La pseudonymisation par l'IA en pratique

Après avoir vu dans les grandes lignes les étapes d'un projet de pseudonymisation grâce à l'IA, nous revenons plus en détails dans cette partie sur ces différentes étapes pour présenter les choix, arbitrages et préconisations techniques que nous avons tirés de nos travaux pour la création d'un moteur de pseudonymisation pour les décisions du Conseil d'État. Ceux-ci sont disponibles sur GitHub.

Formater ses données annotées

Afin de pouvoir utiliser les données annotées pour l'entraînement d'un algorithme d'apprentissage, **celles-ci doivent être converties dans un format spécifique**. Dans l'exemple ci-dessous, un document textuel (ici « Thomas CLAVIER aime beaucoup Paris. ») est alors structuré en un tableau, avec un mot par ligne, et deux colonnes, une pour le mot (ou *token*) et une pour l'annotation linguistique. Ce type de format s'appelle **CoNLL**.

Token	Label
Thomas	B-PER
CLAVIER	I-PER
aime	O
beaucoup	O
Paris	B-LOC

Plus particulièrement, nous utilisons le format IOB2, très commun pour les tâches d'apprentissage séquentiel comme la reconnaissance d'entités nommées, pour labéliser nos données. Ce format permet d'aider l'algorithme d'apprentissage à mieux repérer les entités. Le préfixe B- avant un label indique que le label est le début d'un groupe de mots, le préfixe I- indique que le label est à l'intérieur d'un groupe de mots, et le label O indique que le token n'a pas de label particulier. Il existe d'autres formats similaires à IOB2, tels que IOB/BIO, BILOU, et BIOES.

> **Le format CoNLL**
>
> CoNLL, pour « Conference on Natural Language Learning », est un format général, dont il existe de nombreuses versions, couramment employé pour les tâches de traitement du langage naturel, décrivant des données textuelles en colonne selon un nombre d'attributs (catégorie d'entité nommée, nature grammaticale, etc.). Le format IOB2 que nous utilisons est l'une des méthodes de labélisation du format CoNLL.

Il existe de très **nombreux logiciels ou solutions d'annotation de données textuelles** et les données annotées en sortie peuvent donc avoir différents formats (il existe en effet de multiples formats de données annotées). Pour transformer vos données annotées, un développement spécifique sera probablement nécessaire afin de les convertir au format IOB2, le format des données d'entrée de l'algorithme de reconnaissance d'entités nommées que nous avons choisi. Plusieurs exemples de fonctions et de librairies développées pour le Conseil d'État constitueront néanmoins un point de départ dans le répertoire GitHub de notre projet.

Entraîner son modèle

Dans le code que nous avons développé, nous utilisons la librairie Open Source Flair. Celle-ci permet en effet d'utiliser de nombreux modèles de langage, par exemple les modèles Flair, Bert et CamemBERT et même de combiner plusieurs de ces modèles. **Un modèle de langage permet pour chaque mot d'obtenir une représentation vectorielle** (ou *embedding*). Ces embeddings sont ensuite passés à un classificateur BiLSTM-CRF qui attribue à chaque mot une des classes du jeu de données d'entraînement.

L'entraînement d'un tel classificateur nécessite de choisir la valeur d'un certain nombre d'**hyper-paramètres**. Les hyper-paramètres sont les paramètres de l'algorithmes qui sont fixés avant