1.

$$RSV_d = \log \prod_{t: x_t = q_t = 1} \frac{p_t(1-u_t)}{u_t(1-p_t)} = \sum_{t: x_t = q_t = 1} \log \frac{p_t(1-u_t)}{u_t(1-p_t)}$$

	Documents	Relevant	Non-Relevant	TOTAL
Term Present	$x_t = 1$	s	$df_t - s$	df_t
Term Absent	$x_t = 0$	$S - s$	$(N - df_t) - (S - s)$	$N - df_t$
	TOTAL	S	$N - S$	N

sol: 令 Q_k 為回傳的 document for query. $\Rightarrow$ $p_{tk} = P(x_i = 1 \mid R = 1, Q_k)$.

$u_{tk} = P(x_i = 1 \mid R = 0, Q_k)$.

假設 p_t、u_t 僅跟 query 中是否有出現該字有關，即

$p_t = p_{tk}$、$u_t = u_{tk}$、$\forall k: q_{kt} = 1$.

從表格數據可得：

$$p_t = P(x_i = 1 \mid R = 1, Q) = \frac{s}{S} \Rightarrow \frac{p_t}{1-p_t} = \frac{s}{S-s}$$

$$u_t = P(x_i = 1 \mid R = 0, Q) = \frac{df_t - s}{N - S} \Rightarrow \frac{u_t}{1-u_t} = \frac{df_t - s}{(N - df_t) - (df_t - s)}$$

(1) $$RSV_d = \sum_{t: x_t = q_t = 1} \log \frac{p_t(1-u_t)}{u_t(1-p_t)} = \sum_{t: x_t = q_t = 1} \log \left(\frac{p_t}{1-p_t} - \log \frac{u_t}{1-u_t} \right)$$

假設 document size 夠大（即 $N \gg S$、$N \gg df_t \gg s$），則

$$\frac{p_t}{1-p_t} = \frac{s}{S} \approx \frac{s}{S}$$

$$\frac{u_t}{1-u_t} = \frac{df_t - s}{(N - df_t) - (df_t - s)} \approx \frac{df_t}{N}$$

$$\Rightarrow RSV_d \approx \sum_{t: x_t = q_t = 1} \left(\log \frac{s}{S} - \log \frac{df_t}{N} \right) = \sum_{t: x_t = q_t = 1} \left(\log \frac{s}{S} + \boxed{\log \frac{N}{df_t}} \right)$$

$\boxed{\log \frac{N}{df_t}}$ → IDF.

∴ RSJ 和 TF-IDF 都有考慮 IDF 的資訊，而 RSJ 多考慮了 relevant 的資訊，

$\log \frac{p_t}{1-p_t}$

TF-IDF 則是多考慮了 TF 的資訊.

(2) Probabilistic Retrieval Model:

需要有初始假設，亦即假設所有 query term 會估計出相同的 document relevant value，也就是說，包含相同 terms 的不同文章將產生相同的相關機率估計

Vector Space Model:

不需要任何初始假設，因此在 general cases 下表現較好.

(3) $RSV_d = \sum_{t: x_t = q_t = 1} \log \frac{p_t(1-u_t)}{u_t(1-p_t)}$

$$= \sum_{t: x_t = q_t = 1} \log \frac{\frac{s}{S}\left(1 - \frac{df_t - s}{N-S}\right)}{\frac{df_t - s}{N-S}\left(1 - \frac{s}{S}\right)}$$

$$= \sum_{t: x_t = q_t = 1} \log \frac{\frac{s}{S} \cdot \frac{(N-S)-(df_t-s)}{N-S}}{\frac{df_t - s}{N-S} \cdot \frac{S-s}{S}}$$

$$= \sum_{t: x_t = q_t = 1} \log \frac{s((N-S)-(df_t-s))}{(df_t-s)(S-s)}$$

$$= \sum_{t: x_t = q_t = 1} \log \left(\frac{s}{S-s} \times \frac{(N-S)-(df_t-s)}{df_t-s}\right)$$

以單一 term 來討論 ranking $\Rightarrow \log \frac{s}{S-s} \times \frac{(N-S)-(df_t-s)}{df_t-s}$

$$= \log \frac{s}{S-s} + \log \frac{(N-S)-(df_t-s)}{df_t-s}$$

分子(s)越大，代表相關文章越容易出現該 query term.

分母(S-s)越大，代表相關文章越不容易出現該 query term.

與 IDF 有相同效果(第1小題)，故可用來判斷該 query term 是否具有鑑別性.

取項相加的分數越高，代表該 query term 既與文章相關，又具鑑別性，因此可用於 ranking.

2. how much wood would a woodchuck chuck
if a woodchuck could chuck wood.
he would chuck he would as much as he could
and chuck as much as a woodchuck would.
if a woodchuck could chuck wood.

sol:

(1) MLE-estimated unigram probability model : $\hat{P}(t|M_d) = \frac{tf_{t,d}}{|d|}$.

$\Rightarrow P(\text{would}) = \frac{4}{37}$, $P(\text{chuck}) = \frac{5}{37}$ 對.

(2) MLE-estimated bigram model : $\hat{P}(t_1|t_2) = \frac{tf_{t_1,t_2}}{tf_{t_2}}$.

$\Rightarrow P(\text{wood}|\text{chuck}) = \frac{2}{5}$, $P(\text{chuck}|\text{would}) = \frac{1}{4}$ 對.

3.

Doc ID	Document Text.
1	click go the shears boy click click click.
2	click click.
3	metal here
4.	metal shears click here.

Sol:

Mixture model $\Rightarrow P(q|d) \propto \prod_{1 \le k \le |q|} (\lambda P(t_k|M_d) + (1-\lambda) P(t_k|M_c))$, $\lambda = \frac{1}{2}$.

	click	go	the	shears	buys	metal	here
Model 1	$\frac{4}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	$\frac{1}{8}$	0	0
Model 2	$\frac{2}{2}$	0	0	0	0	0	0
Model 3	0	0	0	0	0	$\frac{1}{2}$	$\frac{1}{2}$
Model 4	$\frac{1}{4}$	0	0	$\frac{1}{4}$	0	$\frac{1}{4}$	$\frac{1}{4}$
Model 5	$\frac{7}{16}$	$\frac{1}{16}$	$\frac{1}{16}$	$\frac{2}{16}$	$\frac{1}{16}$	$\frac{2}{16}$	$\frac{2}{16}$.

(1) Doc 1 : $\frac{1}{2} \times \frac{4}{8} + \frac{1}{2} \times \frac{7}{16} = 0.46875$

Doc 2 : $\frac{1}{2} \times \frac{2}{2} + \frac{1}{2} \times \frac{7}{16} = 0.71875$.

Doc 3 : $\frac{1}{2} \times 0 + \frac{1}{2} \times \frac{7}{16} = 0.21875$.

Doc 4 : $\frac{1}{2} \times \frac{1}{4} + \frac{1}{2} \times \frac{7}{16} = 0.34375$

$\Rightarrow$ Ranking : Doc 2 > Doc 1 > Doc 4 > Doc 3. ✱

(2). Doc 1 : $\frac{1}{2} \times \frac{1}{8} + \frac{1}{2} \times \frac{2}{16} = 0.125$.

Doc 2 : $\frac{1}{2} \times 0 + \frac{1}{2} \times \frac{2}{16} = 0.0625$.

Doc 3 : $\frac{1}{2} \times 0 + \frac{1}{2} \times \frac{2}{16} = 0.0625$.

Doc 4 : $\frac{1}{2} \times \frac{1}{4} + \frac{1}{2} \times \frac{2}{16} = 0.1875$.

$\Rightarrow$ Ranking : Doc 4 > Doc 1 > Doc 2 = Doc 3. ✱

(3). 利用前面 2 小題的結果計算 $P(\text{click}|\#\text{doc}) \times P(\text{shears}|\#\text{doc})$

Doc 1 : $0.46875 \times 0.125 \doteq 0.0586$.

Doc 2 : $0.71875 \times 0.0625 \doteq 0.0449$

Doc 3 : $0.21875 \times 0.0625 \doteq 0.0137$.

Doc 4 : $0.34375 \times 0.1875 \doteq 0.0645$

$\Rightarrow$ Ranking : Doc 4 > Doc 1 > Doc 2 > Doc 3. ✱.

4. Doc 1: He moved from Taiwan, Taipei to Taiwan, Nantou.

Doc 2: He moved from Taiwan, Nantou to Taiwan, Taipei.

Doc 3: He moved from Nantou to Taiwan, Taipei.

sol:

Mixture model $\Rightarrow P(q|d) \propto \prod_{1 \le k \le |q|} \left(\lambda P(t_k|M_d) + (1-\lambda) P(t_k|M_c)\right)$, $\lambda = \frac{1}{2}$.

Query: Taipei Taiwan. $\Rightarrow$ Doc 1: $\left(\frac{1}{2} \times \frac{1}{8} + \frac{1}{2} \times \frac{3}{23}\right) \times \left(\frac{1}{2} \times \frac{2}{8} + \frac{1}{2} \times \frac{5}{23}\right) \fallingdotseq 0.0298$.

Doc 2: $\left(\frac{1}{2} \times \frac{1}{8} + \frac{1}{2} \times \frac{3}{23}\right) \times \left(\frac{1}{2} \times \frac{2}{8} + \frac{1}{2} \times \frac{5}{23}\right) \fallingdotseq 0.0298$.

Doc 3: $\left(\frac{1}{2} \times \frac{1}{7} + \frac{1}{2} \times \frac{3}{23}\right) \times \left(\frac{1}{2} \times \frac{1}{7} + \frac{1}{2} \times \frac{5}{23}\right) \fallingdotseq 0.0246$.

$\Rightarrow$ Ranking : Doc 1 = Doc 2 > Doc 3

5.

	DocID	Words in Document	in c = China
Training Set.	1	Taipei Taiwan	Yes
	2	Macao Taiwan Shanghai	Yes
	3	Japan Sapporo	No
	4	Sapporo Osaka Taiwan.	No.
Testing Set.	5	Taiwan Taiwan Sapporo	?

sol:

(1) $P(\text{Taiwan}|c) = \frac{2+1}{5+7} = \frac{3}{12} = \frac{1}{4}$.

$P(\text{Sapporo}|c) = \frac{0+1}{5+7} = \frac{1}{12}$

$P(\text{Taiwan}|\bar{c}) = \frac{1+1}{5+7} = \frac{2}{12} = \frac{1}{6}$.

$P(\text{Sapporo}|\bar{c}) = \frac{2+1}{5+7} = \frac{3}{12} = \frac{1}{4}$.

(2) $P(c|D_5) = \frac{3}{4}\left(\left(\frac{1}{4}\right)^2 \cdot \frac{1}{12}\right) \fallingdotseq 0.0039$

$P(\bar{c}|D_5) = \frac{1}{4} \cdot \left(\left(\frac{1}{6}\right)^2 \cdot \frac{1}{4}\right) \fallingdotseq 0.0017$

$\Rightarrow$ Yes!!

(3) $P(\text{Taipei}|c) = P(\text{Macao}|c) = P(\text{Shanghai}|c) = \frac{1+1}{2+2} = \frac{2}{4} = \frac{1}{2}$.

$P(\text{Taiwan}|c) = \frac{2+1}{2+2} = \frac{3}{4}$

$P(\text{Japan}|c) = P(\text{Osaka}|c) = P(\text{Sapporo}|c) = \frac{0+1}{2+2} = \frac{1}{4}$.

$P(\text{Taipei}|\bar{c}) = P(\text{Macao}|\bar{c}) = P(\text{Shanghai}|\bar{c}) = \frac{0+1}{2+2} = \frac{1}{4}$.

$P(\text{Taiwan}|\bar{c}) = P(\text{Japan}|\bar{c}) = P(\text{Osaka}|\bar{c}) = \frac{1+1}{2+2} = \frac{2}{4} = \frac{1}{2}$.

$P(\text{Sapporo}|\bar{c}) = \frac{2+1}{2+2} = \frac{3}{4}$.

(d). $P(c|D_5) \propto P(c) \cdot P(\text{Taiwan}|c) \cdot P(\text{Sapporo}|c) \cdot (1-P(\text{Taipei}|c)) \cdot (1-P(\text{Macao}|c)) \cdot (1-P(\text{Shanghai}|c)) \cdot (1-P(\text{Japan}|c)) \cdot (1-P(\text{Osaka}|c))$

$= \frac{3}{4} \times \frac{3}{4} \times \frac{1}{4} \times (1-\frac{1}{2}) \times (1-\frac{1}{2}) \times (1-\frac{1}{2}) \times (1-\frac{1}{4}) \times (1-\frac{1}{4})$

$\fallingdotseq 0.0099$

$P(\bar{c}|D_5) \propto P(\bar{c}) \cdot P(\text{Taiwan}|\bar{c}) \cdot P(\text{Sapporo}|\bar{c}) \cdot (1-P(\text{Taipei}|\bar{c})) \cdot (1-P(\text{Macao}|\bar{c})) \cdot (1-P(\text{Shanghai}|\bar{c})) \cdot (1-P(\text{Japan}|c)) \cdot (1-P(\text{Osaka}|c))$

$= \frac{1}{4} \times \frac{1}{2} \times \frac{3}{4} \times (1-\frac{1}{4}) \times (1-\frac{1}{4}) \times (1-\frac{1}{4}) \times (1-\frac{1}{2}) \times (1-\frac{1}{2})$

$\fallingdotseq 0.0099$

6. (經典題題目太長瞬睿秒ろ~).

sol:

(1) 如 RSJ 模型，使用 $P(R=1|D,Q)$ 做 ranking 等價於使用 odds $O(R|D,Q)$ 做 ranking，Q.

$$O(R|D,Q) = O(R|Q) \cdot \frac{P(D|R=1,Q)}{P(D|R=0,Q)}$$

其中 $O(R|Q)$ 與 document 無關可忽略，故 ranking 等價於 $\frac{P(D|R=1,Q)}{P(D|R=0,Q)}$

使用 multinomial 的假設 $\Rightarrow P(D|R=1,Q) = \frac{n!}{x_1!x_2!\cdots x_V!} p_1^{x_1} p_2^{x_2} \cdots p_V^{x_V}$

$$P(D|R=0,Q) = \frac{n!}{x_1!x_2!\cdots x_V!} u_1^{x_1} u_2^{x_2} \cdots u_V^{x_V}$$

其中 $x_i = c(w_i, D)$，$p_i = P(w=w_i|R=1,Q)$，$u_i = P(w=w_i|R=0,Q)$.

故，ranking 就可以寫成.

$$\frac{P(D|R=1,Q)}{P(D|R=0,Q)} = \frac{\frac{n!}{x_1!x_2!\cdots x_V!} p_1^{x_1} p_2^{x_2} \cdots p_V^{x_V}}{\frac{n!}{x_1!x_2!\cdots x_V!} u_1^{x_1} u_2^{x_2} \cdots u_V^{x_V}} = \prod_{i=1}^{V} \frac{p_i^{x_i}}{u_i^{x_i}}$$

取 log $\Rightarrow \log \prod_{i=1}^{V} \frac{p_i^{x_i}}{u_i^{x_i}} = \sum_{i=1}^{V} \log (\frac{p_i}{u_i})^{x_i}$

$$= \sum_{i=1}^{V} x_i \log \frac{p_i}{u_i}$$

$$= \sum_{w \in V} c(w_i, D) \log \frac{P(w|R=1,Q)}{P(w|R=0,Q)}$$

因此，需要估計的參數為所有 p_i 和 u_i，共 $2|V|$ 個.

但 $\sum_{i=1}^{V} p_i = \sum_{i=1}^{V} u_i = 1$ 為定值，故 free parameters 共 $(2|V|-2)$ 個.

(2) 已知 D_j 與 query Q 無關，因此估計出來的 $P(D_j|R=0,Q)$ 應盡可能大。
套用 MLE 去估計 $u_i = P(W_i|R=0,Q)$：

$$\underset{u_i}{\operatorname{argmax}} \prod_j P(D_j|R=0,Q) = \underset{u_i}{\operatorname{argmax}} \sum_j \log P(D_j|R=0,Q).$$

令 $x_{ij} = c(W_i, D_j)$，$x_{1j}+x_{2j}+\cdots+x_{vj} = m_j = |D_j|$.

使用 multinomial 的假設 $\Rightarrow P(D_j|R=0,Q) = \frac{m_j!}{\prod_i x_{ij}!} \prod_i u_i^{x_{ij}}$.

可將問題寫成：

$$\underset{u_i}{\operatorname{argmax}} \sum_j \log P(D_j|R=0,Q) = \underset{u_i}{\operatorname{argmax}} \sum_j \log \frac{m_j!}{\prod_i x_{ij}!} \prod_i u_i^{x_{ij}}$$

$$= \underset{u_i}{\operatorname{argmax}} \sum_j \left(\log m_j! - \sum_i \log x_{ij}! + \sum_i x_{ij} \log u_i \right).$$

由於每個字都是整個 vocabulary 中的其中一個字，故 $\sum_i u_i = \sum_i P(W_i|R=0,Q) = 1$.
機率總和必為 1.

套用 Lagrange multiplier，為了尋找多元函數在其變數受到一個或多個條件的約束時的極值，將一個有 n 個變數與 k 個約束條件的最佳化問題轉換成一個有 (n+k) 個變數的方程組的解的問題：

$$\mathcal{L} = \sum_j \left(\log m_j! - \sum_i \log x_{ij}! + \sum_i x_{ij} \log u_i \right) + \lambda \left(1 - \sum_i u_i \right).$$

各項的偏微分必須為 0：

$$\frac{\partial \mathcal{L}}{\partial u_i} = \sum_j \frac{x_{ij}}{u_i} - \lambda = 0 \Rightarrow \sum_j x_{ij} = \lambda u_i \quad \text{——①}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 1 - \sum_i u_i = 0 \Rightarrow \sum_i u_i = 1. \Rightarrow \lambda = \sum_i \lambda u_i \quad \text{——②}$$

將①、②合併整理化簡後可得：

$$\lambda = \sum_i \lambda u_i = \sum_i \sum_j x_{ij} = \sum_j \sum_i x_{ij} = \sum_j |D_j|.$$

$$\Rightarrow u_i = \frac{\sum_j x_{ij}}{\lambda} = \frac{\sum_j c(W_i, D_j)}{\sum_j |D_j|}.$$

故，$P(W|R=0,Q) = \frac{\sum_j c(W_i, D_j)}{\sum_j |D_j|}$ //.

(3) 令 $D_Q = Q$ → 視為 the only example of a relevant document.

因此，估計出來的 $P(D_Q|R=1,Q)$ 應盡可能大.

套用 MLE 去估計 $P_i = P(W_i|R=1,Q)$:

$$\underset{P_i}{\operatorname{argmax}}\ P(D_Q|R=1,Q) = \underset{P_i}{\operatorname{argmax}}\ \log P(D_Q|R=1,Q).$$

令 $X_i = c(W_i, D_Q)$，$X_1+X_2+\cdots+X_V = m = |D_Q|$.

使用 multinomial 的假設 $\Rightarrow P(D_Q|R=1,Q) = \frac{m!}{\prod_i X_i!}\prod_i P_i^{X_i}$.

可將問題寫成:

$$\underset{P_i}{\operatorname{argmax}}\ \log P(D_Q|R=1,Q) = \underset{P_i}{\operatorname{argmax}}\ \log \frac{m!}{\prod_i X_i!}\prod_i P_i^{X_i}$$

$$= \underset{P_i}{\operatorname{argmax}}\ \log m! - \sum_i \log X_i! + \sum_i X_i \log P_i$$

由於每個字都是整個 vocabulary 中的其中一個字，故 $\sum_i P_i = \sum_i P(W_i|R=1,Q) = 1$.

↓ 機率總和必為 1.

套用 Lagrange multiplier，為了尋找多元函數在其變數受到一個或多個條件的約束時的極值，將一個有 n 個變數和 k 個約束條件的最佳化問題轉換成一個解有 (n+k) 個變數的方程組的解的問題:

$$\mathcal{L} = \left(\log m! - \sum_i \log X_i! + \sum_i X_i \log P_i\right) + \lambda\left(1 - \sum_i P_i\right)$$

各項的偏微分必須為 0:

$$\frac{\partial \mathcal{L}}{\partial P_i} = \frac{X_i}{P_i} - \lambda = 0 \Rightarrow X_i = \lambda P_i \quad \text{——} ③$$

$$\frac{\partial \mathcal{L}}{\partial \lambda} = 1 - \sum_i P_i \Rightarrow \sum_i P_i = 1 \Rightarrow \lambda = \sum_i \lambda P_i \quad \text{——} ④$$

將 ③、④ 合併整合化簡後可得:

$$\lambda = \sum_i \lambda P_i = \sum_i X_i = |D_Q|.$$

$$\Rightarrow P_i = \frac{X_i}{\lambda} = \frac{c(W_i, D_Q)}{|D_Q|}$$

故，$P(W|R=1,Q) = \frac{c(W_i, Q)}{|Q|}$. ★

(4). 套用以上結果：

$$Score(Q,D) = \sum_{w \in V} c(w,D) \log \frac{P(w|R=1,Q)}{P(w|R=0,Q)}$$

$$= \sum_{w \in V} c(w,D) \left[\log \frac{c(w,Q)}{|Q|} - \log \frac{\sum_{j=1}^{n} c(w,D_j)}{\sum_{j=1}^{n} |D_j|} \right]$$

- 由於 $c(w,Q)$ 在算詞頻，故該項就像 TF 的方法
- 由於 $\sum_{j=1}^{n} |D_j|$ 在算 collection 的大小，而 $\sum_{j=1}^{n} c(w,D_j)$ 在算 word 在 collection 中的數量，兩項的比值就類似 IDF 的算法（皆為 word 數量除以 collection 大小的比值取 log）.

故，此方法有 TF 跟 IDF 的效果，但由於 score 中沒有根據 document 的大小做修正，故沒有 document length normalization 的功能。

(5) 因為題目的式子未考慮文章的長度，故效果會比有考慮該問題的 BM25 差.

方括號中第一項 $c(w,Q)$ 為 0 時，會出現 $\log 0 = -\infty$，故可在分子加上 $\frac{1}{2}$，另外由於分母是由 $|Q| = \sum_{w \in V} c(w,Q)$ 算來的，故須同時在分母加上 $\frac{|V|}{2}$ 以同步，也可以在第一項分子加上 $\frac{1}{2}$，並在分母加上 $\frac{|V|}{2}$，以解決 zero probability problem.

另外，若將整個式子除以 document 的長度 $|D|$，則會使得較長的 document 有較多的調整，進而達到 document length normalization 的效果.

套用以上 trick 後，新的 score 為：

$$\hat{Score}(Q,D) = \sum_{w \in Q} \frac{c(w,D)}{|D|} \left[\log \frac{c(w,D) + \frac{1}{2}}{|Q| + \frac{|V|}{2}} - \log \frac{\sum_{j=1}^{n} c(w,D_j) + \frac{1}{2}}{\sum_{j=1}^{n} |D_j| + \frac{|V|}{2}} \right]$$

註：在這樣的 normalization 效果下，同一篇 document 重覆兩次所得到的新 document 的分數會一樣，也就是，假設 $\bar{D} = D + D$，由於 $c(w,\bar{D}) = 2c(w,D)$，$|\bar{D}| = 2|D|$ 有相同的比例關係，而方括號內的項與對於 $\bar{D}$ 和 D 是相同的，因此可以推得 $\hat{Score}(Q,\bar{D}) = \hat{Score}(Q,D)$，可避免長度較長的文章得到不應得的高分數.