Détection d'entités quasi-dupliquées dans une base de connaissances avec PIKA

Maxime Prieur*, Guillaume Gadek*
Bruno Grilheres*

*Airbus Defence and Space
Elancourt, France
guillaume.gadek [at] airbus.com

Résumé. Cet article explore l'utilisation des modèles de réseaux de neurones adaptés aux graphes pour produire des représentations vectorielles des noeuds afin de résoudre le problème de la détection d'éléments similaires dans une base de connaissances. En s'appuyant sur des modèles pré-entraînés pour la similarité sémantique textuelle, notre méthode proposée, `PIKA`, agrège les caractéristiques hétérogènes (structurées et non structurées) d'une entité et de son voisinage pour produire un vecteur pouvant être utilisé dans différentes tâches telles que la recherche d'information ou la classification. Notre méthode apprend des poids spécifiques pour chaque type d'information apportée par une entité, ce qui nous permet de la traiter de manière inductive.

1 Introduction

Dans une base de connaissances, les éléments partageant des relations sont liés entre eux et peuvent ainsi être visualisés sous la forme d'un graphe de connaissances, très utile pour des tâches telles que l'analyse des réseaux sociaux, la prise de décision, la réponse automatique à des questions ou encore la recommandation de produits à l'utilisateur (Dai et al., 2020; Guo et al., 2020; Xiaohan, 2020). Lorsqu'une telle base de données est construite automatiquement (par l'analyse d'un flux continu de données), la tâche est nommée population automatique de bases de connaissances.

Dans une telle base, il existe un risque d'ajouter un élément déjà existant en raison de fautes de frappe dans les valeurs des attributs, de la mise à jour simultanée de la base de connaissances par plusieurs utilisateurs (humains ou IA) ou de l'ajout d'une entité avec des informations à jour sans se soucier de celle déjà présente dans la base de données. Ces cas sont très préoccupants : ils introduisent des redondances, polluent les données et ont un impact sur les performances des différentes tâches d'exploitation. Dans cet article, nous proposons une solution pour éviter ce type d'erreurs en détectant les entités similaires dans une base de connaissances avant d'insérer de nouveaux éléments.

À notre connaissance, ce problème n'a pas encore été entièrement résolu. Une solution appropriée devrait prendre en compte les différents types d'entités stockées dans une base de données (par exemple Personne, Lieu, etc...), être capable de s'adapter à un grand nombre