Université Pierre et Marie Curie

UE: Statistique et informatique (LI323)

Année scolaire : 2013/2014

Professeur chargé de TD/TME :

Nicolas Baskiotis

Etudiants :

Rémi Cadène n°3000693

Joël Fieux-Herrera n°3003174

Sommaire

README

Caractéristiques d'entrées

-Corpus :

Non vide,
Tout en minuscules,
Sans lettres accentuées,
Sans signes de ponctuations,

-Mots :

Non vide,
Tout en minuscules,
Sans lettres accentuées,
Sans signes de ponctuations,
Ne commence pas par un espace,
Ne fini pas par un espace.

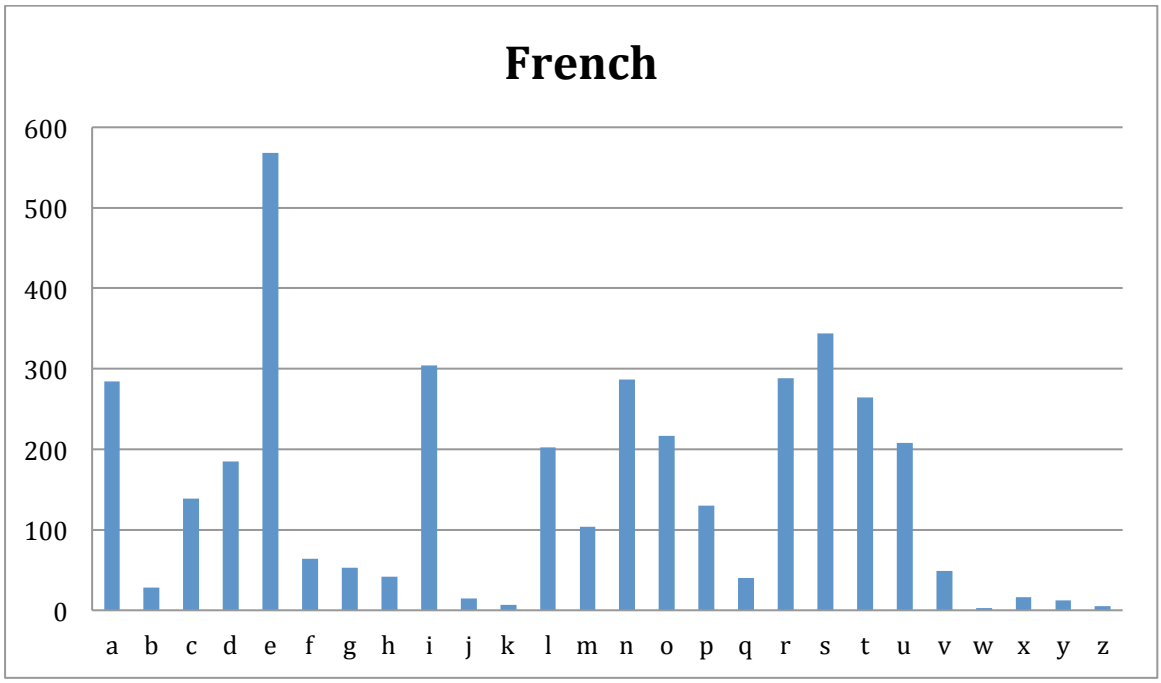
Afin de déterminer la langue d'un mot, il faudra :

- ajouter des corpus en format *.txt* dans le dossier corpus
- créer le corpus général à l'aide de la classe *static CorpusFactory*
- analyser le corpus général obtenu avec la *méthode analyse()*
- utiliser la méthode *guessLanguage(mot)* qui retourne le nom du corpus associé

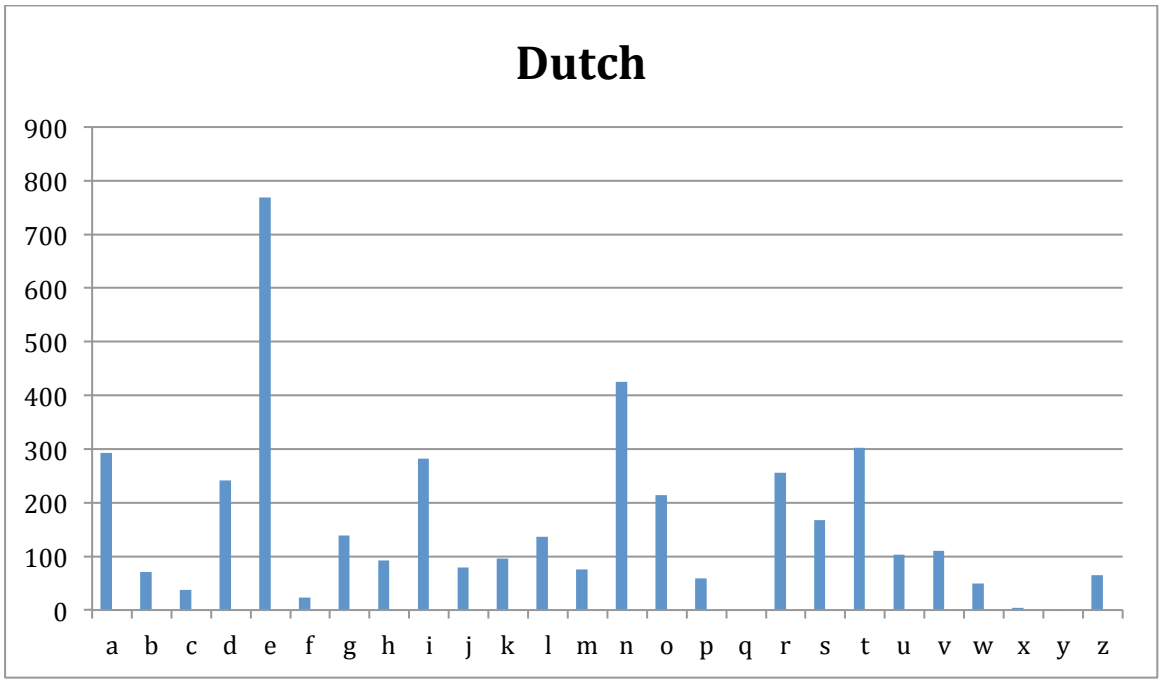
1) Construction d'un modèle de langage

Question 1 :

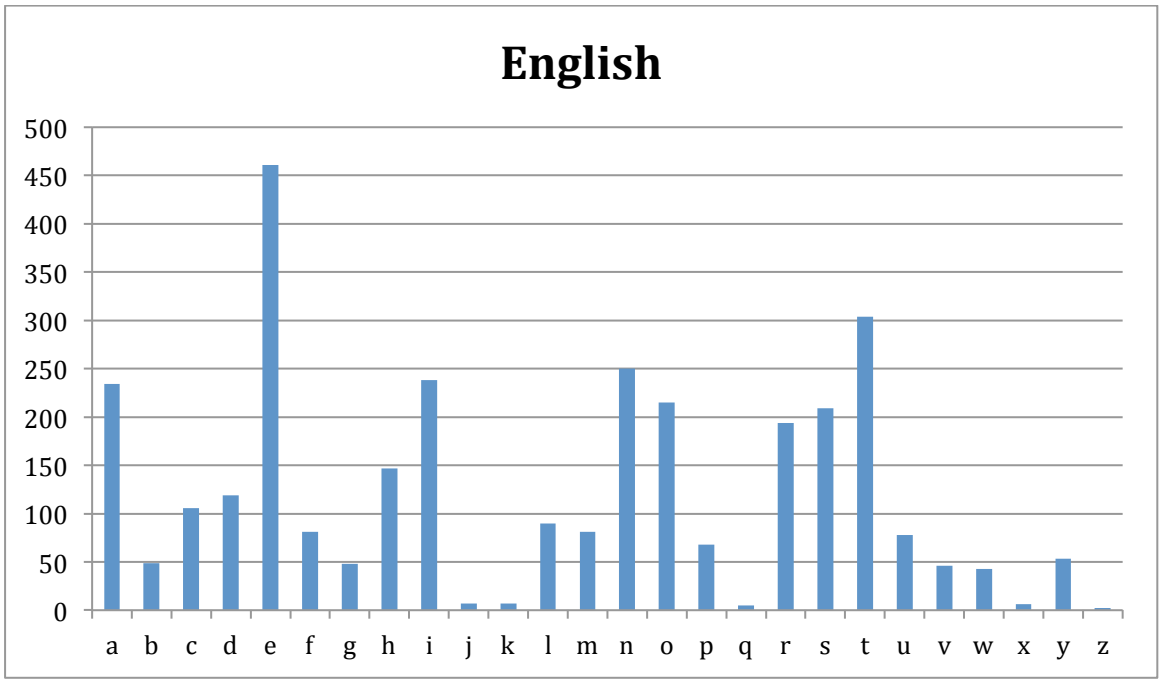
Afin de compter le nombre de fois qu'une lettre w est utilisée, nous parcourons le corpus en comptant le nombre d'occurrence d'une lettre sans prendre en compte les espaces.



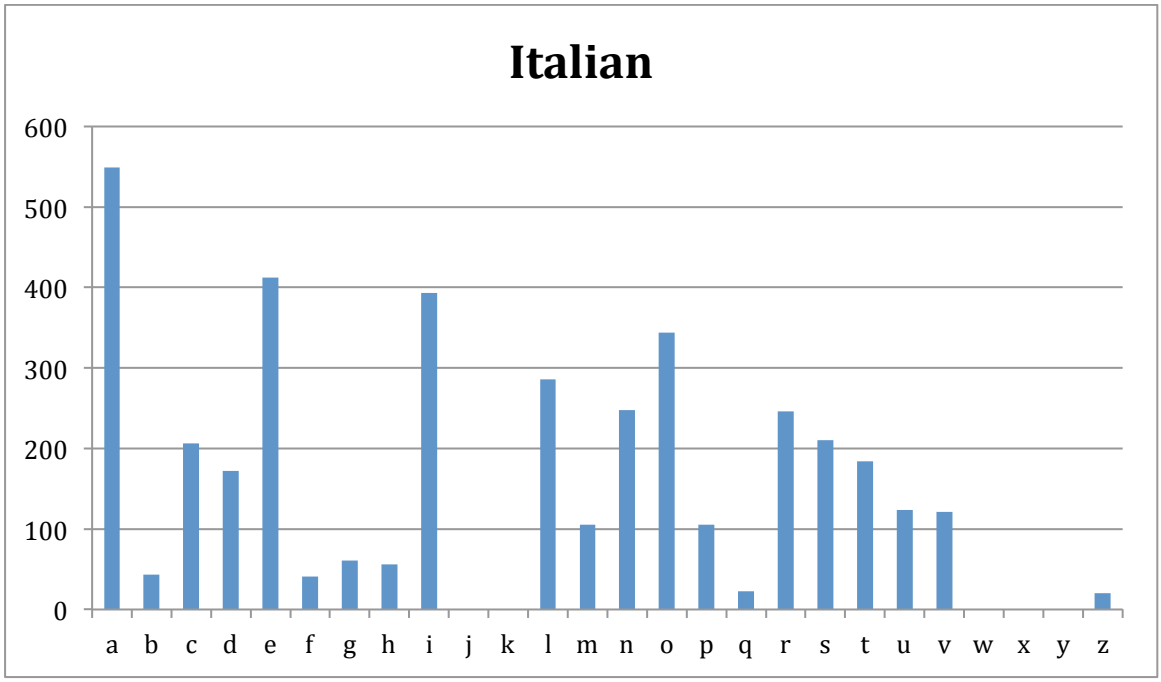
Nombre de lettres du corpus « french »: 3858



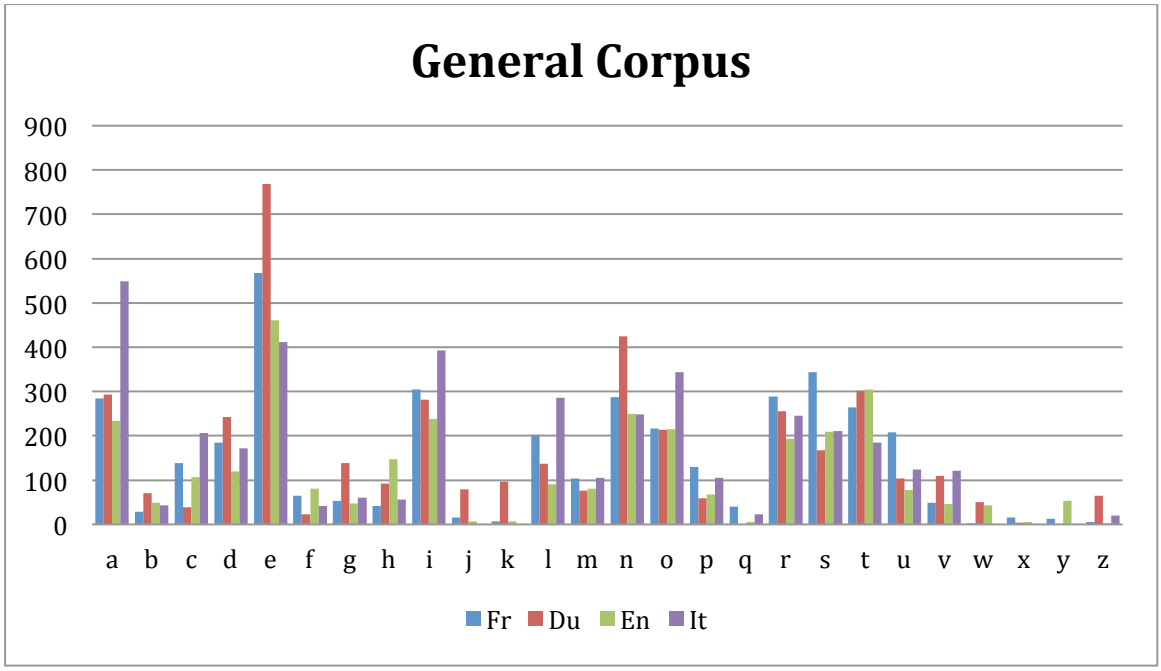
Nombre de lettres du corpus « dutch » : 4094



Nombre de lettres du corpus « english »: 3141



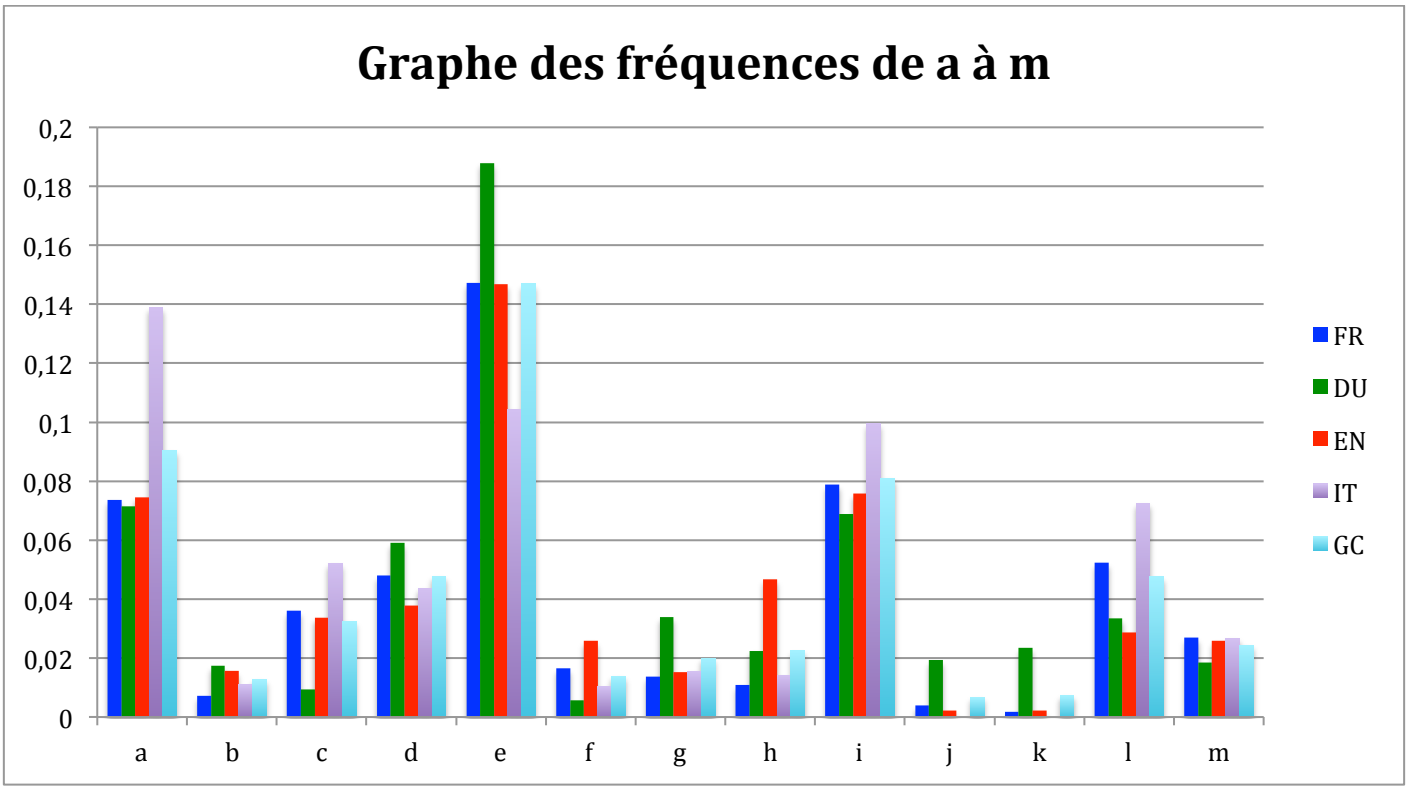
Nombre de lettres du corpus « italian »: 3949

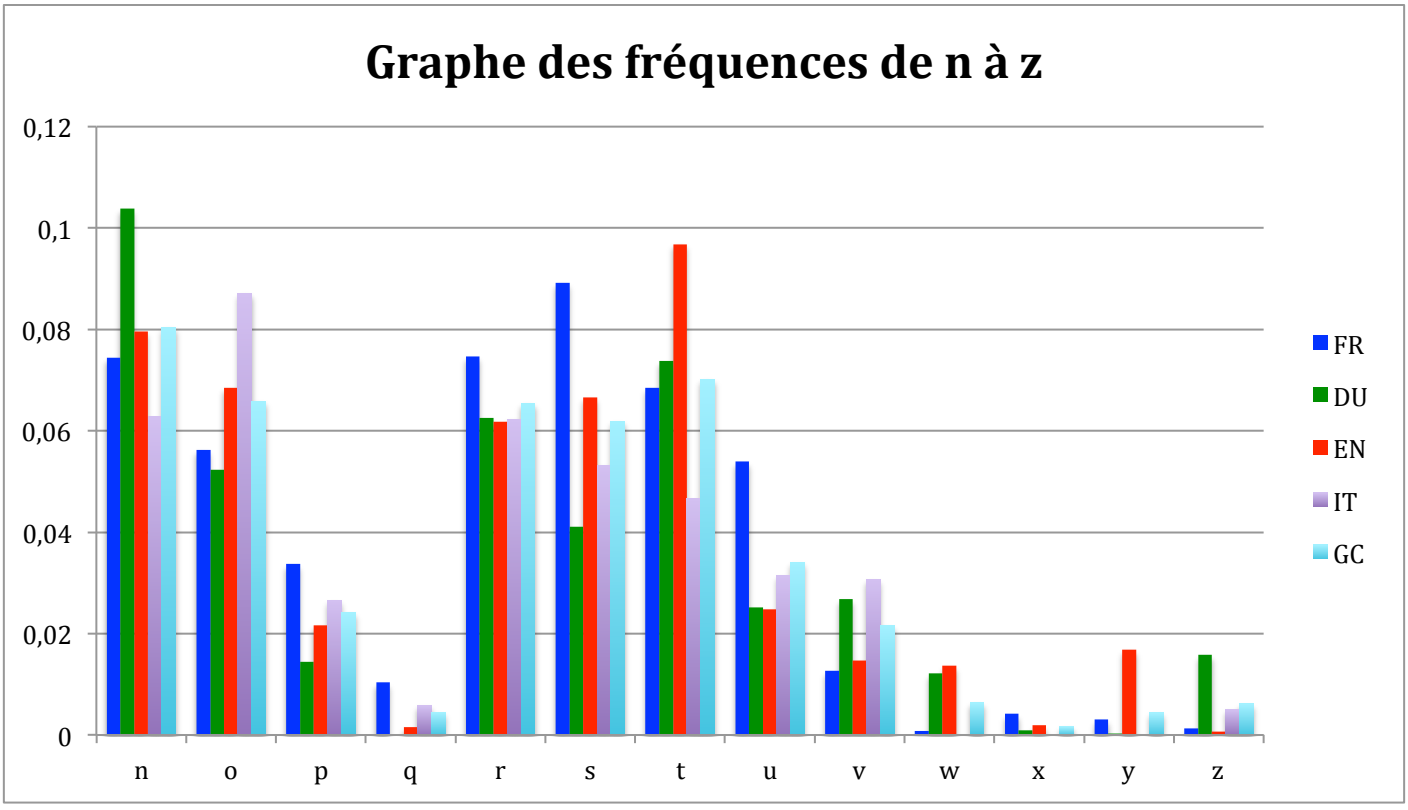


Nombre de lettres du corpus « General Corpus »: 15042

Question 2 :

Afin de calculer la probabilité d'une lettre sachant une langue, nous effectuons le ratio entre le nombre d'occurrence de la lettre et le nombre de caractère total du corpus.





Question 3 :

Afin de calculer la probabilité d'observer le mot anglais "statistics" dans le corpus "english", c'est à dire que le mot appartienne à cette langue, nous avons utilisé la formule suivante :

$$p(\text{statistics} \mid \text{anglais}) = p(s \mid \text{anglais})^3 * p(t \mid \text{anglais})^3 * p(a \mid \text{anglais}) * p(i \mid \text{anglais})^2 * p(c \mid \text{anglais})$$
$$= 3.85 * 10^{-12}$$

De même pour la probabilité d'observer le mot anglais "probability" sur un corpus "general" composé de quatre autres : "french", "english", "deutch", "italian".

$$p(\text{probability}) = p(p) * p(r) * p(o) * p(b)^2 * p(a) * p(i)^2 * p(l) * p(t) * p(y)$$
$$= 1.57 * 10^{-12}$$

2) Prédiction de la langue d'un mot

Question 4 :

Expliquez pourquoi il n'est pas nécessaire de déterminer la probabilité p(w).
On a la formule suivante : p(l/w) = p(w/l) * p(l) / p(w)
Or p(w) est constante, car p(w) correspond à la probabilité que ce mot appartienne aux quatre corpus, il ne dépend pas de la langue choisie dans p(l/w).
On notera aussi que p(l) est constante, car correspond à la probabilité que la langue "l" soit choisie. Notre programme possède quatre corpus, alors p(francais) = p(anglais) = $p(l_i)$ = 1/4.

Question 5 :

Dans le cas où on cherche seulement la fonction permettant de déterminer la langue d'un mot, il nous suffira de calculer la probabilité du mot sachant une langue pour chaque corpus, puis de comparer les valeurs obtenues en choisissant la langue associée au corpus de la plus grande.

Question 6 :

Les résultats obtenus lors de l'évaluation du système sont les suivants:

ERROR: president : french au lieu de english
ERROR: fatta : english au lieu de italian
ERROR: daar : italian au lieu de dutch
ERROR: peter : french au lieu de english
ERROR: che : english au lieu de italian
ERROR: chocolate : italian au lieu de english
ERROR: thanks : dutch au lieu de english
ERROR: mean : dutch au lieu de english

Les performances de notre programme associé à la fonction d'erreur donnée se calcule en fonction du nombre d'erreur (8) et du nombre de réponses (17).
l = 0.53

On notera que certain mot possède la même orthographe dans différent corpus. Dans la liste ci dessus "president" est étiqueté comme appartenant au corpus "english", alors qu'il pourrait appartenir au corpus "french", de même pour "chocolate". Cela peut fausser le taux d'erreur, surtout si la liste est petite.

Question 7 :

En prenant en compte la probabilité à priori p(l), les résultats obtenus sont les suivants :

ERROR: president : french au lieu de english
ERROR: fatta : english au lieu de italian
ERROR: daar : italian au lieu de dutch
ERROR: peter : french au lieu de english
ERROR: statistics : french au lieu de english
ERROR: che : english au lieu de italian
ERROR: chocolate : italian au lieu de english
ERROR: thanks : dutch au lieu de english
ERROR: mean : italian au lieu de english

l = 0.47

On obtient une baisse de la performance du programme suite à l'ajout d'une réponse fausse "statistics" considéré comme appartenant à "french" au lieu d'"english". Par ailleurs, les corpus observés ne sont pas assez représentatif des langues associées, c'est à dire qu'ils ne contiennent pas assez de mots et que leur taille, qui est déterminante dans le calcul de p(l), ne nous permet pas d'analyser les résultats obtenus. Il nous faudrait donc améliorer le modèle.

3) Amélioration du modèle

- Augmenter la taille des données disponibles :

1. L'intérêt d'agrandir la base de test est d'augmenter le nombre de mots associés aux différentes langues, donc d'accroître la capacité d'analyse de notre programme.

2. A priori plus la taille de l'ensemble d'estimation est grande et représentative des langues associées au corpus, plus les performances sont grandes. Par ailleurs l'utilisation d'un alphabet de 26 lettres (sans accents) et le traitement des corpus avant analyse afin de correspondre à celui-ci ne nous permet pas de fournir une analyse efficace. Par exemple, un "ç" serait suffisant pour augmenter considérablement la probabilité d'appartenance du mot à la langue française.

- Utiliser un modèle plus compliqué :

1. Avec ce nouveau modèle la probabilité d'un mot sachant la langue s'écrit de la façon suivante :
   $p(w/l) = \prod p(w_i/w_{i-1}/l)$

2. Afin d'adapter le modèle à notre programme, nous allons considérer que $p(w_1/w_0/l) = p(w_1)$ et que $p(w_i/w_{i-1}/l)$ est égal à la fréquence du couple (w_{i-1}, w_i) dans le corpus l.

3. Nous avons essayé notre fonction de prédiction correspondante au modèle (stratégie "double lettre") avec les probabilités p(l) égales (stratégie "corpus égaux") et deux corpus plus représentatifs : "frenchPlus" et "englishPlus".
   Nos résultats sur la liste de mots de l'exercice précédent sont les suivants :

   ERROR: president : french au lieu de english
   ERROR: peter : french au lieu de english
   ERROR: chocolate : italian au lieu de english
   l = 0.82

   On remarque que les trois mots ci dessus appartiennent aussi aux langues trouvées ce qui augmente les performances réelles de notre programme jusqu'à l=1 pour cette liste de mot.

- Autre possibilité d'amélioration du modèle :

Afin d'améliorer notre modèle, nous aurions par ailleurs pu prendre en compte les terminaisons des mots des corpus étudiés. En italien, les mots se finissent plus fréquemment par un "e", un "a" ou un "i", tandis qu'en anglais souvent par "ing", et français par "e" ou "s".