Rapport de projet :Détection de fraude

08/12/2021

Alexis Guillotin

Yolène Moysan

Clément Marie Brisson

# SOMMAIRE

**Introduction**

1. **Préparation du dataset**
2. **Méthodes utilisées**
3. **Analyses et résultats**

**Conclusion**

# Introduction

De nombreuses fraudes se font dans de nombreux domaines, notamment la fraude sur les bulletins de salaire. Ce projet a pour objectif de détecter des fraudes sur les montants de salaire, en utilisant de l’Intelligence Artificielle (IA) et Python.

Dans un premier temps, nous nous concentrons sur la préparation de notre dataset, ensuite, on s’intéressera aux différentes méthodes utilisées pour détecter la fraude sur les montants de salaire et pour finir, quels résultats nous avons obtenus.

# Préparation de dataset

Notre dataset est composé de bulletins de salaires et des relevés de comptes correspondant. Pour préparer notre dataset. Pour enrichir notre dataset, nous avons créé un oversampling avec les exemples qu’on avait reçus, en ajoutant des exemples modifiés (avec des cas de fraude).

# Méthodes utilisées

L’objectif de notre projet, est de pouvoir trouver une méthode qui nous permettra de détecter la fraude sur les montants de salaire, on récupère le montant net sur le bulletin de salaire et on le compare à celui du relevé de compte. Si le montant est différent, cela s’agit donc d’une fraude.

Dans premier temps, nous avons utilisé une méthode pour pouvoir détecter une fraude sur un unique cas. Cette méthode était de regarder où on pouvait récupérer le salaire net dans le bulletin de salaire. Après avoir remarqué qu’il se situait à la fin du fichier, nous avons choisi de le récupérer, en transformant nos fichiers en TXT, en reprenant la fin du texte et en supprimant « EUR » qui se situait derrière. Ensuite, nous avons fait une boucle qui permet de regarder dans tout le fichier relevé de compte, si un montant correspondait à celui du salaire net qui se situe dans le bulletin de salaire.

Une image contenant texte

Description générée automatiquement

Cette méthode était efficace, mais elle a des limites. Elle ne fonctionne uniquement sur un seul type de bulletin de salaire.

Nous avons donc testé une deuxième méthode passant directement par l’IA. Cette méthode consiste à créer des fichiers CSV, où on renseignait le contenu du texte qu’on avait récupéré, tout en précisant si c’est une fraude ou non et pour finir, des mots clés permettant pouvant correspondre dans le relevé de compte.

Une image contenant texte

Description générée automatiquement

Cette IA s’est faite en différente étapes :

Le principe fondamental de notre projet est le **Natural Language Processing** (**NLP**), qui est un domaine multidisciplinaire impliquant la linguistique, l'informatique et l'intelligence artificielle, qui vise à créer des outils de traitement de la langue naturelle pour diverses applications. Il ne doit pas être confondu avec la linguistique informatique, qui vise à comprendre les langues au moyen d'outils informatiques.

Ensuite, nous avons préparé nos données en utilisant des **Preprocessing**, tel que :

* Stopwords (ex: a, et, ou, puis etc…)
* Lemmatisation (ex: marche ; ‘marcher', ‘marché', ‘marchons’)
* Convertir le texte dans une forme standard
* Réduire la taille du texte
* Bag-of-Words
* term frequency–inverse document frequency

Nous avons aussi utilisé la méthode **Term frequency–inverse document frequency** **(tf-idf)**, qui est une méthode de pondération. Cette mesure statistique permet d'évaluer l'importance d'un terme contenu dans un document, relativement à une collection ou un corpus. Le poids augmente proportionnellement au nombre d'occurrences du mot dans le document.

Une image contenant texte

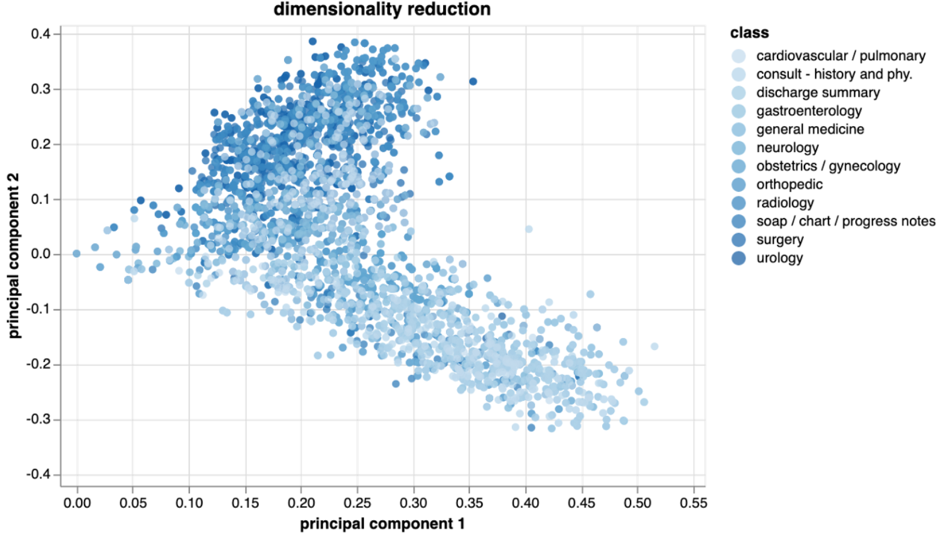
Description générée automatiquementUne image contenant texte

Description générée automatiquement



# Le résultat du calcul tf-idf est une matrice de dimensions N x D, c'est-à-dire nombre de documents multiplié par la taille du dictionnaire, remplie de valeurs tf-idf qui servent de caractéristiques.

La **réduction de la dimensionnalité** permet de ne garder que les attributs avec le poids le plus important, c'est à dire les attributs qui vont pouvoir déterminer le mieux possible si un bulletin est frauduleux ou non. En ne gardant que deux attributs on peut alors l'afficher sous la forme d'un tableau à deux dimensions. Voici un exemple d’un graphique représentant la réduction de la dimensionnalité



Pour finir, nous sommes à l’étape du **model building**. Cette étape permet de choisir le meilleur modèle, dans notre cas random forest, on entraine l’IA pour ainsi la tester.

# Analyses et résultats

Dans un premier temps, nous avions uniquement 3 jeux de bulletins de paie et de relevés de comptes. Alors, nous avons choisi de faire de l’oversampling. Et nous avons donc obtenu un F1 score de 1. Cependant, nous trouvions que l’échantillon n’était pas représentatif. C’est pourquoi nous avons décidé de modifier des bulletins de paie et des relevés de compte afin d’enrichir notre échantillon et ainsi améliorer notre oversampling.

Notre algorithme à rencontré des difficultés sur certains bulletins de salaires et certain relevés de compte suite à la mauvaise qualité de l’image. Nous avions alors pensé à améliorer la qualité des images en implémentant une IA. Cependant, cette méthode s’est avérée coûteuse en temps de développement. Nous avons donc préféré nous concentrer sur la partie de dataprep. Cependant, cette option nous aurait permit d’encore améliorer notre algorithme. En effet, avec une meilleure qualité d’image, l’algorithme aurait détecté le texte avec plus de facilité et aurait certainement eu une meilleure précision.

# Conclusion

Pour conclure, on a utilisé deux méthodes qui ont fait leurs preuves tout ayant des limites. La principale limite étant le manque de données. Des ouvertures d’approfondissement sont possibles, notamment en automatisant pour de plus grande base de données.