Sujet 1: détection automatique de langue d'un texte

October 23, 2022

1 Description du projet

1.1 Démarche

Le but de ce projet est de construire un programme permettant de détecter automatiquement la langue d'un texte.

Pour parvenir à faire une telle chose, nous allons utiliser les probabilités pour fabriquer ce qu'on appelle un *modèle de langue*. Le travail se fait donc en deux étapes:

- une étape d'*apprentissage*, pendant laquelle nous allons construire un *modèle de langue* pour plusieurs langues en observant des textes rédigés dans ces langues;
- une étape d'*exploitation*, pendant laquelle nous allons utiliser ces modèles de langue pour détecter la langue d'un mot.

Ce projet devra être rédigé en Python.

Une attention particulière sera portée à l'utilisation de la programmation orientée objet dans ce projet, au travers de classes et d'objets dédiés facilitant le travail. La partie algorithmique de ce projet est assez légère, et sera bien guidée et expliquée, il faut porter son attention sur la structure du code, sa portabilité et sa potentielle réutilisabilité.

C'est une occasion d'implémenter un algorithme de *machine learning* utilisant uniquement des probabilités, tout en mettant en pratique les notions de classes et d'objets vus en cours.

1.2 Déroulé du projet

Le projet va se dérouler sur le restant du semestre.

Lors de la séance du 24/10, nous allons lancer le projet. Au programme:

- Finalisation des groupes.
- Présentation du projet.
- Explication des algorithmes à implémenter.
- Présentation en détail de ce qui est attendu (code, rapport).
- Lancement du projet (la 2e heure sera du travail de groupe).

Nous aurons une deuxième séance dédiée au projet le 14 novembre, afin de faire un point sur l'avancement de chaque groupe.

1.3 Travail à réaliser

Ce projet va être évalué au travers de trois rendus:

- le code source contenant:
  - un programme permettant de créer un modèle de langue à partir d'un texte.
  - un programme permettant d'exploiter des modèles de langue pour détecter la langue d'un mot.
  - un fichier `README` expliquant comment utiliser votre programme et quelles sont ses dépendances logicielles.
- un rapport dans lequel:
  - vous présenterez l'architecture de votre code, et justifierez vos choix.
  - vous détaillerez les points critiques/intéressants de votre implémentation.
  - vous présenterez les résultats d'expériences sur votre programme.
- une présentation de 10 minutes, pendant laquelle:
  - vous présenterez le problème de détection de langue.
  - vous présenterez votre programme et vos choix architecturaux.
  - vous montrerez, si possible via une démonstration en direct, son fonctionnement.

Le code source, ainsi que les rapports, doivent être envoyés à `maxime.raynal@univ-grenoble-alpes.fr` au plus tard le 20 novembre.

Les soutenances de fin de projet se feront (*à priori*) la séance du 28 novembre .

2 Détecter automatiquement une langue

Dans ce chapitre, nous allons voir en détail les algorithmes à mettre en oeuvre pour détecter automatiquement un language.

Nous allons utiliser les notations suivantes:

- Nous considérons un mot w de taille n. Nous notons ses lettres $w_1 w_2 \dots w_n$.
- Nous considérons une collection de langues L de taille m, avec $L = (l_1, l_2, \dots, l_m)$.
- Pour chacune des langues de L, nous disposons d'un corpus d'entraînement. Nous notons C_i le corpus de la langue l_i.

Le but est donc de savoir, en observant w, à quelle langue de L il appartient.

Pour cela, nous allons utiliser un modèle probabiliste, basé sur l'observation que la fréquence d'apparition d'une lettre dépend de la langue dans laquelle on écrit. Par exemple, un texte écrit en français contiendra beaucoup plus de 'e' qu'un texte rédigé en anglais ou en italien.

2.1 Utiliser les corpus pour calculer $P(a|l)$

Nous pouvons, pour une langue donnée l_i, calculer la probabilité d'apparition de chaque lettre dans l_i. Étant donnés une lettre a et une langue l_i, nous noterons la probabilité d'apparition de a dans l_i par $P(a|l_i)$ (P de a sachant l_i).

Nous allons utiliser pour cela les corpus (de simples textes), et faire l'approximation suivante:

$$P(a|l_i) = \frac{\text{\# d'apparitions de } a \text{ dans } C_i}{\text{\# total de caractères dans } C_i}$$

2.2 Caculer $P(l|w)$

Maintenant que nous savons comment calculer $P(a|l)$, pour n'importe quels lettre a et langue l, voyons comment calculer $P(l|w)$. C'est à dire, la probabilité que la langue d'un mot w soit l.

Pour cela, nous supposons que la probabilité que w appartienne à une langue l donnée ne dépend que des probabilités d'apparition de chacune de ses lettres dans l, et que ces probabilités sont mutuellement indépendantes. Nous notons $P(w|l)$ la probabilité d'observer le mot w dans la langue l. Nous en déduisons que:

$$P(w|l) = \prod_{1 \leq i \leq n} P(w_i|l)$$

Mais ce que nous voulons calculer, ce n'est pas $P(w|l)$. Nous souhaitons calculer $P(l|w)$.

Mais nous savons que (d'après Bayes)

$$P(l|w) = \frac{P(w|l) \cdot P(l)}{P(w)}$$

Or, dans notre cas, on se rend compte que les termes $P(l)$ et $P(w)$ n'ont pas trop de sens. En effet, $P(w)$ représente la probabilité de tomber sur w, parmi tous les mots imaginables. On peut considérer qu'on n'est pas trop biaisés, et qu'on tire chaque mot avec la même probabilité. Ainsi, on peut négliger ce terme s'il est constant. Il en est de même pour le terme $P(l)$. On peut considérer que chaque langue contient le même nombre de mots, et a la même chance d'être tirée. Ce terme vaudrait alors autant pour toutes les langues.

Ce qui nous donne, après simplification

$$P(l|w) = \lambda \cdot P(w|l)$$

avec λ une constante.

Nous cherchons à déterminer, via notre détecteur de langue, ceci:

$$l^\star = \text{argmax}_{l \in L} P(l|w)$$

Et comme les deux termes $P(l|w)$ et $P(w|l)$ sont proportionnels, on a donc

$$l^\star = \text{argmax}_{l \in L} P(w|l)$$

et donc

$$l^\star = \text{argmax}_{l \in L} \prod_{1 \leq i \leq n} P(w_i|l)$$

.