Vector Space Model for Search

TF-IDF weighting

Co-occurrence

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
ANTHONY	157	73	0	0	0	1	
BRUTUS	4	157	0	2	0	0	
CAESAR	232	227	0	2	1	0	
CALPURNIA	0	10	0	0	0	0	
CLEOPATRA	57	0	0	0	0	0	
MERCY	2	0	3	8	5	8	
WORSER	2	0	1	1	1	5	
...							

	Anthony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	Macbeth	...
ANTHONY	5.25	3.18	0.0	0.0	0.0	0.35	
BRUTUS	1.21	6.10	0.0	1.0	0.0	0.0	
CAESAR	8.59	2.54	0.0	1.51	0.25	0.0	
CALPURNIA	0.0	1.54	0.0	0.0	0.0	0.0	
CLEOPATRA	2.85	0.0	0.0	0.0	0.0	0.0	
MERCY	1.51	0.0	1.90	0.12	5.25	0.88	
WORSER	1.37	0.0	0.11	4.15	0.25	1.95	
...							

$$w_{t,d} = (1 + \log \mathrm{tf}_{t,d}) \cdot \log \frac{N}{\mathrm{df}_t}$$

docs

docs with t

ทบทวน vector space model

- โมเดลความหมายของ query และความหมายของ document ด้วย vector (project query และ document ลงบน space)
- cosine similiarity ในการเปรียบเทียบความหมาย

Cosine similarity ระหว่าง query กับ document

query		The Tempest	Hamlet	Othello	Macbeth
0	ANTHONY	0.0	0.0	0.0	0.35
0	BRUTUS	0.0	1.0	0.0	0.0
1	CAESAR	0.0	1.51	0.25	0.0
0	CALPURNIA	0.0	0.0	0.0	0.0
0	CLEOPATRA	0.0	0.0	0.0	0.0
1	MERCY	1.90	0.12	5.25	0.88
1	WORSER	0.11	4.15	0.25	1.95
	...				

query		The Tempest	Hamlet	Othello	Macbeth
0	ANTHONY	0.0	0.0	0.0	0.35
0	BRUTUS	0.0	1.0	0.0	0.0
1	CAESAR	0.0	1.51	0.25	0.0
0	CALPURNIA	0.0	0.0	0.0	0.0
0	CLEOPATRA	0.0	0.0	0.0	0.0
1	MERCY	1.90	0.12	5.25	0.88
1	WORSER	0.11	4.15	0.25	1.95
	...				

$$\frac{1 \cdot 1.9 + 1 \cdot 0.11}{\sqrt{3} \cdot \sqrt{1.90^2 + 0.11^2}}$$

relevance score

$$\cos(\vec{q}, \vec{d}) = \text{SIM}(\vec{q}, \vec{d}) = \frac{\vec{q} \cdot \vec{d}}{|\vec{q}||\vec{d}|} = \frac{\sum_{i=1}^{|V|} q_i d_i}{\sqrt{\sum_{i=1}^{|V|} q_i^2} \sqrt{\sum_{i=1}^{|V|} d_i^2}}$$

normalization

สูตรไหนใช้ได้ก็ใช้ ใช้ไม่ได้ก็เททิ้งไป

Term frequency		Document frequency		Normalization	
n (natural)	$\mathrm{tf}_{t,d}$	n (no)	1	n (none)	1
l (logarithm)	$1 + \log(\mathrm{tf}_{t,d})$	t (idf)	$\log \frac{N}{\mathrm{df}_t}$	c (cosine)	$\frac{1}{\sqrt{w_1^2+w_2^2+\ldots+w_M^2}}$
a (augmented)	$0.5 + \frac{0.5 \times \mathrm{tf}_{t,d}}{\max_t(\mathrm{tf}_{t,d})}$	p (prob idf)	$\max\{0, \log \frac{N-\mathrm{df}_t}{\mathrm{df}_t}\}$	u (pivoted unique)	$1/u$
b (boolean)	$\begin{cases} 1 & \text{if } \mathrm{tf}_{t,d} > 0 \\ 0 & \text{otherwise} \end{cases}$			b (byte size)	$1/CharLength^{\alpha}$, $\alpha < 1$
L (log ave)	$\frac{1+\log(\mathrm{tf}_{t,d})}{1+\log(\mathrm{ave}_{t \in d}(\mathrm{tf}_{t,d}))}$				

ddd.qqq

lnc.ltc

สูตรไหนใช้ได้ก็ใช้ ใช้ไม่ได้ก็เททิ้งไป

$$\sum_{w} tf_{w,Q} \cdot \frac{tf_{w,D}}{tf_{w,D} + \frac{k|D|}{avg|D|}} \cdot \log \frac{|C|}{df_{w}}$$

Evaluation of Relevance Model - Precision, Recall, F1

The usual

- precision
- recall
- f1 = 2 * (P + R) / (P * R)

The usual

- precision

  จำนวนครั้งที่ทายถูก / จำนวนครั้งที่ทาย

  จำนวนเอกสารทายถูกว่าเกี่ยวข้อง / จำนวนเอกสารที่เอามาให้ดู

- recall

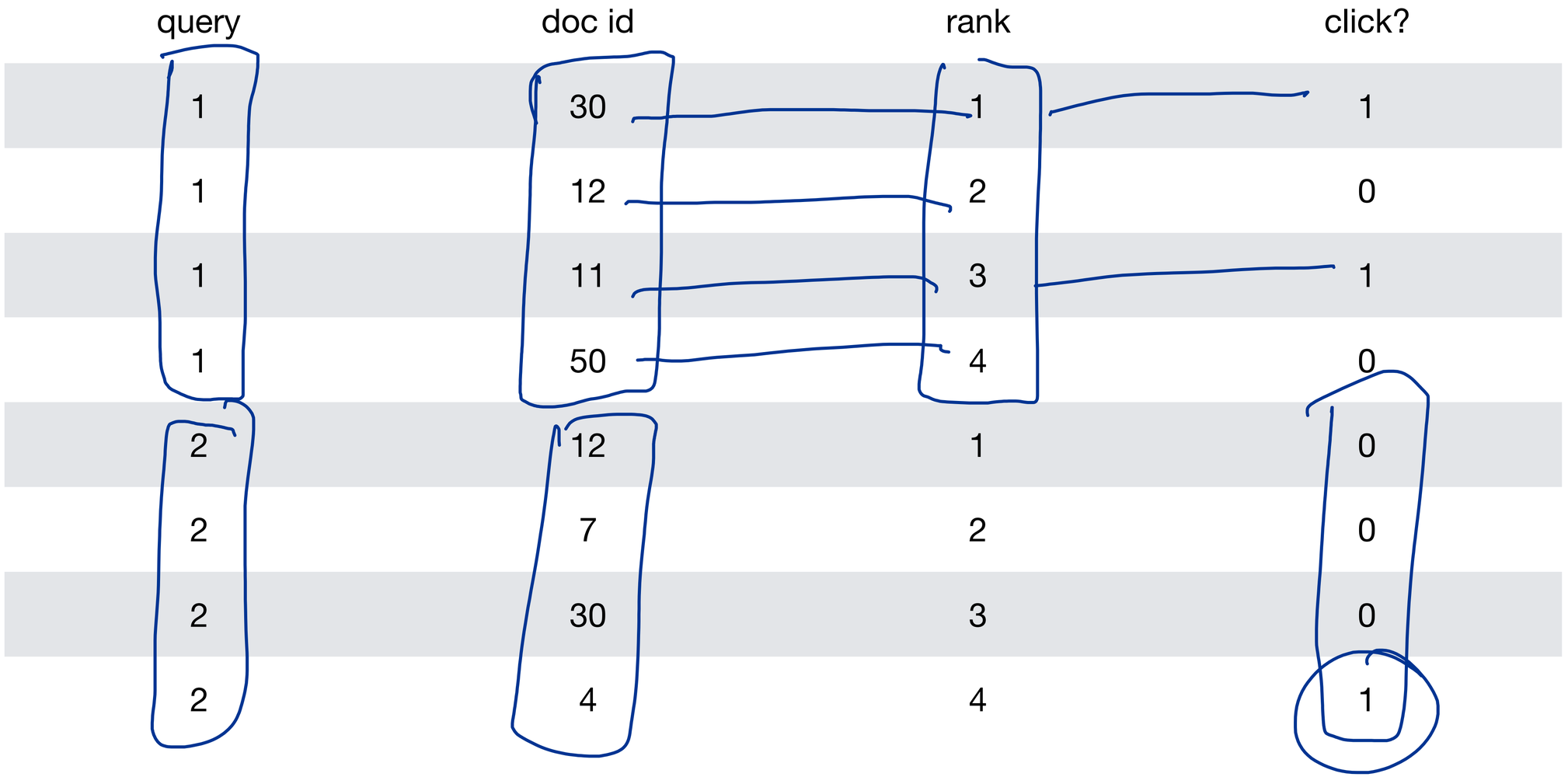
  จำนวนครั้งที่ทายถูก / จำนวนคำตอบที่ถูก

  จำนวนเอกสารทายถูกว่าเกี่ยวข้อง / จำนวนเอกสารที่เกี่ยวข้องทั้งหมด

- f1 = 2 * (P + R) / (P * R)

Click data

~ query log
Search log



Click model

- precision@k

  $$\frac{\text{จำนวนเอกสารทายถูกว่าเกี่ยวข้อง}}{\text{จำนวนเอกสารที่เอามาให้ดู}}$$

  @2 $\frac{1}{4}$ @3 $\frac{2}{6}$

- recall@k

  $$\frac{\text{จำนวนเอกสารทายถูกว่าเกี่ยวข้อง}}{\text{จำนวนเอกสารที่เกี่ยวข้องทั้งหมด}}$$

  $\frac{1}{3}$ $\frac{2}{3}$

query	doc id	rank	click?
1	30	1	1
1	12	2	0
1	11	3	1
1	50	4	0
2	12	1	0
2	7	2	0
2	30	3	0
2	4	4	1

Evaluation of Relevance Model - nDCG

Ranking

query	doc id	rank	click?
1	30	1	1
1	12	2	0
1	11	3	1
1	50	4	0
2	12	1	0
2	7	2	0
2	30	3	0
2	4	4	1

query	doc id	rank	click?
1	30	1	1
1	11	2	1
1	12	3	0
1	50	4	0
2	4	1	1
2	7	2	0
2	30	3	0
2	12	4	0

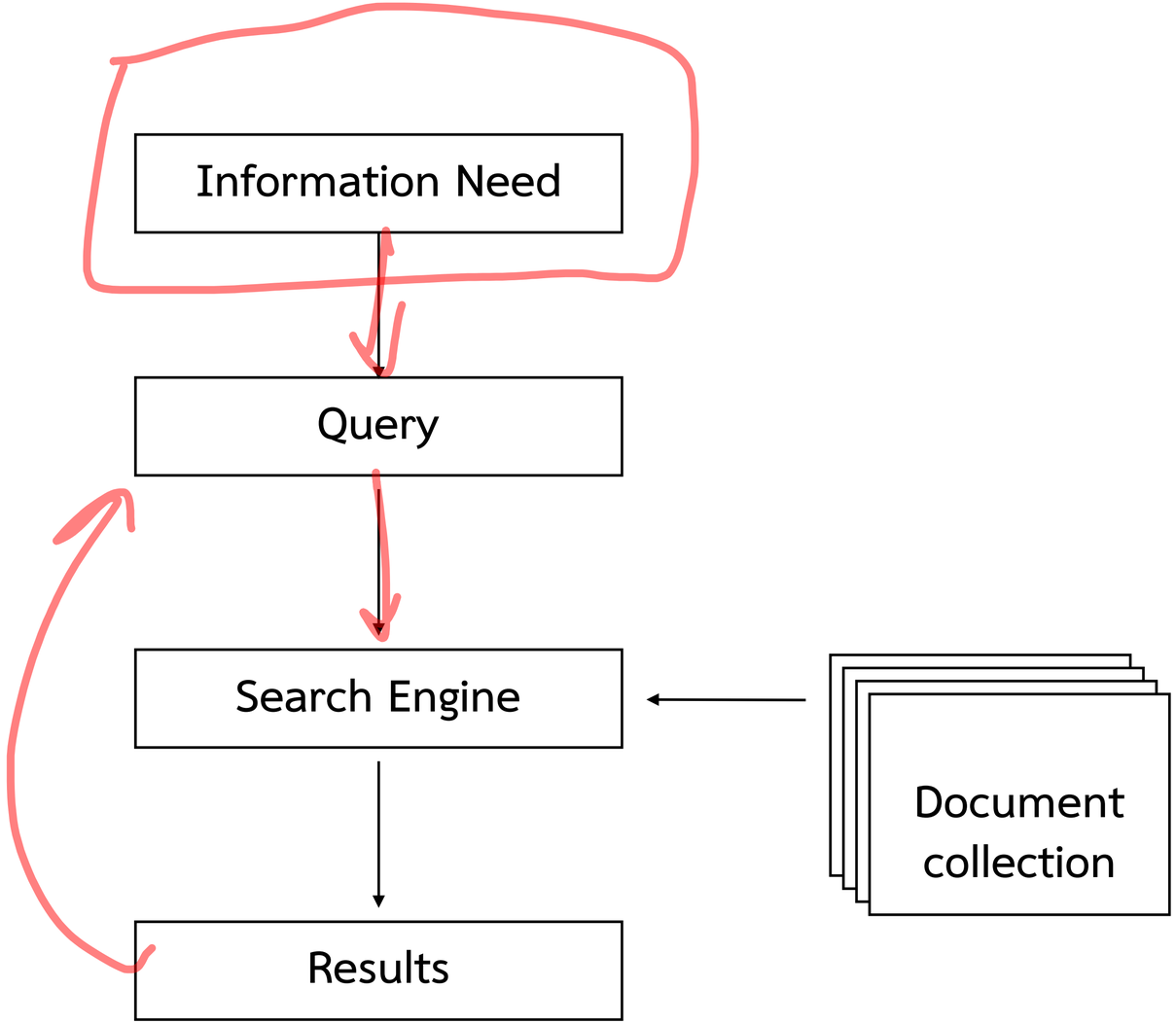
Rank	Judgment (Gain)	Discounted Gain	Discounted Cumulative Gain (DCG)	Ideal Discounted Gain	Ideal Discounted Cumulative Gain (iDCG)	Normalized Discounted Cumulative Gain (NDCG)
1	2	2/1	2	3/1	3.0	0.67
2	0	0/2	2	2/2	4.0	0.5
3	3	3/3	3	2/3	4.67	0.64
4	2	2/4	3.5	0/4	4.67	**0.75**

Evaluation of IR System

ข้อจำกัดของการใช้ Click Data

- ถ้า doc ที่ดีกว่านี้มันไม่อยู่ใน search results แล้วทำไง
- คลิกเยอะแล้วดีจริงเหรอ
- คลิกน้อยแล้วดีจริงเหรอ

Classic Search Model



ปัญหาของการใช้
Intrinsic evaluation

Metric มาตรวัดไหนดีกว่ากัน

Clickthrough Rate (CTR)

อัตราการสั่งอาหาร

อัตราการจองโรงแรม

อัตราการสั่งซื้อสินค้า

A/B Testing (online testing)

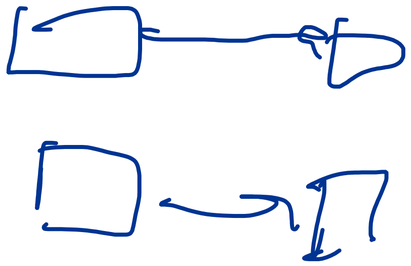
- Word Segmentation —> Thai Character Cluster?
- วิธีการคำนวณ relevance score แบบใหม่

Step 1

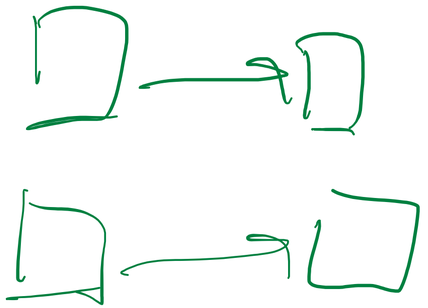
- คำนวณ nDCG@k จาก click data เอาให้แน่ใจว่าจะลอง A/B testing

Step 2 สร้าง search engine สองตัว

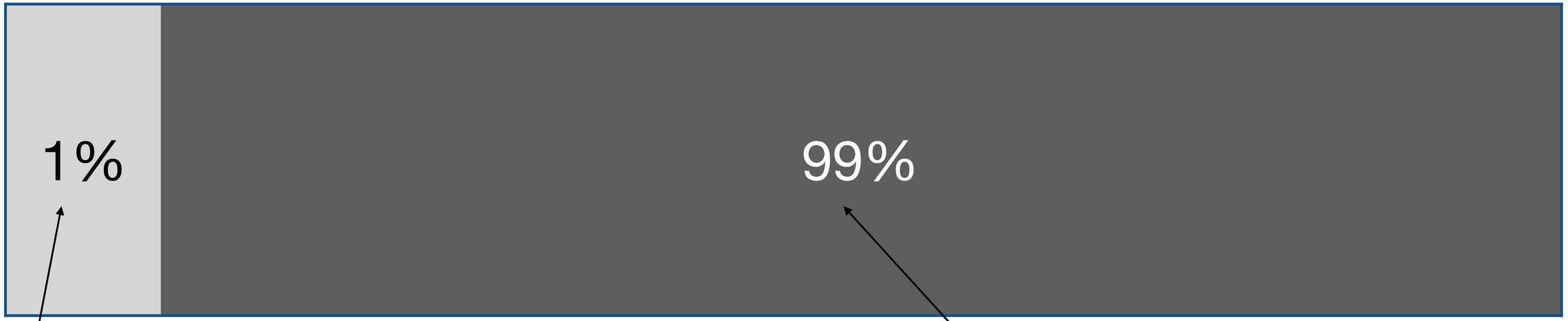
- Word Segmentation (ของเดิม)



- Thai Character Cluster (อยากลอง)



Step 3 แบ่งกลุ่มผู้ใช้



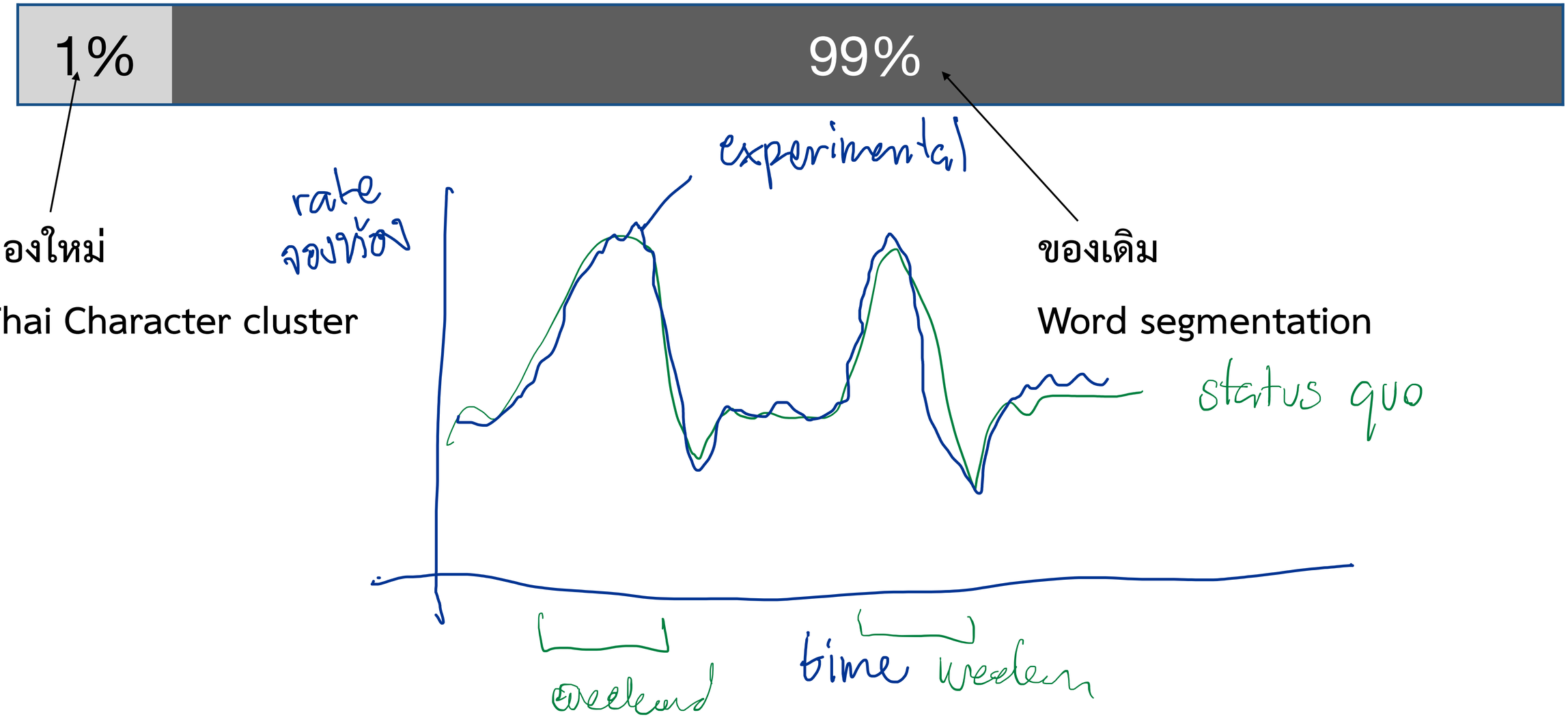
ของใหม่

Thai Character cluster

ของเดิม

Word segmentation

Step 4 รอนานๆ แล้วดูผล



Step 4 รอนานๆ แล้วดูผล

- ผลต่างมักจะเล็กมากๆๆๆๆๆ ต้องรอเก็บสถิติเยอะๆ ถึงจะแน่ใจว่าผลที่เราเห็นมีนัยสำคัญจริงๆ (ไม่ใช่ฟลุค)
    - อัตราการสั่งอาหารอาจจะเพิ่มจาก 0.50% --> 0.51%

      0.01% อาจจะเท่ากับรายได้เพิ่มเป็นแสนๆบาทต่อปี

Step 5 ค่อยๆ ถ่ายไประบบใหม่

- ค่อยๆถ่าย traffic ไปยังระบบใหม่
- เช็คว่า effect ที่เห็นนั้นยังอยู่ไม่หายไปไหน

A/B Testing

- ข้อดี
  - ชัวร์มากว่าตอบโจทย์ information need รึเปล่า
  - จะใช้ metric อะไรก็ได้ที่สนใจและวัดได้
- ข้อเสีย
  - ใช้เวลาเยอะ
  - ตั้งระบบยาก

Semantic Search (Query Expansion)

ระบบไม่ได้แตะความหมายจริงๆ

- รับสมัครครูมัธยม

อาจารย์ ✗

- รับสมัครอาจารย์มัธยม

teacher (m.)

- Lehrer in Berlin

Lehrer/in

Lehrer ✗ in

(f.)

- Lehrerin in Berlin ✗

Semantic Search Hack

- Hack = ลวกๆ ไม่มีหลักการ แต่ว่าลองแล้วมันดันใช้ได้

Lexical Semantics

query expansion

WordNet

Token	Lexical relation	Terms
รับ	-	รับ
สมัคร	antonymy	สมัคร, จ้าง
อาจารย์	synonym	อาจารย์, ครู
โรงเรียน	hypernym	โรงเรียน, สถานศึกษา
มัธยม	-	มัธยม

Computational Lexical Semantics

```
>>> w2v_model.most_similar('อาจารย์')
[('คณาจารย์', 0.537608504295349),
 ('ลูกศิษย์', 0.477556765079498),
 ('ครู', 0.451356738805770),
 ('นักศึกษา', 0.440014481544946),
 ('ศาสตราจารย์', 0.422398895025253),
 ('ศิษย์เก่า', 0.418981373310089),
 ('อาจารย์พิเศษ', 0.412405639886856),
 ('ศิษย์', 0.408566117286682),
 ('นักเรียน', 0.401798427104949),
 ('รองศาสตราจารย์', 0.399857819080352)]
```

1. train embeddings on docs
2. most similar candidates
3. review manual

Query Expansion

- ใช้ (computational) lexical semantics ในการทำความเข้าใจ query และ document

Semantic Search (Query Understanding)

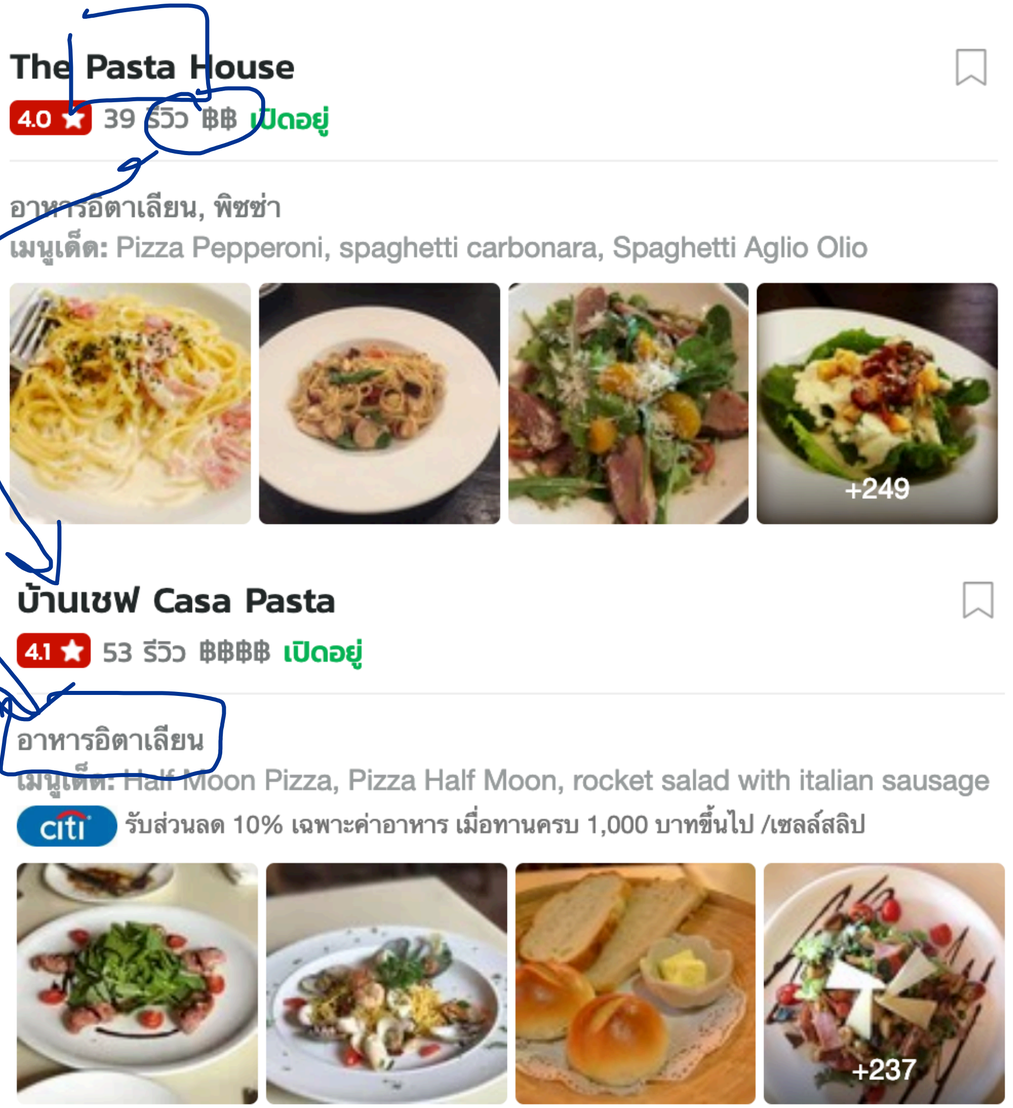
Query Understanding

- พาสต้า โรแมนติก สีลม ไม่ แพง

Italian

location

attribute



query tagging

พาสต้า โรแมนติก สีลม ไม่ แพง

พาสต้า

Category:Italian

โรแมนติก

Location: สีลม

Attribute: ฿฿

Category:Italian

Location: สีลม

Attribute: ฿฿

The Pasta House

4.0 ★ 39 รีวิว ฿฿ เปิดอยู่

อาหารอิตาเลียน, พิซซ่า

เมนูเด็ด: Pizza Pepperoni, spaghetti carbonara, Spaghetti Aglio Olio

บ้านเชฟ Casa Pasta

4.1 ★ 53 รีวิว ฿฿฿฿ เปิดอยู่

อาหารอิตาเลียน

เมนูเด็ด: Half Moon Pizza, Pizza Half Moon, rocket salad with italian sausage

citi รับส่วนลด 10% เฉพาะค่าอาหาร เมื่อทานครบ 1,000 บาทขึ้นไป /เซลล์สลิป

Category:Italian

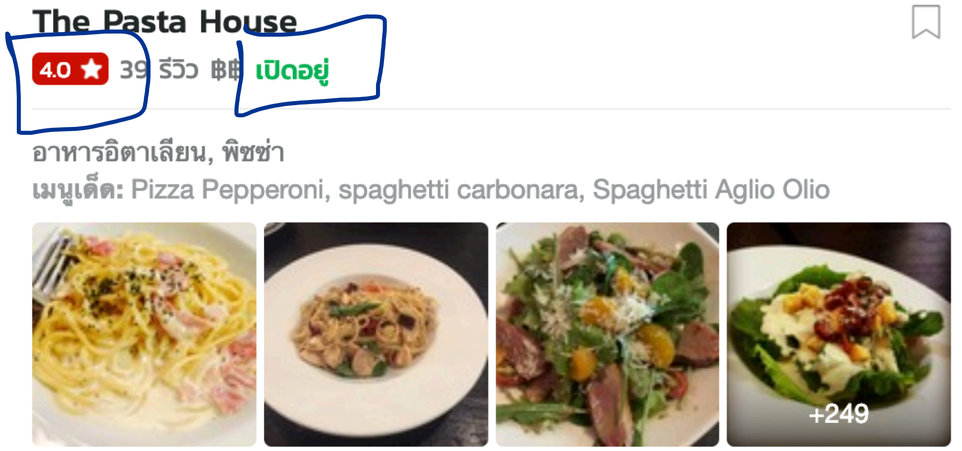
Location: อุดมสุข

Attribute: ฿฿฿฿

Learning to Rank

ปัจจัยอื่นๆ

- ระยะห่างระหว่างคน search กับร้านอาหาร
- จำนวนดาว
- เปิดอยู่รึเปล่า



ปัจจัยอื่นๆ

Zoning

- q = ใบตอบรับ อาจารย์พิทยาวัฒน์

จาก: อาจารย์พิทยาวัฒน์

หัวข้อ: อย่าลืมปิดไฟ

10 ม.ค. 2561

จาก: คณะอักษรศาสตร์

หัวข้อ: ใบตอบรับ

10 ม.ค. 2562

Features for Search

- Score(q, d)
  - TermScore(q, d) โดยใช้ TF-IDF
  - FromScore(q, d)
  - TitleScore(q, d)
  - QueryExpansionScore(q, d)
  - Distance score(u, d)
  - Recency score(u, d)

ทำนายว่าจะ doc จะถูกคลิกมั้ย

features

$$\sum_{w} tf_{w,Q} \cdot \frac{tf_{w,D}}{tf_{w,D} + \frac{k|D|}{avg|D|}} \cdot \log \frac{|C|}{df_w}$$

query	doc id	rank	click?	Term score	Title Score	Recency
1	30	1	1	0.6	0.2	5
1	12	2	0	0.4	0.1	10
1	11	3	1	0.35	0.5	3
1	50	4	0	0.2	0.5	2
2	12	1	0	0.9	0.2	4
2	7	2	0	0.2	0.6	2
2	30	3	0	0.1	0.5	1
2	4	4	1	0.1	0.1	4

Pointwise model

Y = label

features: w_1, w_2, w_3

$P(Y|X)$

$P(\text{click} | w_1, w_2, w_3)$

$= \beta_1$

click?	Term score	Title Score	Recency
1	0.6	0.2	5
0	0.4	0.1	10
1	0.35	0.5	3
0	0.2	0.5	2
0	0.9	0.2	4
0	0.2	0.6	2
0	0.1	0.5	1
1	0.1	0.1	4