Sommaire

1 Les modèles de langues neuronaux **1**
1.1 Cas général 1
1.2 Modèle n-gram 1
1.3 Modèle Neural Network 2

1 Les modèles de langues neuronaux

1.1 Cas général

Définition 1.1 *Un modèle de langue est une distribution de probabilité d'une séquence de mots.*

Dans un modèle de langue, il s'agit d'attribuer une probabilité $\mathbb{P}$ pour chaque mot ou séquence de mots s dans une langue. Un modèle de langue va donc permettre de modéliser l'agencement (ou la distribution) des mots dans une langue.

Connaissant la probabilité de chaque mot, nous souhaitons prédire l'apparition d'un mot sachant une séquence de mots. On parle alors de méthode « générative ». Le fait de prédire des séquences de mots va nous permettre de générer du texte. Les applications de cette méthode sont diverses :

— traduction

— résumé

Autrement dit, quelle est la probabilité d'obtenir d'observer un mot ou un séquence sachant une séquence de mots déjà observée ? Mathématiquement, nous nous intéresserons à une probabilité conditionnelle.

Soit s une séquence de N mots, notés $m_1, \dots, m_N$. La probabilité d'observer la séquence s dans une langue dans l'absolue est calculée comme suit :

$$\begin{aligned}
\mathbb{P}(s) &= \mathbb{P}(m_1 \dots m_N) \\
&= \mathbb{P}(m_N | m_1 \dots m_{N-1}) \mathbb{P}(m_1 \dots m_{N-1}) \\
&= \mathbb{P}(m_N | m_1 \dots m_{N-1}) \mathbb{P}(m_{N-1} | m_1 \dots m_{N-2}) \mathbb{P}(m_1 \dots m_{N-2}) \\
&= \prod_{i=1}^{N} \mathbb{P}(m_i | m_1 \dots m_{i-1}) \text{ (récurrence)}
\end{aligned}$$

Si la séquence de mots est longue, cela devient rapidement coûteux en mémoire (taille de type factoriel). En effet, il faut être capable d'estimer la probabilité $\mathbb{P}(m_i | m_1 \dots m_{i-1})$. Nous sommes donc amenés à effectuer des approximations dans les calculs pour estimer ces probabilités. Ces différentes estimations conduisent à la définition de différents modèles de langues neuronaux.

Nous distinguons les modèles de langue suivants :

— les modèles n-grams

— les modèles Neural Network (NN)

1.2 Modèle n-gram

Dans un modèle n-gram, nous supposons que la probabilité d'apparition du mot $m - i$ ne dépend que de $n - 1$ prédécesseurs.

— Cas $n = 1$: Modèle unigram : $\mathbb{P}(s) = \prod_{i=1}^{N} \mathbb{P}(m_i)$

— Cas $n = 2$: Modèle bigram : $\mathbb{P}(s) = \prod_{i=1}^{N} \mathbb{P}(m_i | m_{i-1})$

— Cas $n = 3$: Modèle trigram : $\mathbb{P}(s) = \prod_{i=1}^{N} \mathbb{P}(m_i|m_{i-2}m_{i-1})$

— Cas $n > 3$: Modèle n-gram : $\mathbb{P}(s) = \prod_{i=1}^{N} \mathbb{P}(m_i|m_{i-(n-1)} \dots m_{i-1})$

Il est cependant très difficile de calculer ces probabilités dans l'absolue. Nous estimons alors ces probabilités sur un corpus de textes et nous supposons que le corpus de textes reflète la langue dans l'absolue. Nous devons donc disposer d'un très grand corpus de textes. Il s'agit là d'une approche statistique.

Etant donné que nous travaillons sur un corpus de textes fini, nous utilisons naturellement pour probabilité la mesure de comptage. Ainsi, le calcul de probabilité conditionnelle devient :

$$\mathbb{P}(m_i|m_{i-(n-1)} \dots m_{i-1}) = \frac{|m_{i-(n-1)} \dots m_{i-1}m_i|}{|m_{i-(n-1)} \dots m_{i-1}|}$$

Limitations : Etant donné que nous travaillons sur un corpus fini, nous avons une combinaison de mots finis. Il est possible qu'un mot qui n'apparaît pas dans le modèle. Sa probabilité d'apparition est donc nulle : $\mathbb{P}(m_k) = 0$. On parle de sparcité. Cette probabilité nulle pose problème : toute séquence de mots qui contient un mot qui n'est pas dans le corpus a une probabilité égale à 0 d'apparaître. Notre modèle reconnaît donc uniquement des séquences connus.

Pour palier ce problème et pouvoir généraliser à des séquences de mots non connus, nous pouvons effectuer un « lissage », qui consiste à attribuer une valeur de probabilité non nulle pour les mots n'apparaissant jamais dans le corpus.

1.3 Modèle Neural Network

Les modèles n-gram utilisent une approche statistique pour estimer les probabilités d'apparition d'une séquence de mots. Pour qu'elle soit efficace, il est nécessaire d'avoir un grand corpus de textes. Une autre approche de représentation des probabilités est d'utiliser les réseaux de neurones. L'idée est de capturer les liens (ou caractéristiques) que les mots peuvent avoir entre eux via un réseau de neurones. Ces liens sont représentés par les différentes connexions qui existent entre les neurones du réseau. On parle de « représentation distribuée ». Nous passons alors d'une représentation discrète à une représentation continue à travers un vecteur. Chaque mot est représentée par un vecteur dans $\mathbb{R}^m$. Cette représentation continue permet de représenter énormément de combinaisons possibles en modifiant très légèrement le vecteur qui représente la séquence de mots.