L3 – 2010-2011 – Projet – Programmation objet

Reconnaissance de la langue d'un texte écrit

Objectif

L'objectif du projet est d'implanter un système capable
- de détecter dans quelle langue est écrit un texte d'au moins une page ;
- d'apprendre à reconnaitre une nouvelle langue quand on lui présente un texte d'au moins quelques pages écrit dans cette langue.

Méthode

Voici le principe de la méthode à utiliser. Il existe des calculs statistiques qui permettent d'estimer si deux textes sont écrits dans la même langue ou non. On recense les mots de chacun des deux textes, et pour chaque mot, le **nombre d'occurrences**, c'est-à-dire le nombre de fois où il apparait. Exemple pour le texte *Et d'autres peuples ont d'autres habitudes* :

Et 1
d 2
autres 2
peuples 1
ont 1
habitudes 1

Pour le texte *Mais les uns ont aidé les autres* :

Mais 1
les 2
uns 1
ont 1
aidé 1
autres 1

Ensuite, on calcule le **produit scalaire** des deux listes :

autres $2 \times 1 = 2$
ont $1 \times 1 = 1$
(les autres mots donnent 0)
total $2 + 1 = 3$

En général, la formule $\frac{p}{l_1 l_2} > s$ est vraie si et seulement si les deux textes sont écrits dans la même langue. Dans cette formule, p est le produit scalaire, l_1 et l_2 sont la taille des textes en nombre de mots, et s est un seuil proche de $4{,}5 \times 10^{-3}$ (vous pourrez fixer sa valeur après vos expériences). Dans l'exemple, $\frac{3}{8 \times 7} = 5{,}4 \times 10^{-2} > 4{,}5 \times 10^{-3}$, et effectivement les deux phrases sont écrites dans la même langue. (C'est un peu par chance, car cette méthode ne marche pas bien pour des textes aussi petits. Elle ne marche pas non plus avec les langues où on n'écrit pas d'espace entre les mots.)

Pour reconnaître la langue d'un texte, on applique la formule pour comparer le texte à un jeu de textes dont on dispose déjà et dont on connaît la langue.

Apprendre à reconnaître une nouvelle langue consiste à préparer le calcul pour un texte écrit dans cette langue.

Pour vos tests, vous pouvez utiliser les textes que vous voulez du moment qu'ils sont assez longs, qu'ils n'ont pas de balises, qu'ils utilisent tous le même jeu de caractères, et que vous savez dans quelle langue ils sont écrits. On vous fournit six textes dont deux en français, deux en anglais et deux en portugais. Vous ajouterez au moins une autre langue.

Algorithme de délimitation des mots : un mot est une séquence ininterrompue de lettres. Les lettres sont les caractères reconnus par la séquence \p{L} dans un Pattern Java.

Travail demandé

Écrire des classes Java qui proposent les fonctionnalités suivantes :

- apprendre à reconnaitre une nouvelle langue lorsque l'utilisateur fournit un texte et le code de la langue ; pour les codes des langues, respecter la norme ISO 639-1 ;
- donner la langue d'un texte s'il s'agit d'une des langues déjà apprises.

Le système doit lever des exceptions clairement compréhensibles au moins dans les cas suivants :

- s'il n'arrive pas à trouver pas la langue du texte,
- si le texte vérifie la formule pour plusieurs langues différentes,
- s'il n'arrive pas à ouvrir un fichier contenant un texte.

Faire une classe Main qui illustre les fonctionnalités en allant chercher dans un fichier les données d'apprentissage (codes des langues et textes correspondants), puis les noms des fichiers à traiter.

Bien respecter les principes de programmation objet, l'encapsulation et la réutilisabilité, l'abstraction et l'héritage. Réutiliser des classes disponibles. Soigner la lisibilité du code. Produire une documentation pour l'utilisateur et une documentation pour le développeur (voir ci-dessous dans le format de rendu).

Faire des tests. Dans les tests, utiliser au moins une autre langue de votre choix, écrite dans l'alphabet latin. Commenter les tests.

Format de rendu

Le projet est à faire par binôme. Si vous êtes seul(e), contactez les enseignants : un seul projet individuel sera admis.

Rendre le projet par courrier électronique dans une archive au format .zip, au plus tard le 23 janvier 2011 à 23 h 59 **(date limite remise au mercredi 26 janvier à 23 h 59)**, aux deux adresses eric.laporte@univ-paris-est.fr et mlandsnet@yahoo.fr, en indiquant dans le champ sujet du message : Projet L3 et les noms des auteurs.

Voici le nom des répertoires et fichiers qui doivent être contenus dans l'archive .zip :

- un répertoire **src** contenant l'ensemble des sources (.java) dans leur(s) paquetage(s) ;
- un répertoire **classes** contenant l'ensemble des classes (.class) correspondant aux sources, dans leur(s) paquetage(s) ;
- un répertoire **docs** contenant trois documents au format PDF :
  - la documentation pour l'utilisateur **user.pdf** contenant une description de l'application, comment la compiler, l'exécuter, l'utiliser, et une liste des bugs connus ;
  - la documentation pour le développeur **dev.pdf** contenant
    - une description des structures de données choisies,
    - une description de chaque classe ;
  - la documentation des tests **tests.pdf** contenant une description des tests effectués et un commentaire.

Bonus

Si vous avez plus de temps à consacrer à ce projet, voici quelques idées.

- Sauvegarder le résultat de l'apprentissage dans un fichier de données.
- Récupérer le résultat de l'apprentissage depuis le fichier de sauvegarde.
- Variantes : avant le calcul statistique, remplacer les majuscules par des minuscules ; ou utiliser, au lieu des mots, les séquences de 2 lettres incluses dans des mots (Ma 1, ai 2, is 1, le 2, es 3, un 1, ns 1, on 1, nt 1, id 1, dé 1, au 1, ut 1, tr 1, re 1), ou encore les séquences de 3 lettres…