Sujet 2: génération automatique de texte

October 23, 2022

1 Description du projet

1.1 Démarche

Le but de ce projet est de construire un programme permettant de générer automatiquement du 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* à partir de textes d'entraînement.
- une étape d'*exploitation*, pendant laquelle nous allons utiliser ce modèles de langue pour générer "aléatoirement" du texte.

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 corpus de textes.
  - un programme permettant d'exploiter le modèle de langue pour générer du texte aléatoirement.
  - 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 Génération aléatoire de texte

Dans ce chapitre, nous allons voir en détail les algorithmes à mettre en oeuvre pour générer automatiquement du texte.

Nous allons utiliser les notations suivantes:

- Nous considérons un corpus de textes $C = w_1 w_2 w_3 \dots w_n$, composé de n mots consécutifs.
- Nous notons W l'ensemble des mots apparaissant dans le corpus.

Nous allons utiliser ce corpus pour calculer, pour chaque paire de mots w, w' de W, la probabilité que w' soit le mot qui apparaisse après w. Nous notons cette probabilité $P(w'|w)$. On calcule cette probabilité comme ceci:

$$P(w'|w) = \frac{\text{\# de w' suivant un w}}{\text{\# total de w}}$$

.

2.1 Générer aléatoirement du texte

Une fois qu'on a calculé $P(w|w')$ pour chaque couple w, w' de W, on peut générer aléatoirement du texte. On commence par donner un premier mot de W à l'algorithme, qui génère un mot en fonction du mot précédent. Pour générer le $i^{\text{ème}}$ mot, on utilise le $i-1^{\text{ème}}$ mot (notons le w_{i-1}). On tire le $i^{\text{ème}}$ mot w avec une probabilité $P(w|w_{i-1})$, et ainsi de suite.