假设我们有一个文件集 (Collection)，其中包含了如下几个文件 (Document)：词频tf_{t_i} 词项t_i

- $D_1 = \{2\ t_1,\quad 3\ t_2,\quad 4\ t_5\}$
- $D_2 = \{1\ t_2,\quad 2\ t_3,\quad 2\ t_5,\quad 2\ t_6\}$
- $D_3 = \{1\ t_2,\quad 1\ t_3,\quad 5\ t_4\}$

我们在这个文件集中定义一个查询 $Q = \{\ 2\ t_1,\quad 1\ t_4\}$

倒排表 (Fichier inversé)

我们可以先根据以上信息得出倒排表：

1. 先按照词项（Term）升序排序，再按照 docID 升序排序
2. 将同一篇文档中多次出现的词合并，记录其*出现在文档中的次数*，即*文档频率* df
3. 作出如下*倒排表*

词项 Term	出现在文档中的次数 df	Posting lists (出现的文档ID和词频 tf)
t_1	1	$[D_1, 2]$
t_2	3	$[D_1, 3], [D_2, 1], [D_3, 1]$
t_3	2	$[D_2, 2], [D_3, 1]$
t_4	1	$[D_3, 5]$
t_5	2	$[D_1, 4], [D_2, 2]$
t_6	1	$[D_2, 2]]$

布尔模型

布尔模型是二值匹配，也即 0 和 1。我们可以分别对查询 Q 的*查询词项*做*析取*（Disjunction，$\vee$）运算与*合取*（Conjunction，$\wedge$）运算，观察其结果。

$$RSV(D_i, Q) = \begin{cases} 1, & \text{if 在文档中} D_i \text{中满足} Q \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

在如上述所示的例题中，我们有

Disjunction	Conjunction
$RSV(D_1, t_1 \vee t_2) = 1$	$RSV(D_1, t_1 \wedge t_2) = 0$
$RSV(D_2, t_1 \vee t_2) = 0$	$RSV(D_2, t_1 \wedge t_2) = 0$
$RSV(D_3, t_1 \vee t_2) = 1$	$RSV(D_3, t_1 \wedge t_2) = 0$

向量模型

对数词频权重 tf

在一个文件 D 中包含若干个词项 t，我们可以统计一个词项 t_i ***在该文件中出现的次数***，称为词频 tf_{t_i}。我们可以通过词频 tf 来确定和比较不同词项与该文件的相关程度。但是，原始的词频 tf 值不太合适（比较关系不***应该是线性的***），所以我们通常使用***对数***表示：

$$w_{t,d} = \begin{cases} 1 + \log_{10} tf_{t_i}, & \text{if } tf_{t_i} > 0 \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

对数逆文档频率权重 idf

想象一下，如果一个词项 t_i 在许多的文档中都出现了，那么我们使用这个词项 t_i 查询到想要的文档的概率就会变小。所以对于这种***常见词项***，我们希望赋予***低权重***；相反地，对于这种***罕见词项***，我们希望赋予***高权重***。因此，我们作出如下定义

$$\text{文档频率} df_{t_i} \rightarrow \text{出现词项} t_i \text{的文档数目} \tag{3}$$

$$\text{逆文档频率} idf_{t_i} = \log_{10}(\frac{\text{文档总数} N}{df_{t_i}}) \tag{4}$$

> ***注意***⚠️：
>
> - 我们使用 $\log_{10}(N/df_{t_i})$ 来抑制 idf 的作用。
> - 只有在对于两个以上查询词的 Query 时，idf 才会影响排序结果

tf.idf 权重

我们通常将以上两种结合起来，形成 ***tf.idf***：

$$w_{t,d} = tf \times idf = (1 + \log_{10} tf_{t_i}) \times \log_{10}(\frac{\text{文档总数} N}{df_{t_i}}) \tag{5}$$

这种关于词项的权重 $\mathbf{w}$ 既可以用在文档 $\mathbf{w}_{(t,d)}$ 中，也可以用在查询 $\mathbf{w}_{(t,q)}$ 中。我们采用 ***SMART 标记***，即 ***ddd.qqq***来定义 $\mathbf{w}_{(t,d)}$ 和 $\mathbf{w}_{(t,q)}$。

除了 **tf.idf** 权重外还有很多***其他机制（详见课件）***。

$$RSV(d,q)=\sum_{t\in q}\mathbf{w}_{(t,d)}\times\mathbf{w}_{(t,q)} \tag{6}$$

概率模型

概率：事件A的概率是指*事件A发生的可能性*，记作 $P(A)$

条件概率：已知事件A发生的条件下，事件B发生的概率称为A条件下B的条件概率，记作 $P(B\mid A)$

独立事件概率：事件A、B同时发生的概率等于A、B的概率乘积，若 $P(AB)=P(A)P(B)$

PRP模型

$$RSV(q,d)=\frac{P(R\mid d)}{P(NR\mid d)}=\prod_{i=1}^{n}\frac{P((t_i=x_i)\mid R)}{P((t_i=x_i)\mid NR)} \tag{7}$$

BIM模型

对于该模型，$x_i=\{0,1\}$

$$RSV(q,d)=\log\prod_{t_i\in q}\frac{P(t_i=1\mid R)\times P(t_i=0\mid NR)}{P(t_i=1\mid NR)\times P(t_i=0\mid R)} \tag{8}$$

下面给出一个*估算方法*:

	文档	相关 Pertinent	不相关 Non-Pertinent	总计
词项出现	$t_i=1$	r	$n-r$	n
词项不出现	$t_i=0$	$R-r$	$N-n-R+r$	$N-n$
总计		R	$N-R$	N

在上表中：

- r：包含 t_i 的*相关文档*的个数
- n：包含 t_i 的文档个数
- R：*相关文档*的总个数（相关文档：应该是包含所有查询词之一的文档，∨合取）
- N：文件集 (Collection) 中文件的总个数

$$RSV(q,d)=\sum\log\frac{\frac{r+0.5}{R-r+0.5}}{\frac{(n-r+0.5)}{N-n-R+r+0.5}} \tag{9}$$

往年题目

Donner l'impact des procédures ci-dessous, en termes de rappel et la précision : (***ANSWERED BY CHATGPT***)

1. utilisation de ***lemmatisation des mots*** ;
    - The use of ***word lemmatization*** can increase precision by reducing variations of words to their base form, but may decrease recall if inflections that carry semantic meaning are lost. 使用 ***word lemmatization*** 可以通过将单词的变体减少到其基本形式来提高精确度，但如果丢失了带有语义意义的词形变化，则可能会降低召回率。
2. utilisation des ***synonymes*** ;
    - The use of ***synonyms*** can increase recall by finding documents that use synonyms for the query terms, but may decrease precision if the synonyms are not semantically related to the query.
3. utilisation de la ***position*** des termes dans les documents;
    - Using the ***position*** of terms in the documents can increase both recall and precision by giving more weight to terms that appear closer to each other in the document, indicating they are semantically related. 使用文档中术语的 ***position*** 可以提高召回率和准确率，方法是为文档中彼此更接近的术语赋予更多权重，表明它们在语义上相关。
4. utilisation des ***expressions*** ;
    - The use of ***expressions*** can increase precision by allowing more complex and specific queries, but may decrease recall if expressions are not used in the relevant documents. 使用 ***expressions*** 可以通过允许更复杂和特定的查询来提高精确度，但如果相关文档中未使用表达式，则可能会降低召回率
5. utilisation des ***représentations distribuées de mots*** à la place des mots simples ;
    - The use of ***distributed word representations*** can increase both recall and precision by capturing semantic meaning beyond just the individual words themselves. 使用***分布式词表示***可以通过捕获单个词本身之外的语义来提高召回率和准确率
6. utilisation de ***l'opérateur `AND`*** dans le modèle booléen
    - The use of the ***`AND` operator*** in the boolean model can increase precision by requiring all query terms to be present in a document, but may decrease recall if relevant documents only contain some of the query terms. 在布尔模型中使用 ***`AND` 运算符*** 可以通过要求所有查询词都出现在文档中来提高精确度，但如果相关文档仅包含一些，则可能会降低召回率 的查询词。

Répondre par ***Vrai*** ou ***Faux*** et justifiez votre réponse. (ANSWERED BY ***CHATGPT***)

1. Dans un système de de recherche d'information qui utilise des ***n-grammes*** de caractères plutôt que des mots, la ***racinisation*** n'est ***pas nécessaire***.
    - TRUE. In information retrieval systems that use character n-grams rather than words, stemming may no longer be necessary, since n-grams can capture the form in which words appear in documents. However, stemming can still be useful to reduce the morphological variation of a word and can help improve the recall and precision of the system. 在使用字符 n-gram 而不是单词的信息检索系统中，可能不再需要词干提取，因为 n-gram 可以捕获单词在文档中出现的形式。 然而，词干提取仍然有助于减少单词的形态变化，并有助于提高系统的召回率和准确率。
2. La ***racinisation*** permet-elle ***d'améliorer*** le ***rappel*** ou la ***précision*** ou les deux ? justifier ?
    - TRUE. Racinization can improve both recall and precision by reducing the vocabulary mismatch and reducing the number of false positive hits. 词根化可以通过减少词汇不匹配和减少误报次数来提高召回率和准确率。
3. ***L'analyse sémantique latente*** permet de récupérer des documents pertinents même si ces documents n'ont aucun terme en commun avec la requête.
    - TRUE. ***Latent semantic analysis*** (LSA) can retrieve relevant documents even if they do not have any terms in common with the query by identifying underlying relationships between terms in a corpus of documents. 潜在语义分析 (LSA) 可以通过识别文档语料库中术语之间的潜在关系来检索相关文档，即使它们与查询没有任何共同的术语。
4. Un document ne comportant ***aucun terme*** de la requête ***ne peut pas être pertinent*** pour cette requête?
    - False. The absence of query terms in a document does not necessarily imply that the document is not relevant to the query. The ***Latent semantic analysis*** can help retrieve relevant documents even if they don't have any terms in common with the query. 文档中没有查询词并不一定意味着该文档与查询不相关。潜在语义分析可以帮助检索相关文档，即使它们与查询没有任何共同的术语。
5. La ***normalisation par la longueur*** des documents vise à éviter que les documents courts ne soient classés trop haut
    - FALSE. Length normalization aims to avoid biasing the results towards longer documents, not shorter ones. It can eliminate the impact of length differences between long documents and short documents on relevance. 长度归一化旨在避免将结果偏向较长的文档，而不是较短的文档。 它可以消除长文档和短文档之间的长度差异对相关性的影响。
6. En général, comme les documents d'un ensemble de résultats sont listés par ordre décroissant de pertinence estimée, ***la précision diminue*** à mesure que ***le rappel augmente***.
    - FALSE. Accuracy may increase as recall increases, but this depends on the algorithms and models used for relevance ranking. However, in general there is a trade-off between precision and recall, where an increase in one may cause a decrease in the other. 准确性可能会随着召回率的增加而增加，但这取决于用于相关性排名的算法和模型。 然而，一般来说，精确率和召回率之间存在权衡，其中一个的增加可能导致另一个的减少。
7. Le représentation en ***sac de mots*** permet de ***capturer*** (représenter) le ***sens*** des mots?

- FALSE. The bag-of-words representation does not capture the meaning of the words, but rather the frequency of the words in a document. This representation treats words as individual tokens and does not take into account the context in which they appear, so the meaning of the words is lost. 词袋表示不捕获单词的含义，而是捕获文档中单词的频率。 这种表示将单词视为单独的标记，没有考虑它们出现的上下文，因此单词的含义丢失了。

8. L'utilisation des ***représentation distribuées de mots*** (Word embedding) permet de sélectionner des documents pertinents même si ces derniers n'ont ***aucun terme*** en commun avec la ***requête***.
   - TRUE. The use of word embeddings can capture the semantic meaning of words and their relationships, allowing for relevant documents to be selected even if they don't contain common terms with the query. 词嵌入的使用可以捕获词的语义及其关系，允许选择相关文档，即使它们不包含查询的常用术语。
9. Un système de recherche d'information basé sur un modèle de langue de type ***bigrammes*** renvoie ***deux listes différentes*** pour les requêtes suivantes : "Information retrieval" et "Retrieval information"
   - FALSE. A language model based on bigrams would return the same list of documents for both the queries. The order of the terms wouldn't matter in this model as it considers pairs of adjacent words (bigrams) rather than individual words. 基于二元语法的语言模型将为两个查询返回相同的文档列表。 术语的顺序在此模型中无关紧要，因为它考虑的是成对的相邻单词（二元组）而不是单个单词。
0. A votre avis, quel modèle de RI (parmi ceux étudiés en cours) serait le ***plus approprié*** pour rechercher des ***Tweets*** ? Justifiez ? On suppose qu'un Tweet est formé d'un message où les mots se répètent rarement.
   - In my opinion, the most appropriate IR model for searching tweets would be the ***n-gram*** character model. This is because tweets often have informal language, misspelled words, and slang, which can result in low recall using word-based models like the bag of words or TF-IDF. 在我看来，最适合搜索推文的 IR 模型是 ***n-gram*** 字符模型。 这是因为推文通常包含非正式语言、拼写错误的单词和俚语，这可能导致使用基于单词的模型（如词袋或 TF-IDF）的低召回率。