Compression et index inversé

Accès et Recherche d'Information

Exercice 1

Dans la collection de Wikipédia français il y a $N = 1,349,539$ documents et le nombre total d'identifiants de documents dans l'index inversé construit sur cette collection est $373,163,766$*.

1.1 Quel est le nombre de *bits* minimal qu'il faudrait pour coder tous les identifiants dans l'index inversé ? Quelle serait la taille non-compressée de l'index sur le disque ?

Pour un terme assez fréquent, ce type de codage n'est pas très optimal puisqu'on va utiliser le même nombre de bits pour coder les très nombreux identifiants de documents dans lesquels ce terme apparaît, alors que les écarts entre les numéros de ces identifiants seraient assez faibles. Donc au lieu de coder les identifiants des documents, on pourrait économiser beaucoup d'espace Le codage optimal des entiers naturels non-nuls très souvent utilisé est le codage gamma d'Elias. Ce codage se fait en deux étapes : (1) chaque entier est d'abord codé en binaire et le bit de poids fort est enlevé, (2) la taille du code restant est exprimé en code unaire et elle est ajoutée à gauche du code binaire tronqué.

1.2 Représenter les entiers naturels suivants en utilisant le codage gamma.

nombre n	binaire b_n	binaire privé du bit de poids fort $\bar{b}_n$	longueur de $\bar{b}_n$ en unaire	code gamma γ_n
1				
5				
17				
1021				

1.3 Décoder la séquence d'entiers suivants codés suivant le codage gamma 11101011100011010

*Ce chiffre correspond au cas où on ne filtrerait pas les documents par un anti-dictionnaire, une stratégie appliquée dans certains systèmes de recherche actuels.

1.4 Quelle est la longueur en nombre de bits du code gamma associé à un entier naturel non-nul, n ?

EXERCICE 2

On suppose que la présence d'un mot dans un document est le résultat du tirage aléatoire avec remise dans l'ensemble des M_y différents types de mots de la collection de documents.

2.1 Soit X_k la variable aléatoire binaire correspondant au résultat de l'évènement; E_k*: le* k^{eme} *mot le plus fréquent a été tiré lors du tirage.* Montrer que $P(X_k = 1) = \frac{1}{k \times \ln(M_y)}$. *Indication. pour simplifier les calculs on suppose que la somme de la série harmonique* $\sum_{i=1}^{r} \frac{1}{i}$ *est approximée par* $\sum_{i=1}^{r} \frac{1}{i} \approx \int_1^r \frac{dx}{x} = \ln r$

2.2 Soit d un document de longueur n (en nombre de mots). On cherche à calculer le nombre moyen de fois où le k^{me} mot le plus fréquent de la collection apparaît dans d. Montrer que la somme de n variables $X_{k,i}, i \in \{1, ..., n\}$ indépendantes exprimant chacune le résultat de l'évènement E_k, S_k, suit une loi binomiale dont on précisera les paramètres. En déduire l'espérance de S_k.

2.3 Pour un document de la collection de Wikipédia, quel serait donc le nombre moyen d'apparition du mot le plus fréquent, *de*, dans un document de taille $n = 416$ mots? Comparer ce résultat avec le nombre moyen d'apparition observé de *de* dans un document de la collection (i.e. 27).

2.4 On considère que la taille de tous les documents de la collection de Wikipédia est $n = 416$ mots, combien de mots apparaissent au moins une fois dans tous ces documents?

On va maintenant appliquer le codage gamma pour coder les écarts entre les identifiants de documents. Pour cela on va d'abord trier les termes du vocabulaire du plus fréquent au moins fréquent. On divise ensuite la liste triée par des bloques de longueur fixe L, tel que les mots du premier bloque soient ceux qui apparaissent dans tous les documents de la collection, ceux du deuxième bloque apparaissent dans un document sur deux, ... , et les mots du i^{me} bloque apparaissent dans 1 document sur i. Si dans les listes inversées, on trie les identifiants des documents par ordre croissant, l'écart entre les identifiants des documents du i^{me} bloque est, avec le principe précédent, d'au plus i.

2.5 Calculer la taille totale pour coder les écarts entre les identifiants des documents contenant les termes du i^{me} bloque avec le codage gamma, au pire cas (c'est à dire dans le cas où ces écarts seraient tous égaux à i)?

2.6 Montrer que la taille *maximale* nécessaire en nombre de bits pour coder l'index inversé en entier est approximativement :

$$\frac{N \times L}{\ln 2} \times \ln^2 \frac{V}{L}$$

Indication. On fera le même genre d'approximation que précédemment en supposant que $\sum_{i=1}^{r} \frac{\ln(i)}{i} \approx \int_1^r \frac{\ln(x)dx}{x} = \frac{1}{2}\ln^2 r$. *On rappelle que* $\forall a \in \mathbb{R}_+^*, \log_2 a = \frac{\ln a}{\ln 2}$.

2.7 Par quel facteur peut-on ainsi espérer comprimer l'index inversé de la collection de Wikipédia sur le disque, est ce que l'approximation précédente est pertinente dans ce cas?
On rappelle que $L = 31$ *et* $V = M_{Nor} = 604,444$.

2.8 Lorsque l'on applique l'algorithme de compression gamma, on trouve que la taille de l'index de le collection de Wikipédia sur le disque est de 456 méga-octets. Est-ce que ce résultat est en concordance avec la taille de l'index estimée à la question précédente ? Sinon, qu'est ce qui expliquerait la différence entre les deux tailles présumée et trouvée ?