MASTER IIA (Natural Language Processing : NLP)

Mini-Projet 3 : Etiquetage Morphosyntaxique

L'étiquetage morphosyntaxique est une opération qui s'insère dans la chaîne d'analyse du langage naturel et qui a pour objectif de **lever les ambigüités lexicales possibles**

Elle consiste à attacher à chacun des mots d'un corpus une étiquette donnant des informations morphosyntaxiques sur ce mot : Sa catégorie grammaticale mais aussi d'autres informations comme par exemple le mode pour un verbe. Un jeu d'étiquettes est préalablement choisi soigneusement puis tout le problème est d'attacher à chaque mot du corpus l'étiquette correcte (**on suppose alors qu'il n'y a qu'une étiquette possible par mot**).

Objectif : On veut écrire un étiqueteur morphosyntaxique à la **BRILL**. Cet étiqueteur est fondé sur des règles apprises à partir **d'un corpus dont on sait que l'étiquetage est correct**. Les phases de l'algorithme de construction sont les suivantes :

Phase I : A partir du corpus d'apprentissage C, on construit un lexique L des mots de ce corpus associés à leur étiquette la plus fréquente

Exercice 1 :

- Ecrivez une fonction apprendre **Etiquettes (corpus)** qui a pour paramètre un corpus étiqueté et **qui retourne un lexique donnant pour chaque mot l'ensemble des étiquettes rencontrées avec le nombre de leurs occurrences.** Le corpus se présente sous forme d'une liste de couples (mot, étiquette) et le lexique sous forme d'un dictionnaire dont chaque entrée est elle-même un dictionnaire.
- Ecrivez ensuite une fonction **Choix_Etiquette (liste_etiquette)** qui a pour paramètre un dictionnaire dont les clés sont des étiquettes et les valeurs correspondantes un nombre d'occurrences et **qui retourne l'étiquette la plus fréquente dans le dictionnaire. (Quand il y a plusieurs solutions possibles, on choisit la première qui arrive).**
- A partir des deux fonctions précédentes, écrivez une fonction **Apprendre_lexique(corpus)** qui a pour paramètre un corpus étiqueté et **qui retourne un lexique donnant pour chaque mot l'étiquette la plus fréquente selon ce corpus.**

Exemple : Soit un corpus d'apprentissage étiqueté artificiel « **C** » :

- **C = [('la', 'DET'), ('belle', 'ADJ'), ('ferme', 'V'), ('la', 'DET'), ('porte', 'N'), ('.', 'PONCT'), ('la', 'DET'), ('belle','ADJ'), ('fille', 'N'), ('est', 'COP'), ('ferme','ADJ'), ('.', 'PONCT'), ('une','DET'), ('belle','N'), ('la','PRO'), ('porte','V'), ('.', 'PONCT'), ('une','DET'), ('belle','ADJ'), ('fille','N'), ('la','PRO'), ('porte','V'), ('.', 'PONCT'), ('il','PRO'), ('la','PRO'), ('porte','V'), ('.', 'PONCT')]]**
- Le lexique obtenu est alors : **L = {'est' : 'COP', 'la' : 'PRO', 'belle' : 'ADJ', 'porte' : 'V', '.' : 'PONCT', 'fille' : 'N', 'une' : 'DET', 'ferme': 'ADJ', 'il' : 'PRO'}**

Phase II : A l'aide du lexique L, on ré-étiquette le corpus d'apprentissage et on obtient un corpus C0, qui contient bien entendu des erreurs.

Exercice 2 :

- Ecrivez une fonction **Reetiqueter_corpus (corpus, lexique)** qui a pour paramètre un corpus étiqueté et un lexique de mots étiquetés **et qui retourne un nouveau corpus étiqueté où les étiquettes du corpus initial ont été remplacées par celles fournies par le lexique.**

Exemple : Dans l'exemple précédent, nous obtenons un nouveau corpus C0 qui comporte 6 erreurs :

- **C0 = [('la', 'PRO'), ('belle', 'ADJ'), ('ferme', 'ADJ'), ('la', 'PRO'), ('porte', 'V'), ('.', 'PONCT'), ('la', 'PRO'), ('belle', 'ADJ'), ('fille', 'N'), ('est', 'COP'), ('ferme', 'ADJ'), ('.', 'PONCT'), ('une', 'DET'), ('belle', 'ADJ'), ('la', 'PRO'), ('porte', 'V'), ('.', 'PONCT'), ('une', 'DET'), ('belle', 'ADJ'), ('fille', 'N'), ('la', 'PRO'), ('porte', 'V'), ('.', 'PONCT'), ('il', 'PRO'), ('la', 'PRO'), ('porte', 'V'), ('.', 'PONCT')]**

- Ecrivez une fonction **Comparer (corpus reference, corpus test)** qui permet de comparer un corpus étiqueté à tester par rapport à un corpus de référence avec les mêmes mots et avec des étiquettes correctes. La fonction doit retourner le nombre d'erreurs dans le corpus à tester

Phase III : Pour corriger ces erreurs, on va chercher des règles de modification des étiquettes en fonction de leur contexte droit ou gauche, ou des deux.

- Une règle se présente comme un **quadruplet : (étiquette à modifier, nouvelle étiquette, contexte gauche, contexte droit)**.
- Le contexte gauche (droit) est l'étiquette qui doit être immédiatement à gauche (à droite) de l'étiquette à modifier pour que la règle s'applique.
- Ce contexte peut être indéfini (on peut l'indiquer par le symbole *).

En essayant toutes les possibilités, on cherche **la règle R0** qui corrige le maximum d'erreurs **dans C0**. Dans notre exemple, **la règle R0 est : ('PRO', 'DET', '*', 'ADJ').** On l'applique et on obtient **un nouveau corpus C1**.

Exercice 3: Maintenant, dans un contexte général :

- Ecrivez une fonction **Appliquer_regle(corpus, ancienne_etiquette, nouvelle_etiquette, contexte_gauche, contexte_droit)** qui a pour paramètre un corpus étiqueté et une règle de ré-étiquetage exprimée par les 4 paramètres (ancienne_etiquette, nouvelle_etiquette, contexte_gauche, contexte_droit) **et qui retourne un nouveau corpus où chaque occurrence de « ancienne_etiquette » est remplacée par « nouvelle_etiquette » quand elle est précédée de l'étiquette « contexte_gauche » et suivie de l'étiquette « contexte_droit ».** On peut remplacer un seul des deux contextes par * qui indique que celui-ci est ignoré.

- Ecrivez une fonction **Choisir_regle(corpus reference, corpus test)** **qui retourne la règle de modification d'étiquettes** sous forme d'un quadruplet qui permet de corriger un maximum d'erreurs dans corpus test ,compte tenu de corpus « reference »

Phase IV : **On itère l'opération précédente à partir de C1 (le nouveau corpus obtenu dans la phase précédente) et ce, jusqu'`a ce que le nombre d'erreurs soit au-dessous d'un certain seuil. On obtient ainsi une liste de règles : R0, R1, . . , Rn.**

Exercice 4 :

- Ecrivez une fonction **Generer_regles (corpus_reference, corpus_test, seuil)** qui **retourne une liste de règles** permettant de corriger le corpus étiqueté « corpus_test » pour que le taux d'erreurs par rapport au nombre d'étiquettes soit au-dessous de « seuil », compte tenu du corpus de référence «corpus_reference».

Phase V : **On considère maintenant un corpus à étiqueter. On commence par l'étiqueter à l'aide du lexique L (cela sous-entend que tous les mots du corpus sont dans le lexique) puis on lui applique dans l'ordre les règles R0, R1, . . . , Rn. Le résultat est le corpus étiqueté qu'on obtient après l'application de la règle Rn.**

Exercice 5 :

- Ecrivez une fonction **Etiqueter_corpus(corpus_test, lexique, liste_regles)** qui prend en paramètres un corpus brut : « corpus_test », un lexique de mots étiquetés : « lexique» permettant d'initialiser l'étiquetage du corpus brut et une liste de règles de ré-étiquetage à appliquer « liste_regles » et **qui retourne un corpus étiqueté après application des règles.**