Modalité de déroulement du projet

Par groupe de 5

Présentation du sujet

Il s'agit d'approcher le fonctionnement d'un moteur de recherche textuel. Pour ce faire vous disposez :
D'un ensemble de textes (impérativement en anglais à cause des accents et des restrictions du langage C à ce sujet)

1. D'un sous-système qui vous permet d'indexer l'ensemble de vos fichiers textes, c'est le moteur d'indexation qui imite le « web scraping » de google
2. D'un sous-système moteur de recherche qui vous permet de rechercher de l'information dans l'ensemble de vos fichiers textes et de classer par pertinence les textes qui répondent le mieux à la recherche. Dans notre système il remplace l'algorithme de Pagerank de google.

Cas d'usage : l'exemple à la page suivant vous permettra de mieux cerner les attentes du projet.
Nous avons 3 textes dans notre base de textes (ceux-là sont en français)

1. La coupe du roi de Thulé (fichier_1)
2. Sanglot (fichier_2)
3. Chanson d'automne (fichier_3)

Principe de fonctionnement

Principe de fonctionnement du moteur d'indexation

Notre moteur d'indexation doit indexer tous les mots de chaque texte et produire pour chacun de ces textes un compte rendu d'indexation contenant tous les mots du texte avec le nombre d'occurrences de chacun de ces mots dans le texte. Les comptes rendu d'indexation seront suffixés par .CRI
Les comptes rendus d'indexation doivent respectivement donner pour le mot sanglot :
fichier_1.CRI nombre d'occurrences de sanglot = 1
fichier_2.CRI nombre d'occurrences de sanglot = 3
fichier_3.CRI nombre d'occurrences de sanglot = 1
Nota : les fichier CRI pourront être des fichiers gérés en mode binaire car ils peuvent être composés de structures contenant un champ texte et un champ nombre. Le mot en majuscule ou en minuscule doit être compté de la même manière. Les mots au singulier et au pluriel doivent être comptés au même endroit.

Principe de fonctionnement du moteur de recherche

Il se base sur les comptes rendus d'indexation des fichiers précédemment indexés.
Par exemple notre moteur de recherche sur la recherche de l'information « sanglot » devra faire apparaitre le fichier2 en premier, suivi des deux autres.
En revanche sur la recherche d'information multi critères « flot » et « onde » c'est le fichier1 qui devra apparaitre (unique puisque flot et onde n'apparaissent pas dans les deux autres textes).
Nota : dans une version minimale on se contentera d'une recherche de type ET entre plusieurs mots. Les fichiers CRI devront être chargés en mémoire avant de commencer les traitements.

Votre ensemble de fichiers texte

Sur le net vous trouverez les scripts des séries telles que les Simpson, Adam's family ou encore des textes de vos chanteurs/poètes préférés.
Pour avoir un échantillon pertinent il faut compter au moins 20 textes.

La coupe du roi de Thulé Louise Ackermann	Sanglot Rainer Maria Rilke	Chanson d'automne Paul Verlaine
[…] Comme pour emporter une dernière ivresse, Il te vida d'un trait, étouffant ses sanglots, Puis, de son bras tremblant surmontant la faiblesse, Te lança dans les flots. D'un regard déjà trouble il te vit sous les ondes T'enfoncer lentement pour ne plus remonter : C'était tout le passé que dans les eaux profondes Il venait de jeter. Et son coeur, abîmé dans ses regrets suprêmes, Subit sans la sentir l'atteinte du trépas. En sa douleur ses yeux qui s'étaient clos d'eux-mêmes Ne se rouvrirent pas. […]	Sanglot, sanglot, pur sanglot ! Fenêtre, où nul ne s'appuie ! Inconsolable enclos, plein de ma pluie ! C'est le trop tard, le trop tôt qui de tes formes décident : tu les habilles, rideau, robe du vide !	Les sanglots longs Des violons De l'automne Blessent mon coeur D'une langueur Monotone. Tout suffocant Et blême, quand Sonne l'heure, Je me souviens Des jours anciens Et je pleure Et je m'en vais Au vent mauvais Qui m'emporte Deçà, delà, Pareil à la Feuille morte.

Exigences concernant le projet

Exigences minimales

Techniques

1. Programme en langage C
2. Environnement de développement Linux
3. Programme découpé en plusieurs modules
4. Utilisation de l'allocation dynamique lors des chargements de fichiers CRI.

Fonctionnelles

5. Indexer automatiquement un ensemble de fichier (majuscules, minuscules, singulier, pluriel)
6. Rechercher et classer par pertinence en fonction de la recherche les fichiers qui répondent le mieux aux critères.

Exigences fonctionnelles avancées

7. Chargement automatique d'un ensemble de fichiers à partir d'un répertoire
8. En cas d'égalité calcul de la proximité des mots (cas des recherches ET). Attention il faut alors revenir sur le fichier texte initial.
9. Recherche avec tolérance orthographique (proposition de correction comme sur le moteur google)
10. Classement avec affichage de la partie du texte contenant les mots recherchés