

## Exemple d'application pour les modèles de langage : algorithme prédictif pour l'aide à la saisie de texte

Dans cet exercice de travaux pratiques, il vous est proposé de construire un modèle de langage de type  $n$ -gram afin de prédire les mots en cours de saisie les plus probables. Cette probabilité est calculée en fonction de l'historique de la frappe (*i.e.* des mots précédents).

Pour cela, vous aurez d'abord besoin d'estimer un modèle de langage. Cela consiste dans un premier temps à calculer les probabilités d'apparition de chacun des  $n$ -grams du corpus d'apprentissage. Ici, nous utiliserons un modèle *trigram*. Il s'agit alors d'utiliser la formule suivante :

$$P(w_i | w_{i-2} w_{i-1}) = \frac{c(w_{i-2} w_{i-1} w_i)}{c(w_{i-2} w_{i-1})}$$

où  $c(.)$  correspond au nombre d'occurrences de  $'.'$  dans le corpus d'apprentissage.

Algorithmiquement c'est très simple :

1. calculer  $c(w_{i-2} w_{i-1})$  pour chaque bigram  $w_{i-2} w_{i-1}$
2. calculer  $c(w_{i-2} w_{i-1} w_i)$  pour chaque trigram  $w_{i-2} w_{i-1} w_i$
3. pour chaque trigram  $w_{i-2} w_{i-1} w_i$ , calculer  $P(w_i | w_{i-2} w_{i-1})$

On considèrera que le vocabulaire des modèles de langage correspond à l'ensemble des mots présents dans le corpus d'apprentissage.

➔ Une fois l'estimation de ces probabilités de trigrams réalisée, faites la même chose pour les bigrams et les unigrams.

➔ Tester vos modèles avec des trigrams ou des bigrams existants dans le corpus d'apprentissage.

➔ Tester vos modèles avec des trigrams ou des bigrams absents du corpus d'apprentissage (mais composés de mots de votre vocabulaire).

Vous avez pu constater qu'aucune probabilité n'est affectée aux  $n$ -grams absents du corpus d'apprentissage. Pourtant, il n'est pas raisonnable de considérer que ces  $n$ -grams ont une probabilité nulle.

Afin d'affecter une probabilité aux  $n$ -grams non vus dans le corpus d'apprentissage, il est généralement appliqué une technique dite de *discounting*.

Cette estimation sera réalisée en utilisant la technique simple (et loin d'être la plus efficace) de *discounting* dite de lissage additif (*additive smoothing*). Cette technique considère que n'importe quel  $n$ -gram a été vu au moins un (très) petit nombre de fois :

ce nombre est noté  $\delta$  avec généralement  $0 < \delta \leq 1$ . Ainsi la probabilité d'un trigramme devient :

$$P_{add}(w_i | w_{i-2}w_{i-1}) = \frac{\delta + c(w_{i-2}w_{i-1}w_i)}{\delta|V| + c(w_{i-2}w_{i-1})}$$

où  $|V|$  est le nombre de mots du vocabulaire.

➔ Une fois l'estimation de ces nouvelles probabilités de trigrams réalisée, faites la même chose pour les bigrams et les unigrams.

➔ Tester vos modèles avec des trigrams ou des bigrams existants dans le corpus d'apprentissage.

➔ Tester vos modèles avec des trigrams ou des bigrams absents du corpus d'apprentissage.

**À l'aide de ces modèles, construisez un programme qui propose, à chaque frappe, les 5 mots les plus probables au cours de la saisie.**