Step 0

Answer

2.1 เลือกใช้ Extend Boolean Model เนื่องจากลักษณะของ Query เป็นแบบ Boolean และโจทย์กำหนดให้ Keyword ไม่สัมพันธ์กัน

Boolean Model

- เนื่องจาก **Boolean Model** มีข้อเสียคือการไม่สนใจน้ำหนักของ **Keyword**
- **Vector Space Model** มีข้อเสียคือการเชื่อมต่อทางตรรกะทำได้ยาก

จึงได้มีความพยายามที่นำข้อดีของทั้งสองมารวมกัน ทำเป็นModel ใหม่ขึ้นมา เรียกว่า **Extended Boolean Model**

เอกสาร 10 เอกสารมีการแจกแจง Keyword ดังนี้

```
D1: {bird,cat,bird,cat,dog,dog,bird}
D2: {cat,tiger,cat,dog}
D3: {dog,bird,bird}
D4: {cat,tiger}
D5: {tiger,tiger,dog,tiger,cat}
D6: {bird,cat,bird,cat,tiger,tiger,bird}
D7: {bird,tiger,cat,dog}
D8: {dog,cat,bird}
D9: {cat,dog,tiger}
D10: {tiger,tiger,tiger}
```

	Bird	Cat	Dog	Tiger	Max
Doc1	3	2	2	0	3
Doc2	0	2	1	1	2
Doc3	2	0	1	0	2
Doc4	0	1	0	1	1
Doc5	0	1	1	3	3
Doc6	3	2	0	2	3
Doc7	1	1	1	1	1
Doc8	1	1	1	0	1
Doc9	0	1	1	1	1
Doc10	0	0	0	3	3
n	5	8	7	7	

Query = รักแมวและสุนัข แต่ไม่รักเสือ

Query = (Cat AND Dog) AND NOT Tiger (OR ก็ได้)

Step 1 หา นน. แต่ละ key ใน แต่ละ Doc

tf แบบ normalize

idf แบบ normalize

น้ำหนักของ Keyword ในเอกสาร

น้ำหนักของ Keyword "i" ในเอกสาร "j"

$$w_{i,j} = tfnorm_{i,j} * idfnorm_i$$

$$tfnorm_{i,j} = \frac{tf_{i,j}}{tfmax_{i,j}}$$

tf ของ Keyword "i" ในเอกสาร "j"; ค่าสูงสุดที่หาได้ของ tf ของ Keyword ในเอกสาร "j"

normalized TF ของ Keyword "i" ในเอกสาร "j"

$$idfnorm_i = \frac{idf_i}{idfmax_g}$$

idf ของ Keyword "i" ในเอกสารทั้งหมด; ค่าสูงสุดที่หาได้ของ idf ของ Keyword ในเอกสารทั้งหมด

normalized IDF ของ Keyword "i" ในเอกสารทั้งหมด

Doc1

หาทุก Doc นะจ๊ะ

$$tf_{bird} = \frac{3}{3} = 1.000$$
$$tf_{cat} = \frac{2}{3} = 0.667$$
$$tf_{dog} = \frac{2}{3} = 0.667$$
$$tf_{tiger} = \frac{0}{3} = 0.000$$

normorlize เรียบร้อย ไม่ต้องคิดต่อ เพราะ max = 1

ใช้ทุก step การคำนวณ

$$idf_{bird} = \log(\frac{10}{5}) = 0.301 \qquad idf_{norm,bird} = \frac{0.301}{0.301} = 1.000$$
$$idf_{cat} = \log(\frac{10}{8}) = 0.097 \qquad idf_{norm,cat} = \frac{0.097}{0.301} = 0.322$$
$$idf_{dog} = \log(\frac{10}{7}) = 0.155 \qquad idf_{norm,dog} = \frac{0.155}{0.301} = 0.515$$
$$idf_{tiger} = \log(\frac{10}{7}) = 0.155 \qquad idf_{norm,tiger} = \frac{0.155}{0.301} = 0.515$$

(10 → จำนวนเอกสารทั้งหมด; 5 → เอกสารที่มี)

$$w_{bird} = 1.000 * 1.000 = 1.000$$
$$w_{cat} = 0.667 * 0.322 = 0.215$$
$$w_{dog} = 0.667 * 0.515 = 0.343$$
$$w_{tiger} = 0.000 * 0.515 = 0.000$$

น้ำหนักของแต่ละ Keyword ในแต่ละเอกสาร

	Bird	Cat	Dog	Tiger
Doc1	1.000	0.215	0.343	0.000
Doc2	0.000	0.322	0.257	0.257
Doc3	1.000	0.000	0.257	0.000
Doc4	0.000	0.322	0.000	0.515
Doc5	0.000	0.107	0.172	0.515
Doc6	1.000	0.215	0.000	0.343
Doc7	1.000	0.322	0.515	0.515
Doc8	1.000	0.322	0.515	0.000
Doc9	0.000	0.322	0.515	0.515
Doc10	0.000	0.000	0.000	0.515

Step 2 คำนวณ

Query = รักแมวและสุนัข แต่ไม่รักเสือ

Query = (Cat AND Dog) AND NOT Tiger

ความครอบคลุมของ Query

OR

AND

NOT = 1 − W keyword i

AND

$$sim(q_{and}, dj) = 1 - \sqrt{\frac{(1-W_{1,j})^2 + (1-W_{2,j})^2}{2}}$$

(W_1: cat and dog; W_2: not tiger)

ถ้ามี 3 ก็จะ

$$\sqrt{\frac{(1-w_{1,j})^2 + (1-w_{2,j})^2 + (1-w_{3,j})^2}{2}}$$

$$sim(q_{and}, dj) = 1 - \sqrt{\frac{\left(1-\left(1-\sqrt{\frac{(1-WCat_{,j})^2+(1-Wdog_{,j})^2}{2}}\right)\right)^2 + (1-(1-W_{Tiger,j}))^2}{2}}$$

เอกสาร 1

$$sim(q_{and}, d_1) = 1 - \sqrt{\frac{\left(1-\left(1-\sqrt{\frac{(1-0.215)^2+(1-0.343)^2}{2}}\right)\right)^2 + (1-(1-0.000))^2}{2}}$$

$$sim(q_{and}, d_1) = 0.488$$

→ ทำทุก Doc

ได้ degree

	Sim
Doc1	0.488
Doc2	0.465
Doc3	0.377
Doc4	0.295
Doc5	0.291
Doc6	0.320
Doc7	0.447
Doc8	0.583
Doc9	0.447
Doc10	0.205

Step 3 จัด rank

Rank → Doc8,Doc1,Doc2,Doc7,Doc9,Doc3,Doc6,Doc4,Doc5,Doc10