UNIVERSITÉ PARIS 8
VINCENNES-SAINT-DENIS

M1 MIASHS : Big Data et Fouille de données

Interface d'automatisation du processus de recrutement

Organisme d'accueil
AMA Associates

Auteur
GHOUIBI Ghassen

Encadreur - Organisme d'accueil
Aurélien MICHEL

Encadreur - Université
Jean-Jacques Mariage

Résumé

La révolution digitale que nous vivons actuellement attire un nombre grand d'utilisateurs, qui se traduit un besoin important de spécialiste dans le domaine de l'informatique.
Ces dernières années l'apparition des ESN [1] prend de plus d'en plus d'ampleur pour chercher le candidat idéal qui à permis la naissance à plusieurs plateforme comme LinkedIn en 2002. Le processus de rectrutement devient de plus en plus lourd surtout quand il s'agît de séléctionner un candidat parmis plusieurs de même les rectruteurs ne peuvent pas faire un tri dans des métadonnées.
Le développement d'une interface qui automatise ce processus présente un véritable défi. En effet les recruteurs passent en moyenne 40 secondes pour lire un CV et 1 minute, 20 secondes pour décider si le candidat sera retenu dans la sélection.
Notre but est créer une interface qui pourrai nous présenter les meilleurs candidat par rapport à une fiche de poste. Les NLP présentent une solution pour résoudre ce problème, ces solutions donnent des résultats plutôt trés correcte, en revanche les modèles qui existent actuellement trouve une difficulté à détecter des compétences générales par rapport à des compétences techniques.
En mettant l'accent sur un modèle qui pourrai à la fois détecter les compétences générales intéressantes qui pourrait contribuer à la monté en compétence de candidat, aussi que donnée l'accès à des formations seraît un plus pour faire des économies au niveau de l'embauche aussi bien que donnée une chance équitable à chaque candidat.
Dans ce papier, notre but sera de créer une architecture qui pourrait prendre un nombre massive des données qui sera représenter par un banque de CV. Ensuite extraires les informations nécéssaire à partir d'une fiche de poste nous allons essayer de coincider cette dernière avec une sélection des meilleurs CV notamment un modèle basée sur les réseaux de neuronnes ainsi utiliser Word2vec, Text2Vec pour atteindre notre objectif.

Mots-clefs : Word2vec, LSTM, Text2Vec, Naive Bayes, TF-IDF, NLP

1. Une entreprise de services du numérique

Problèmatique

Les recrutements dans une entreprise sont un processus trés important pour le développement d'une société aussi bien qu'il permet de trouver la bonne formule et la bonne équipe à travailler ensemble.
En effet, le choix d'un candidats se fait sur plusieurs critères mais devient long dans certain grande entreprise vu le nombre de candidats, séléctionner à partir d'un centaine de profils devient coûteux, comment economiser le temps de fouille pour un profil ?

Est-ce que un système de recommendation de candidats sera efficace sans vraiment oublié le côté humain dans le processus du recrutements ?
Les CV reçu sur un espace de recrutements sont nombreux ils peuvent se présenter sous plusieurs format (exemple : Docx,Doc,Pdf ...) et certain encodage (exemple : utf-8,utf-16 ...), comment procèder pour regouper ces fichiers sous un seul format ?
L'extraction d'information à partir d'un CV est une étape trés importante avant de procèder à mettre les CV en compétition, le défi sera de comprendre ces derniers est pouvoir les regrouper en sorte cluster.
Est ce que les systèmes de recommendation actuelles sont efficaces pour prendre un décision à la place de l'humain ?
L'interface proposer dans se papier pourrait faire des économies en terme de charge salariale et gagner du temps sur d'autres opérations comme signature des contrats ... etc
Dernièrement, la problèmatique pertiente est-ce que l'algorithme sera-t-il efficace pour remplacer les choix humains sans aide extérieur constante ?, comment peut on améliorer les résultats obtenu et alimenter notre base de données ?

Table des matières

Chapitre 1

Introduction

La dernière décennie à vu l'émergence d'internet, des nouveaux emplois ont vu le jour grâce à l'automatisation de plusieurs processus. Pour çelà plusieurs entreprise essayent de trouver les bons candidats à leur entreprise mais la mission devient trés difficile quand on parle des `ESN`.
L'apparition de plusieurs plateforme qui traitent notre curriculum vitæ pour prédire le meilleur candidat pour un poste comme LinkedIn. Notre but c'est concevoir une interface qui permet de correspondre une fiche de poste à notre base de données de candidats.

Les avancées de recherche dans le domaine du `text-mining` ne peuvent que présenter une solution idéale pour ce genre de problème néanmoins, il faudrait d'abord comprendre le processus de recrutement et la selection d'un candidats par rapport à un autre. En effet la touche humaine ne peut pas être négliger pour choisir un candidats, plusieurs plateforme échoue dans plusieurs essai pour plusieurs raisons comme le format ou l'encodage.
Malgré plusieurs avancer mais jusqu'a l'heure actuelle aucune plateforme n'a réussi à détecter les compétences générales qui veut dire un candidat intéressant mais qui aura besoin d'une petite formation çelà présente beaucoup d'avantage pour l'entreprise au point de vu économique.
La conception d'une interface qui permet à la fois de trouver un curriculum vitæ idéale en prennant en compte tous les aspects techniques mais aussi les aspects humains et donner une chance à tous les candidats d'une manière équitable. Vu le grand nombre de candidatures sur un poste dans une grande entreprise il sera impérative d'utiliser un algorithme qui permet de filtrer tous les documents reçu et les classées.
Notre dataset qu'on va utiliser dans ce papier serait fournit par `Zoho` [1], en premier lieu on va travailler sur ce dataset qui semble complet avec des fiches de postes et des curriculum vitæ néanmoins ce logiciel présente la fonction-

1. Zoho Office Suite est une suite bureautique en ligne sur le Web contenant du traitement de texte, des feuilles de calcul, des présentations, des bases de données, la prise de notes, des wikis, des conférences Web, la gestion de la relation client, la gestion de projet, la facturation et d'autres applications.

nalités de correspondre une fiche de poste avec un résume sauf que le but de l'entreprise est d'abondonner ce logiciel voir le remplacer au fils des années par un produit fait maison, aussi bien que dans le cas notre dataset ne sera pas suffissant opter pour le scrapping sur LinkedIn sera une des solutions.

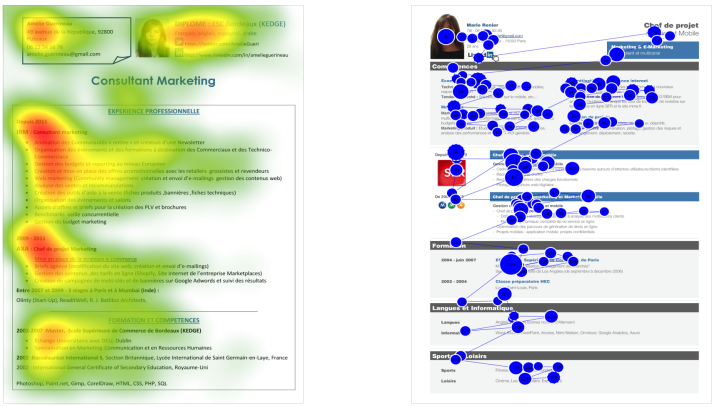

FIGURE 1.1 – Une lecture structurée d'un curriculum vitæ (lecture en F)

Dans la figure 1.1, nous remarquons la fameuse lecture en F et çelà nous permet en premier lieu de comprendre la structure d'un curriculum vitæ.
D'où nous pouvons déduire que la lecture d'un résume se base plutôt sur les mots clés, la plupart du temps çelà mèner à lire les postes occupées au par avant sans plonger dans les détails,la formation, et les informations du candidat.
En effet, c'est tout à fait normal que les rectruteurs vont opter pour une lecture en F vu qu'ils voient des centaines de curriculum vitæ et ça devient automatique de comprendre rapidement un résume juste en identifiant les mots clés.
Pour pouvoir identifier les compétences, les langues, les expèriences... etc nous avons besoin tout d'abord de pouvoir trouver des mots clés sachant que le nom et prénom sont aussi des mots clés, Résumons le processus que nous allons adataper pour pouvoir comprendre et analyser un curriculum vitæ.
Notre approche vise plutôt les documents *PDF* il faudrait tout d'abord pouvoir analyser tout type de documents qui nous donnera encore plus de donner pour tester notre modèle comme *Doc*, *Docx*, *HTML* ..etc
Ensuite classifier le texte à partir de l'indentification des mots clés dans ce dernier et produire des données structurés.
Du même principe, nous analyserons la fiche de poste notre algorithme va essayer de faire correspondre les mots clés présent dans les deux documents.
Comme précisé au par avant plusieurs recherches ont était effectuées sur ce sujet, bien que plusieurs interface voit le jour, un article proposer par Patrice Darmon,Rabah Mazouzi, Otman Manad et Mehdi Bentounsi intitulé

TeamBuilder : D'un moteur de recommandation de CV notés et ordonnés à l'analyse sémantique du patrimoine informationnel d'une société.

Le processus de séléction consiste à attribuer des scores pour toutes les compétences extraites d'un curriculum vitæ. Deux types de scores, les scores statiques sont calculés directement à partir du CV, les scores radars qui sagit plutôt d'une agrégation des plusieurs scores statiques.[1].
Comme illustre la figure ci-dessous, notre interface doit pouvoir classifier par ordre d'importance les curriculum vitæ le mieux adapter a notre fiche de poste. Nous remarquons aussi l'orientation du profil comme Data Science, Développement, Infrastructure.

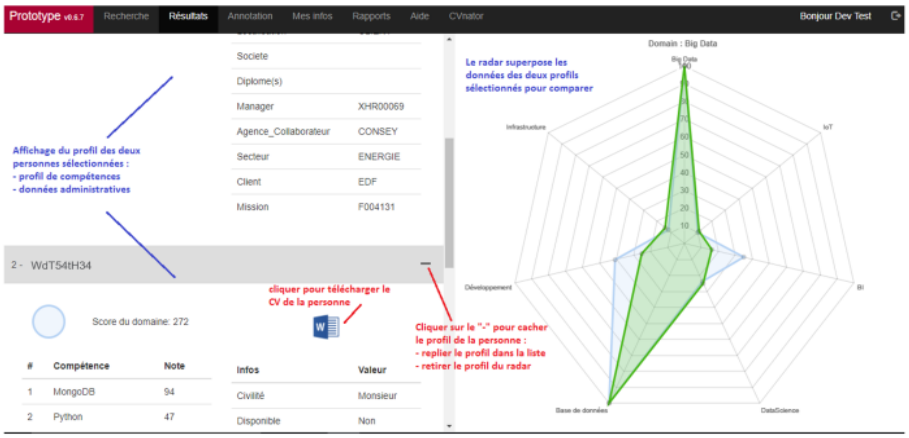

FIGURE 1.2 – Liste des CV scorés et ordonnées

Le papier sera découper de la manière suivante, Dans la deuxième section, nous allons aborder les différents travaux réaliser au tour de ce sujet. Ensuite la troisème section nous allons détaillé l'architecture utilisée. Puis la quatrième nous allons décrire notre modèle, Pour Finir avec une dernière section qui sera réservé pour la conclusion suite a notre résultat.

Chapitre 2

État de l'art

Les avancées sur les processus ne cessent que d'améliorer et çelà est liée au recherches réaliser dans le domaine de l'intelligence artificiel. Nous remarquons plusieurs entreprises emplois massivement des interfaces qui permettent d'extraire des données à partir d'un curriculum vitæ.
Ce sujet est d'actualités dans plusieurs communauté comme dans l'article suivant *A Two-Step Resume Information Extraction Algorithm[2].*
La classification textuelle est une étape trés importante, nous remarquons trois types de texte *simple* qui représente un texte court, *KeyValue* représenter par une clé valeur exemple Data Scientist : Safran généralement séparer par un signe de ponctuation, *Complexe* contient plusieurs ponctuations avec un long texte. Aprés l'extraction de données, la similarité sera calculé sur la base de `TFIDF` et `Kmeans` montrera le cluster d'attributs ensuite ces clusters seront mis en correspondance avec l'attribut du profil.
Un curriculum vitæ est composé d'un ensemble de compétences techniques et transversales, Les ontologies par exemple : Compétences $\in \{BigData, Web, IOT...\}$, il sera donc indispensable de mettre en place des filtres sur plusieurs paramètres [1].

`TF-IDF`, est une méthode trés utiliser dans la fouille texte, cette dernière permet d'évaluer l'importance d'un terme contenu dans un document. Le papier suivant[2] traite la problèmatique de notre sujet en utilisant cette méthode, il présente plutôt des résultats trés satisfisante avec une présicion de 0.94 pour l'extraction de la formation, 0.914, 0.831 simultanément pour le nom de la société et la profession.
En effet, il existe plusieurs méthodes pour correspondre du texte : l'article suivant [3] utilise `MV-LSTM` pour combiner la réprésentation basée sur la position existante avec une couche basée sur l'attention, l'objectif est de savoir plus sur les mots les plus importants et les plus informatifs en rendant plus efficace la correspondance avec d'autres texte.

Les systèmes de recommandation existent en trois modes, *démarrage à chaud* recommandation et illustré par le défi Netflix[9], *démarrage à froid* autrement dit les nouveaux candidats sont à la recherche de nouveaux postes, Finalement le mode *démrrage semi-à froid* les recommandations exploitent les interactions pasées[8]. Plusieurs entreprises investissent dans des systèmes de recommandation de CV, suite au nombre de candidature reçu, comme Société Générale, Allianz ... etc mais les résultats de leur recherche ou plutôt les astuces utilisées reste toujours un mystère vu la concurrence présente sur ce sujet.

Word2vec est une technique de traitement du langage naturel. L'algorithme word2vec utilise un modèle de réseau neuronal pour apprendre les associations de mots à partir d'un grand corpus de texte. Une fois formé, un tel modèle peut détecter des mots synonymes ou suggérer des mots supplémentaires pour une phrase partielle.
Dans cet article[10], le modèle détaille que la mise à niveau des vecteurs de Word2vec améliore la sortie des vecteurs de 13,9%, contre -7,3% indiquent que l'approche proposé sans mise à niveau est meilleure que les vecteurs de Word2vec modernisés. Avec des tests réaliser sous un processeur `E3-1246 v3` le modèle `Dict2Vec` se montre rapide sur 50M de fichier par rapport à `Word2Vec` 15m30 contre 4m09 pour `Dict2Vec`.

Le but est réussir à évaluer la similitude entre les mots plus en particulier entre une fiche de poste et un CV, en effet trouver une évaluation de la similitude humaine des paires de mots et de la similitude cosinus du mot correspondant aux vecteurs nous suivons le même protocole utilisé par Word2vec et fastText en supprimant les paires inutiles.

Bibliographie

[1] Patrice Darmon, Otman Manad, Rabah Mazouzi et Mehdi Bentounsi : *TeamBuilder : D'un moteur de recommandation de CV notés et ordonnés à l'analyse sémantique du patrimoine informationnel d'une société, Oct 2018.*

[2] Jie Chen, Chunxia Zhang, et Zhendong Niu : *A Two-Step Resume Information Extraction Algorithm, Fév 2018.*

[3] Thiziri Belkacem, Taoufiq Dkaki, José G. Moreno, Mohand Boughanem : *aMV-LSTM : an attention-based model with multiple positional text matching, Jan 2019.*

[4] Gilles Didier : *Optimal pattern matching algorithms, Mai 2016.*

[5] Houda Amazal, Mohammed Ramdani, Mohamed Kissi : *Distributed multi-Label classification approach for textual Big Data, Sep 2019.*

[6] François Gonard : *Cold-start recommendation : from Algorithm Portfolios to Job Applicant Matching, Juin 2018.*

[7] Manon Ansart, Stéphane Epelbaum, Geoffroy Gagliardi, Olivier Colliot, Didier Dormont, Bruno Dubois, Harald Hampel, Stanley Durrleman : *Reduction of recruitment costs in preclinical AD trials. Validation of automatic pre-screening algorithm for brain amyloidosis, Fév 2019.*

[8] Thomas Schmitt, François Gonard, Philippe Caillou, Michèle Sebag : *Language Modelling for Collaborative Filtering : Application to Job Applicant Matching, Déc 2017.*

[9] R. M. Bell and Y. Koren : *Lessons from the Netflix prize challenge, ACM Sigkdd Explorations Newsletter, vol. 9, no. 2, pp. 75–79, 2007..*

[10] Julien Tissier, Christophe Gravier, Amaury Habrard *Dict2vec : Learning Word Embeddings using Lexical Dictionaries 12 Oct 2017.*