Language Modeling

บริบททางภาษา

grammatical error

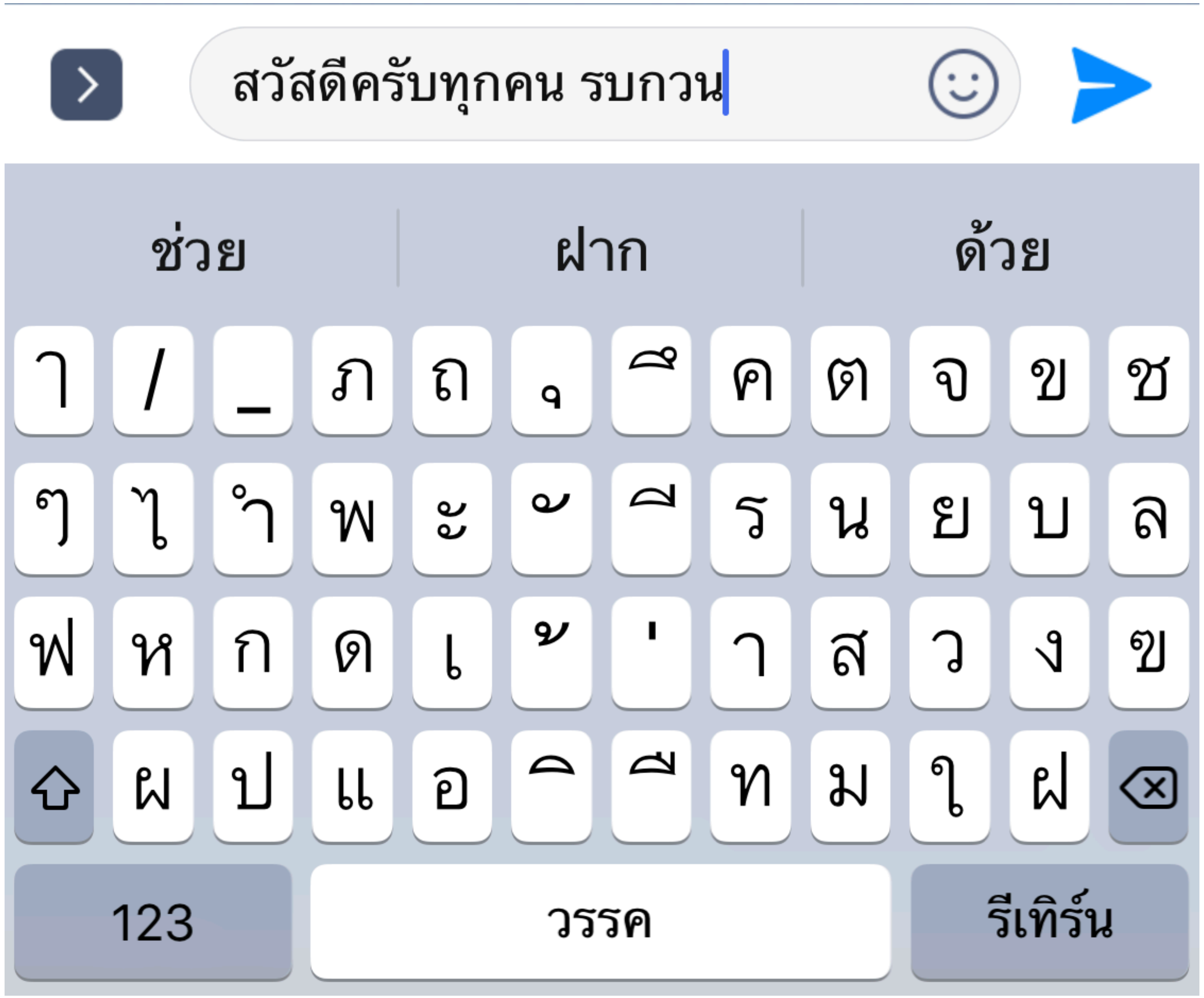
- He had **beef** for lunch vs He had **a beef** for lunch

- **อา|นอน|ตาก|ลม** vs **อาน|อน|ตา|กลม** word segmentation

speech recognition

- **I send him a letter** vs **I send dim a led her**

- กรุงเทพมีตึกสูงเยอะ machine translation
    - Bangkok has many **high** buildings vs Bangkok has many **tall** buildings


สวัสดีครับทุกคน รบกวน
ช่วย
ฝาก
ด้วย
123
วรรค
รีเทิร์น

LM ใช้ทำอะไร

- หาความน่าจะเป็น/ความเป็นไปได้ของประโยค
- ไวยากรณ์โดยไม่ต้องเขียนกฎโดยตรง
- ทำนายคำถัดไปโดยใช้บริบท

N-Gram Language Model

Model แบบโง่สุด

- Unigram Language Model
  $P(w_1, w_2, w_3, \ldots, w_n) = P(w_1)\, P(w_2)\, P(w_3)\, ..\, P(w_n)$

- Bangkok has many **high** buildings vs
  Bangkok has many **tall** buildings

$P(\text{Bangkok}) \cdot P(\text{has}) \cdot P(\text{many}) \cdot \boxed{P(\text{high})}\ P(\text{buildings})$

〃 〃 〃 $\boxed{P(\text{tall})}$ · 〃

Unigram Language Model

- fifth, an, of, futures, the, an, incorporated, a,
- a, the, inflation, most, dollars, quarter, in, is, mass
- thrift, did, eighty, said, hard, 'm, july, bullish
- that, or, limited, the

Language Model แบบมีบริบท

- Bigram Language Model
  $P(w_1, w_2, w_3, \ldots, w_n) = P(w_1|START)\ P(w_2|w_1)\ P(w_3|w_2) \ldots P(w_n|w_{n-1})$

- Bangkok has many **high** buildings vs
  Bangkok has many **tall** buildings

P(Bangkok | START) . P(has | Bangkok) . P(many | has)
. P(high | many) . P(buildings | high) .

P(tall | many) . P(buildings | tall)

Bigram Language Model

texaco, rose, one, in, this, issue, is, pursuing, growth, in, a, boiler, house, said, mr., gurria, mexico, 's, motion, control, proposal, without, permission, from, five, hundred, fifty, five, yen

outside, new, car, parking, lot, of, the, agreement, reached this, would, be, a, record, november

Trigram and 4-gram LM

- Trigram Language Model

$$
\begin{aligned}
P(w_1, w_2, w_3, \ldots, w_n) = \; & P(w_1|START1, START2) \\
& P(w_2|START2, w_1) \\
& P(w_3|w_1\ w_2) \\
& P(w_4|w_2\ w_3)\ ..\ P(w_n|w_{n-2}\ w_{n-1})
\end{aligned}
$$

P(tall | has many)

- 4-gram Language Model

$$
\begin{aligned}
P(w_1, w_2, w_3, \ldots, w_n) = \; & P(w_1|START1, START2, START3) \\
& P(w_2|START2, START3, w_1) \\
& P(w_3|START3\ w_1\ w_2) \\
& P(w_4|w_1\ w_2\ w_3)\ ..\ P(w_n|w_{n-3}\ w_{n-2}\ w_{n-1})
\end{aligned}
$$

P(tall | Bangkok has many)

โมเดลมันก็ยังโง่ๆ อยู่ดี

- Long distance dependencies (e.g. relative clauses)
    - The computers that I bought from the new mall **are/is** broken.
- 5-gram ดีๆ ส่วนใหญ่มักจะเพียงพอ

5-gram Language Model

Chain Rule of Probability

$$P(X, Y, Z) = P(X \mid YZ) \cdot \underline{P(Y, Z)}$$

Chain Rule $= P(X \mid YZ) \cdot P(Y \mid Z) \cdot P(Z)$

Chain Rule for LM

- P(<s> Bangkok has many tall shopping malls </s>) =

  P(Bangkok)

  P(has | Bangkok)

  P(many | ~~Bangkok~~ has)

  P(tall | ~~Bangkok has~~ many)

  P(shopping | ~~Bangkok has many~~ tall)

  P(malls | ~~Bangkok has many tall~~ shopping)

P(has many tall shopping malls | Bangkok)

P(has | Bangkok) · P(many tall shop malls | Bangkok has)

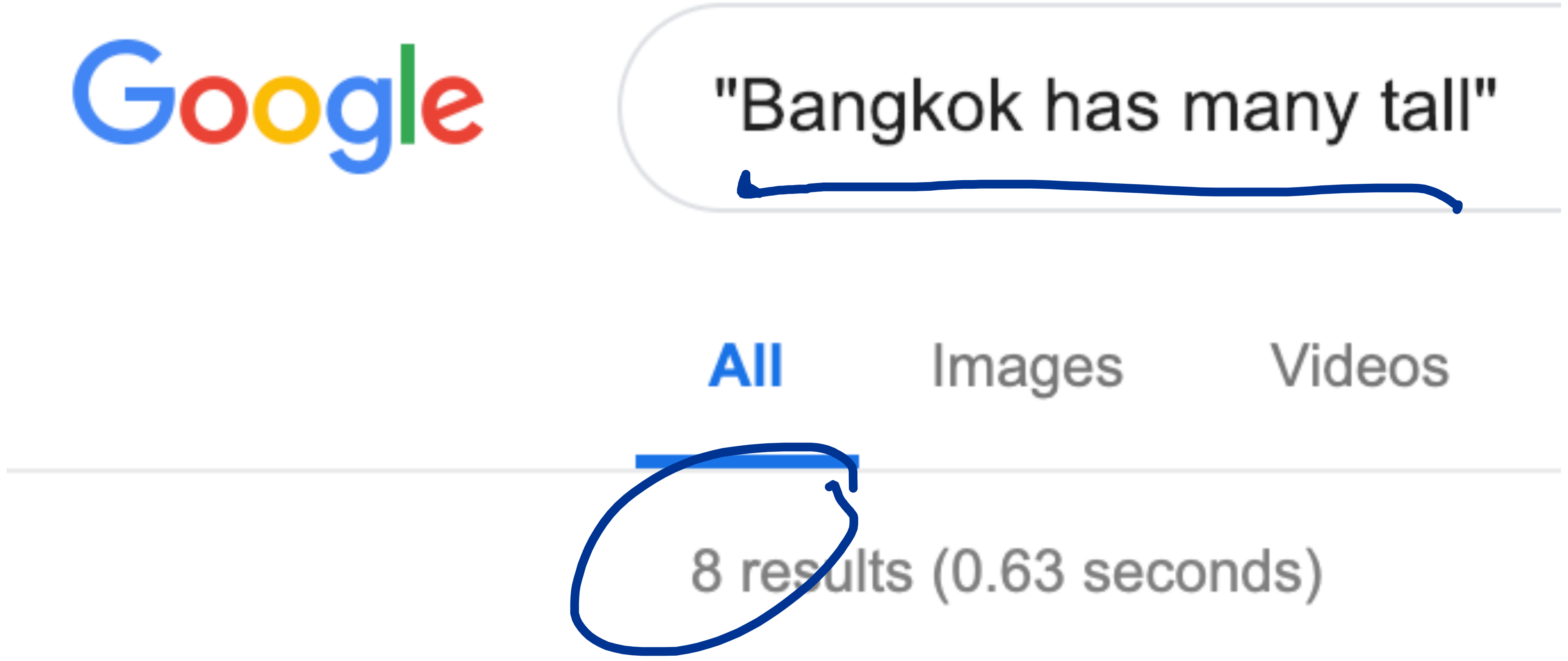
Markov Assumptions

Independence Assumption

การประมาณค่า Unigram Probability

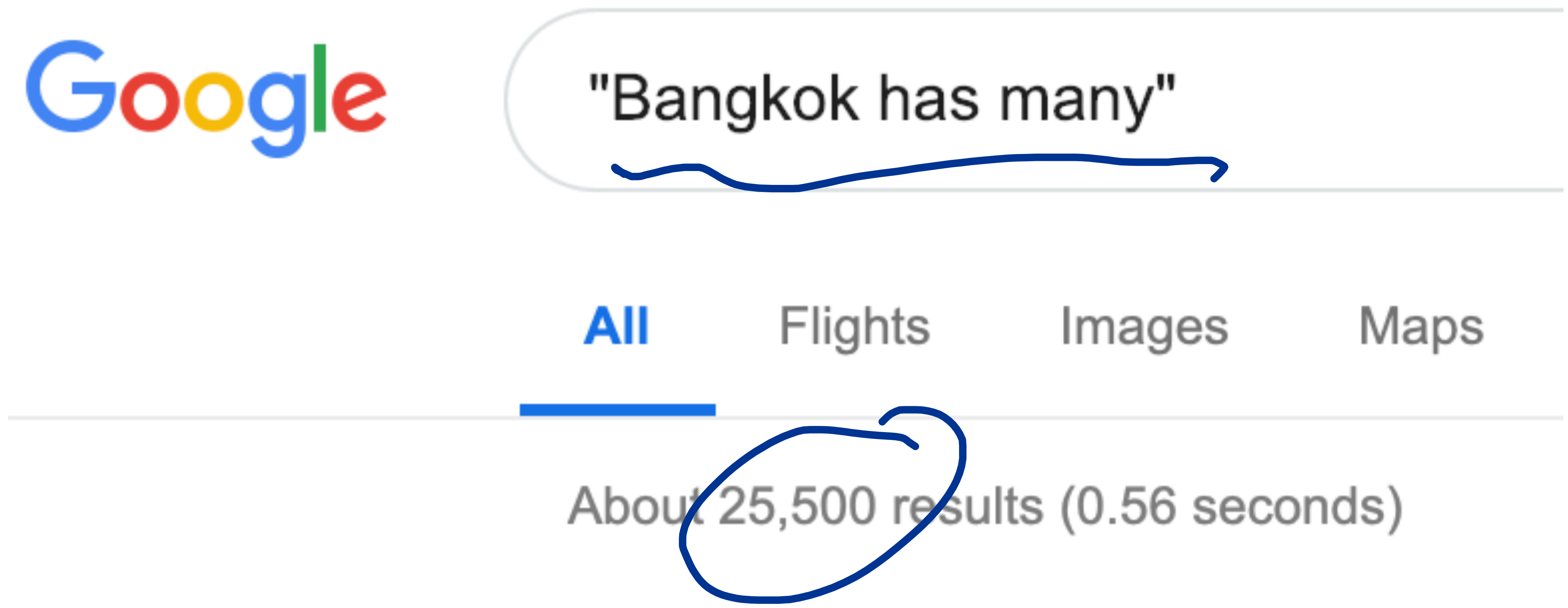
- $\hat{P}(\text{Bangkok}) = \dfrac{C(\text{Bangkok})}{\text{จำนวนคำทั้งหมด}}$

การประมาณค่า Conditional Probability



$$= 0.00031372549$$

- $P(\text{tall} \mid \text{Bangkok has many}) \approx \dfrac{C\,(\text{Bangkok has many tall})}{C\,(\text{Bangkok has many})} = 8 / 25500$



ทำไมถึงไม่ใช้ 6-gram ล่ะ

Hulk kissed Robin

$$P(\text{Robin} \mid \text{Hulk kissed}) = \frac{C(\text{Hulk kissed Robin})}{C(\text{Hulk kissed})} \quad \frac{0}{0}$$

- P(<s> Bangkok has many tall shopping malls </s>) =
  P(Bangkok)
  P(has | Bangkok)
  P(many | Bangkok has)
  P(tall | Bangkok has many)
  P(shopping | Bangkok has many tall)
  P(malls | Bangkok has many tall shopping)

$$0$$

$$= \frac{C(\text{Ban} \text{------} \text{malls})}{C(\text{Bangkok} \ldots \text{shopping})}$$

ตัวอย่างการฝึก LM

An example

$$P(w_i \mid w_{i-1}) = \frac{c(w_{i-1}, w_i)}{c(w_{i-1})}$$

<s> I am Sam </s>
<s> Sam I am </s>
<s> I do not like green eggs and ham </s>

$$\frac{C(I, am)}{C[I]} = \frac{2}{3}$$

$P(\texttt{I} \mid \texttt{<s>}) = \frac{2}{3} = .67$ $\quad P(\texttt{Sam} \mid \texttt{<s>}) = \frac{1}{3} = .33$ $\quad P(\texttt{am} \mid \texttt{I}) = \frac{2}{3} = .67$

$P(\texttt{</s>} \mid \texttt{Sam}) = \frac{1}{2} = 0.5$ $\quad P(\texttt{Sam} \mid \texttt{am}) = \frac{1}{2} = .5$ $\quad P(\texttt{do} \mid \texttt{I}) = \frac{1}{3} = .33$

More examples: Berkeley Restaurant Project sentences

- can you tell me about any good cantonese restaurants close by
- mid priced thai food is what i’m looking for
- tell me about chez panisse
- can you give me a listing of the kinds of food that are available
- i’m looking for a good place to eat breakfast
- when is caffe venezia open during the day

Dan Jurafsky

Raw bigram counts

- Out of 9222 sentences

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

Dan Jurafsky

Raw bigram probabilities

$$P(\text{want} \mid I) = \frac{C(I\ \text{want})}{C(I)}$$

- Normalize by unigrams:

i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

- Result:

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

Dan Jurafsky

Bigram estimates of sentence probabilities

P(<s> I want english food </s>) =

P(I|<s>)

× P(want|I)

× P(english|want)

× P(food|english)

× P(</s>|food)

= .000031

Dan Jurafsky

What kinds of knowledge?

- P(english|want) = .0011
- P(chinese|want) = .0065
- P(to|want) = .66
- P(eat | to) = .28
- P(food | to) = 0
- P(want | spend) = 0
- P (i | <s>) = .25

domain knowledge

grammar/syntax

discourse

การประเมินความสามารถของ LM
(Model evaluation)

ระเบียบวิธีการประเมิน

Bigram Trigram

- แบ่งข้อมูลออกเป็นสามส่วน
  - Training set – collect counts ✓ ✓
  - Development set / validation set 50 $\boxed{30}$
  - Test set $\boxed{35}$
- มาตรวัดความสามารถ (evaluation metric)

Extrinsic Evaluation

- ใช้มาตรวัดใน Tasks อื่นๆ ที่จำเป็นต้องมี LM
    - Machine Translation
    - Speech Recognition
- A/B Testing - ใช้กับผู้ใช้จริงๆ
    - Spell checker / Grammar checker
    - Predictive keyboard

A	B
Model เดิม	model ใหม่

Perplexity for LM

intrinsic

- Standard evaluation metric for LM

- Perplexity = ความมึนงง

  ยิ่งน้อย ยิ่งดี

  ถ้า LM เราดีจริงเวลาเห็นคำต่อไปเราไม่ควรจะมึน

  Patients spend a lot of time waiting _______ _______ .

  for doctors

$P(\text{for})$

$P(\text{for} \mid \text{waiting})$

bigram LM

$P(\text{doctors} \mid \text{for})$

$P(\text{doctors} \mid \text{waiting for})$

trigram LM

Perplexity for LM

$$PP(W) = P(w_1 w_2 \ldots w_N)^{-\frac{1}{N}}$$

จำนวนคำใน test set

$$= \sqrt[N]{\frac{1}{P(w_1 w_2 \ldots w_N)}}$$

ต้องสูง

P(W) สูง ⟷ perplexity ต่ำ ⟷ performance ดี

Dan Jurafsky

Lower perplexity = better model

- Training 38 million words, test 1.5 million words, WSJ

N-gram Order	Unigram	Bigram	Trigram
Perplexity	962	170	109

Implementing LM

log P(W) ดีกว่า P(W)

$$P(w_1, w_2, \cdots w_{100}) = P(w_1) \cdot P(w_2) \cdot \ldots P(w_{100})$$

$$= 0.001^{100} \quad \text{underflow}$$

$$\log(P(w_1) \cdot P(w_2) \ldots P(w_{100}))$$

$$= \log P(w_1) + \log P(w_2) \ldots + \log P(w_{100})$$

$$= -100.46$$

$$P(w_1 \ldots w_{100})^{-\frac{1}{100}} = \exp\left(\log P(w_1 \ldots w_{100})^{-\frac{1}{100}}\right)$$

$$= \exp\left(-\frac{1}{100} \log P(w_1 \cdots w_{100})\right)$$

Overfitting and Underfitting

Overfitting

- Training corpus ควรจะเหมือนกับ test corpus

- Training corpus ต้องมีจำนวนคำมาก

$$P(\text{buildings} \mid \text{many tall}) = \frac{1}{100000}$$

$$= \frac{C(\text{many tall buildings})}{C(\text{many tall})} = \frac{1}{10000}$$

Out-of-vocabulary (OOV)

- เปลี่ยนบางคำใน training set เป็น UNK
    - คำที่เกิดน้อยว่า k ครั้ง
    - คำที่ไม่อยู่ใน top 50000

OOV rate 7%

- ตอนเปรียบเทียบโมเดลต้องใช้ vocabulary เดียวกัน

ความน่าจะเป็นของคำที่เกิด 0 ครั้ง