q|d)P(d)}{P(q)} \approx P(q|d)$$

It is a constant for every
document in the collection.

Let's assume a document d can be represented by its language model ϕ

THEORETIC REASONING BASED ON PRP

It can be a constant...
or it can be a proxy for
document popularity,
document credibility,
document readability...

$$P(d|q) = \frac{P(q|d) \cancel{P(d)}}{\cancel{P(q)}} \approx P(q|d)$$

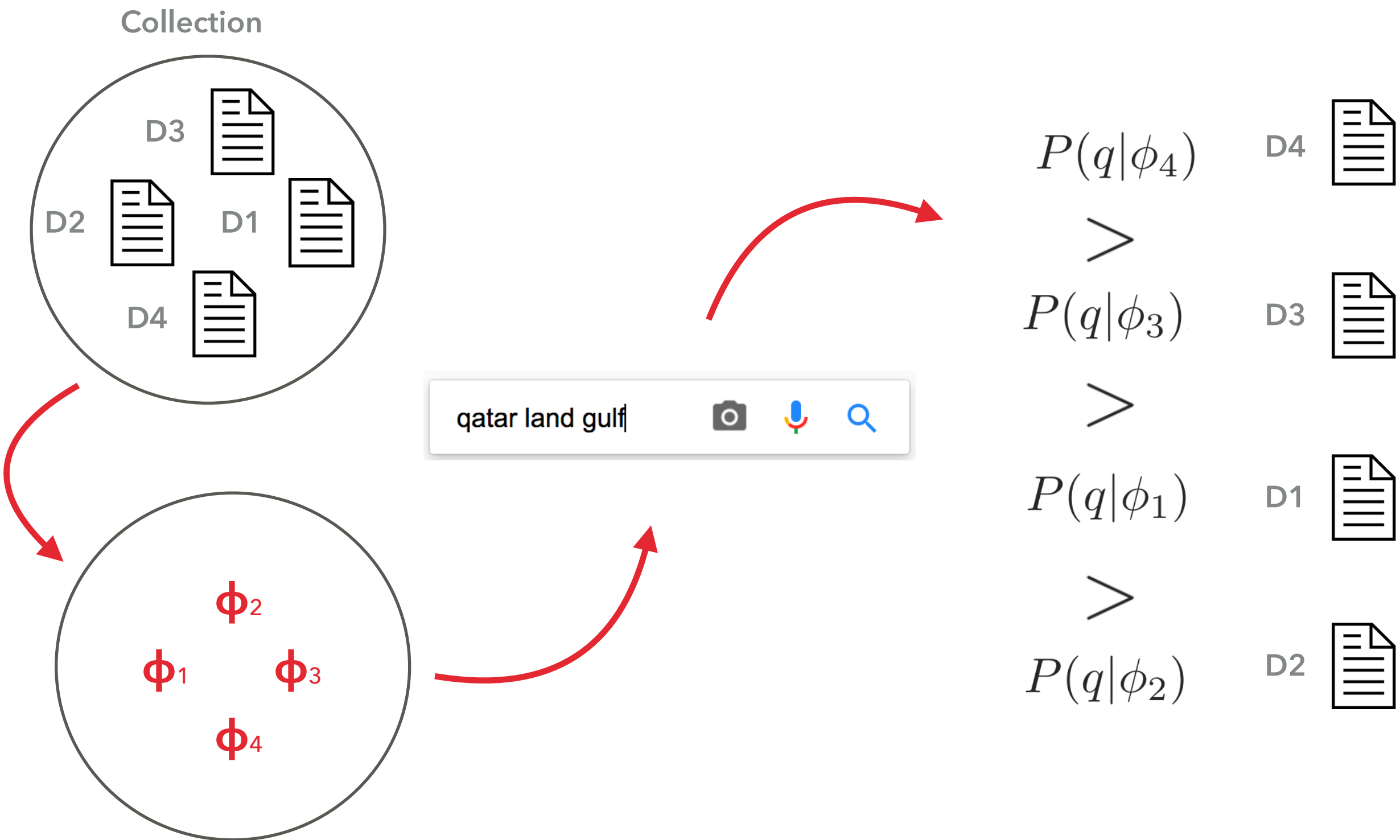
It is a constant for every
document in the collection.

Let's assume a document d can be represented by its language model ϕ

$$P(q|\phi)$$

That is all that we need to calculate!

THEORETIC REASONING BASED ON PRP



LM WITH MAXIMUM LIKELIHOOD ESTIMATION

$$P(q|\phi)$$

- Document D transformed in LM ϕ :

$$P(\text{"qatar"}) = 3/79$$

$$P(\text{"asia"}) = 1/79$$

$$P(\text{"brazil"}) = 0/79$$

$$P(\text{"."}) = 3/79$$

$$P(\text{"land"}) = 1/79$$

$$P(\text{"food"}) = 0/79$$

$$P(\text{"gulf"}) = 2/79$$

$$P(\text{"persian"}) = 2/79$$

$$P(\text{"continent"}) = 0/79$$

$$P(\text{"arab"}) = 3/79$$

$$P(\text{"peninsula"}) = 2/79$$

$$P(\text{"house"}) = 0/79$$

- Query: "qatar land gulf"

- $P(\text{"qatar land gulf"} | \phi) = 3/79 * 1/79 * 2/79 = 1.21e-05$

LM WITH MAXIMUM LIKELIHOOD ESTIMATION

$$P(q|\phi)$$

- Document D transformed in LM ϕ :

$$P(\text{"qatar"}) = 3/79$$

$$P(\text{"asia"}) = 1/79$$

$$P(\text{"brazil"}) = 0/79$$

$$P(\text{"."}) = 3/79$$

$$P(\text{"land"}) = 1/79$$

$$P(\text{"food"}) = 0/79$$

$$P(\text{"gulf"}) = 2/79$$

$$P(\text{"persian"}) = 2/79$$

$$P(\text{"continent"}) = 0/79$$

$$P(\text{"arab"}) = 3/79$$

$$P(\text{"peninsula"}) = 2/79$$

$$P(\text{"house"}) = 0/79$$

- Query: "qatar land gulf"

- $P(\text{"qatar land gulf"} | \phi) = 3/79 * 1/79 * 2/79 = 1.21e-05$

Remember: UNIGRAM model

independence between any two tokens

$$P(q|d) = \prod_w P(w|)$$

LM WITH MAXIMUM LIKELIHOOD ESTIMATION

- ▶ Actually, we DO NOT calculate the multiplication of the probabilities...

LM WITH MAXIMUM LIKELIHOOD ESTIMATION

- ▶ Actually, we DO NOT calculate the multiplication of the probabilities...

$$P(q|d) = \prod_w P(w|) = \sum_w \log P(w|\phi)$$

$$P(w|d) = \frac{c(w, d)}{|d|}$$

PROBLEM...

- ▶ Document D transformed in LM ϕ :

$P(\text{"qatar"}) = 3/79$	$P(\text{"asia"}) = 1/79$	$P(\text{"brazil"}) = 0/79$
$P(\text{"."}) = 3/79$	$P(\text{"land"}) = 1/79$	$P(\text{"food"}) = 0/79$
$P(\text{"gulf"}) = 2/79$	$P(\text{"persian"}) = 2/79$	$P(\text{"continent"}) = 0/79$
$P(\text{"arab"}) = 3/79$	$P(\text{"peninsula"}) = 2/79$	$P(\text{"house"}) = 0/79$

- ▶ Query: "qatar land continent"

PROBLEM...

- ▶ Document D transformed in LM ϕ :

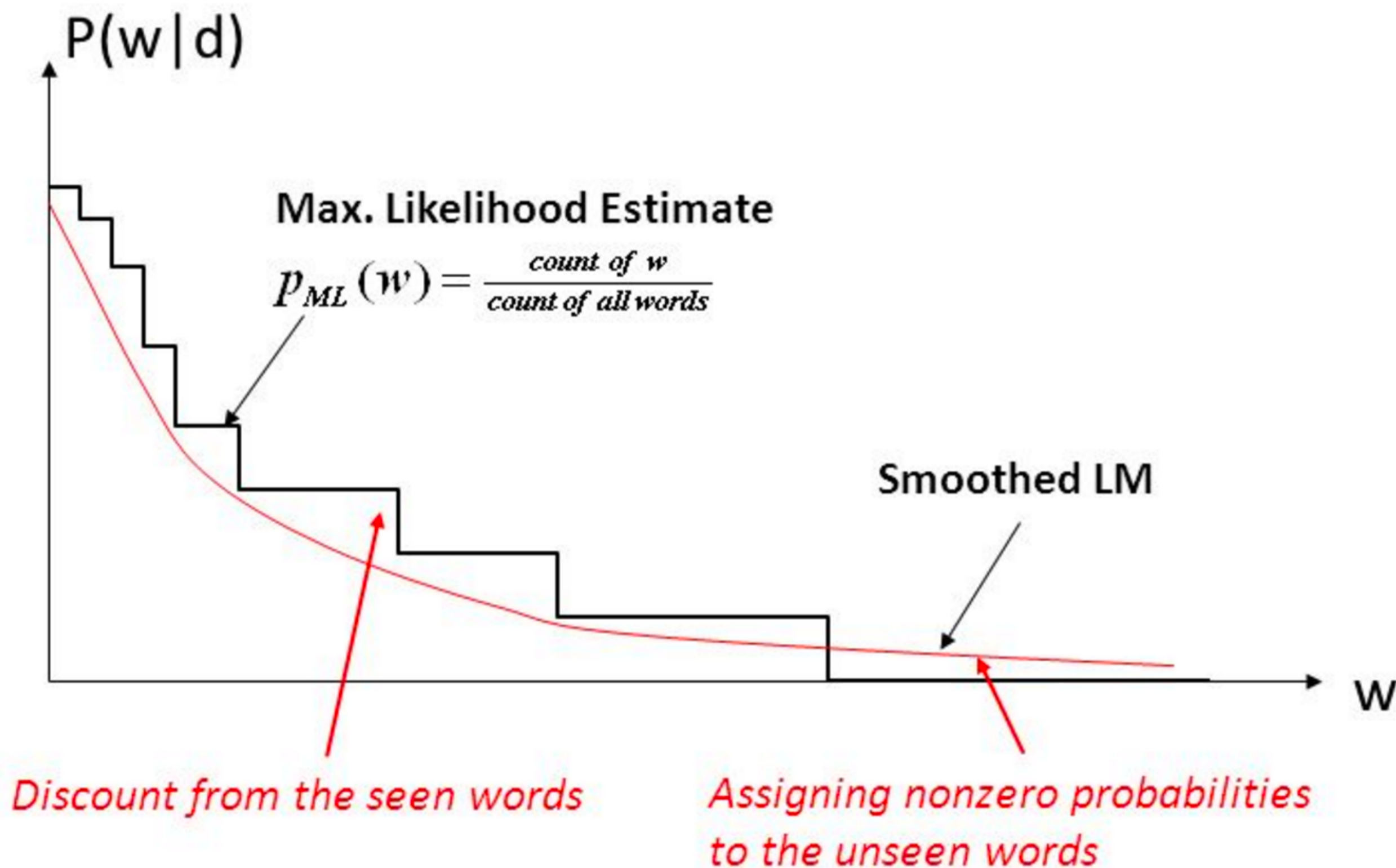
$P(\text{"qatar"}) = 3/79$	$P(\text{"asia"}) = 1/79$	$P(\text{"brazil"}) = 0/79$
$P(\text{"."}) = 3/79$	$P(\text{"land"}) = 1/79$	$P(\text{"food"}) = 0/79$
$P(\text{"gulf"}) = 2/79$	$P(\text{"persian"}) = 2/79$	$P(\text{"continent"}) = 0/79$
$P(\text{"arab"}) = 3/79$	$P(\text{"peninsula"}) = 2/79$	$P(\text{"house"}) = 0/79$

- ▶ Query: "qatar land continent"
- ▶ $P(\text{"qatar land continent"} \mid \phi) = 3/79 * 1/79 * 0/79 = 0$

DOES IT MEAN THAT A DOCUMENT WITHOUT ANY QUERY KEYWORD IS NOT RELEVANT?

IDEA OF ANY SMOOTHING METHOD

- ▶ We want to give non-zero probabilities for unseen keywords
- ▶ We discount a very tiny bit of the probability of each seen word and re-allocate this tiny bit to unseen words
- ▶ In a smoothed LM: $\forall w \ P(w \mid \phi) > 0.0$



SMOOTHING METHODS

- ▶ Method 1: Additive Smoothing / Laplace smoothing

$$P(w|d) = \frac{c(w, d) + \alpha}{|d| + \alpha|V|}$$

- ▶ Often alpha is set to 1:

$$P(w|d) = \frac{c(w, d) + 1}{|d| + |V|}$$

SMOOTHING METHODS

- ▶ Method 2: Linear Interpolation, Jelinek-Mecer

$$P(w|d) = (1 - \lambda) \frac{c(w, d)}{|d|} + (1 - \lambda)p(w|Collection)$$

- ▶ Often lambda takes any value between 0 and 1

**PROBABILITIES CAN BE CALCULATED FROM ANY COLLECTION
(YOUR OWN COLLECTION? WIKIPEDIA? WHOLE WEB?)**



SMOOTHING METHODS

- ▶ Method 3: Dirichlet Prior / Bayesian

$$P(w|d) = \frac{c(w, d) + \mu p(w|Collection)}{|d| + \mu}$$

- ▶ It is kind of a mix from previous methods

SUMMARY – USING LM FOR IR

- ▶ Choose your favorite smoothing method and parameter
- ▶ Calculate smoothed $P(Q | D)$ for each D in collection
- ▶ Rank documents with respect to their probabilities
- ▶ Return top K documents (e.g., $K = 10$) to the user
- ▶ Statistical natural language processing motivation

WHAT DID WE SEE? WHAT SHOULD YOU KNOW?

- ▶ Notes on floating representation in a computer
- ▶ Language Model
- ▶ Smoothing Methods

TODAY'S LECTURE IN THE STANFORD IR BOOK

- ▶ Chapter 12: Language models for information retrieval

HOMEWORK 1

- ▶ Comments about it from students
- ▶ Comments about it from me

HOMEWORK 2

► Explanation