Analyse des offres d'emploi en ligne : comment coder le métier ?

Paul Andrey, Ensae
Maxime Bergeat, Dares

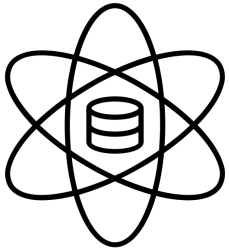
contexte

- Internet: important canal de diffusion d‘offres d‘emploi
- Refonte de la publication « Tensions sur le marché du travail »
- ESSnet Big Data – Workpackage 1: Scraping Job Vacancies

chaîne de traîtement

Created by Gregor Cresnar
from Noun Project

collecte de données
d‘offres d‘emploi



Created by Nirbhay
from Noun Project

structuration
de l‘information

Created by Yamini Ahluwalia
from Noun Project

production
de statistiques

exemple: comment identifier
le métier ?

Sources de données et pré-traitements

sources de données

Offres partenaires de Pôle emploi

- Panel de sites (> 100)
- Source relativement stable

- Offres partenaires de 2016
  ~ 4.3 millions d‘offres
- Traitées par Pôle emploi

Offres scrapées par la Dares

- Applicable à (presque) tout site
- Coût technique et temporel

- Offres d‘un site à une date
  ~ 60 000 offres
- Données brutes

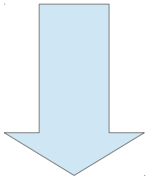
nettoyage des données textuelles

Notre belle entreprise, « Super café » recrute !

Vous aurez de nombreuses tâches:

-Vous préparerez et servirez du café.

-Vous (et vos collègues) sourirez aux clients.Et plus !



beau entreprise super café recrute avoir nombreux tâche préparer servir café collègue sourire client plus

nettoyage des données textuelles

- Normalisation
  - gestion des caractères spéciaux — *é* → *é*
  - uniformisation ou retrait de la ponctuation — *café
-Vous* → *café. Vous*

- Lemmatisation
  - (identification des fonctions grammaticales) — *préparerez* → *[verbe]*
  - remplacement des mots par leur lemme → *préparer*

- Filtrage
  - retrait des « mots vides » (*stopwords*) — *préparer et servir* → *préparer servir*
  - retrait des caractères non désirés — *café. (collègues)* → *café collègues*

outils de lemmatisation

Morphalou

- Lexique de formes fléchies
- Lemmatisation mot par mot

→ utilisé sur les titres

Tree Tagger

- Part-of-Speech tagging
- Lemmatisation texte par texte

→ utilisé sur les descriptifs

Codification du métier par appariement

procédure d'appariement

libellé d'offre

serveur salon thé moderne

identification des termes apparaissant dans les appellations de référence du ROME (nettoyées)

serveur / salon / thé

sélection des appellations du référentiel contenant au moins l'un de ces termes

Appellation métier	Code ROME
serveur salon thé	G1801
...	...
serveur restaurant	G1803

calcul de la similarité entre le titre et chaque appellation

Similarité
.875
...
.375

meilleur score

ROME	Score	Appellation métier
G1801	.875	serveur salon thé

fonction de similarité

$$simil(t_1, t_2) = \frac{\sum_{w \in t_1} f_2(w)}{Card(t_1)} \times .5 + \frac{\sum_{w \in t_2} f_1(w)}{Card(t_2)} \times .5$$

$$f_k(w) = \exists w' \in t_k, jaro\_winkler(w, w') \geq .9$$

Distance d'édition de Jaro-Winkler:

- part des lettres identiques à une position proche au sein des deux mots
- nombre de transpositions pour aligner les lettres identiques
- valorisation des mots partageant les mêmes premières lettres

Apprentissage automatique de la codification du métier

vectorisation des descriptifs

$Corpus = \{t_1, \ldots, t_N\}$ $\quad \forall t \in Corpus,\ t = \{m_{t,1}, \ldots, m_{t,Card(t)}\}$

$Voc = \{m_1, \ldots, m_K\}$: ensemble de mots (lemmes) considérés, fixé *ex ante*

:= ensemble des mots apparaissant dans au moins 2% des descriptifs associés à l'une des Fap (sur 22)

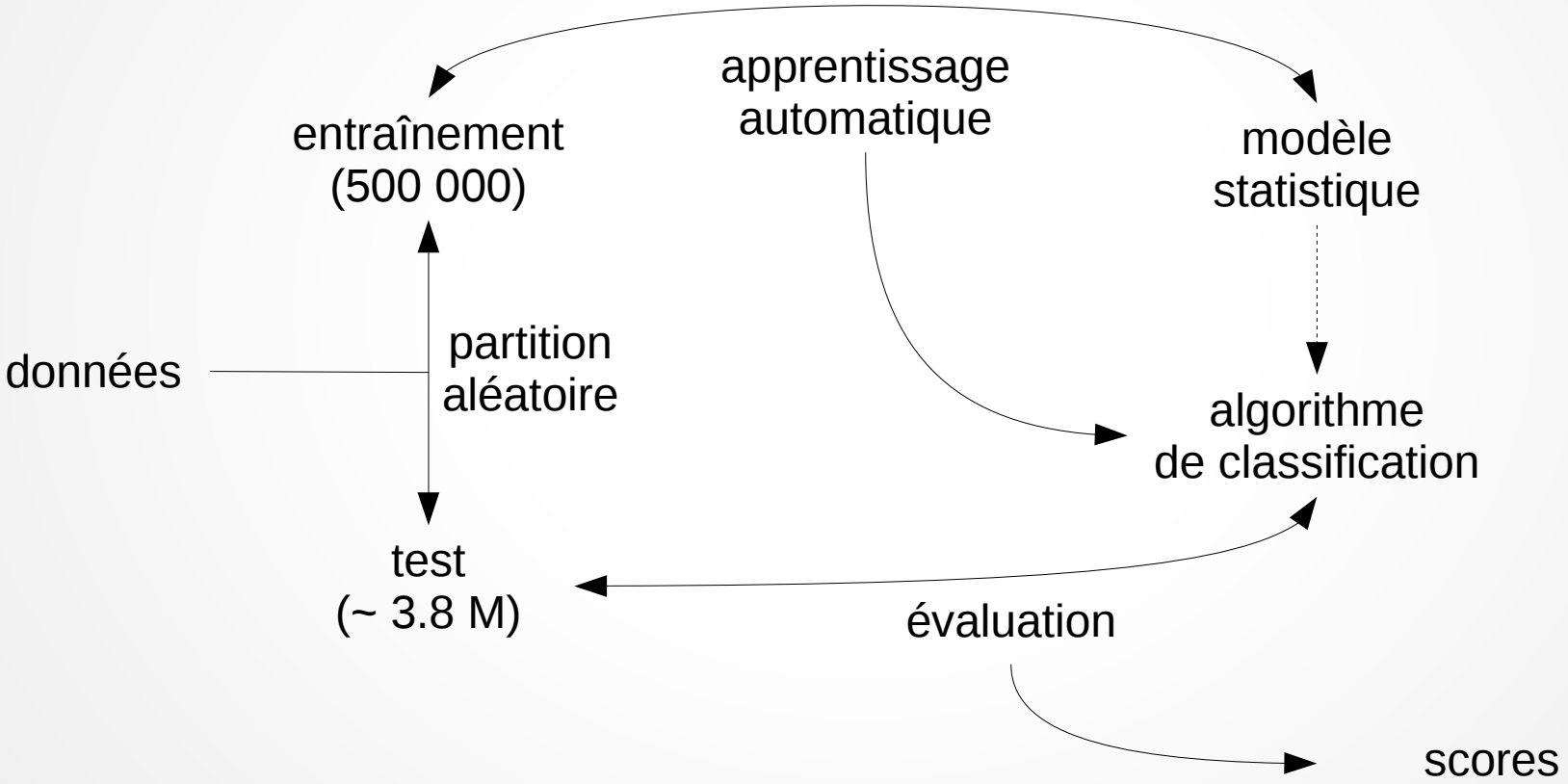
$\forall t \in Corpus, v_t = (f_t(m_1), \ldots, f_t(m_K))$

f_t : fonction quantifiant la présence d'un terme dans le texte t

:= tf-idf (*term frequency – inverse document frequency*)

$$dtm = \begin{bmatrix} v_1 \\ \ldots \\ v_N \end{bmatrix} \in \mathcal{M}_{N,K}(\mathbb{R})$$ (matrice documents-termes)

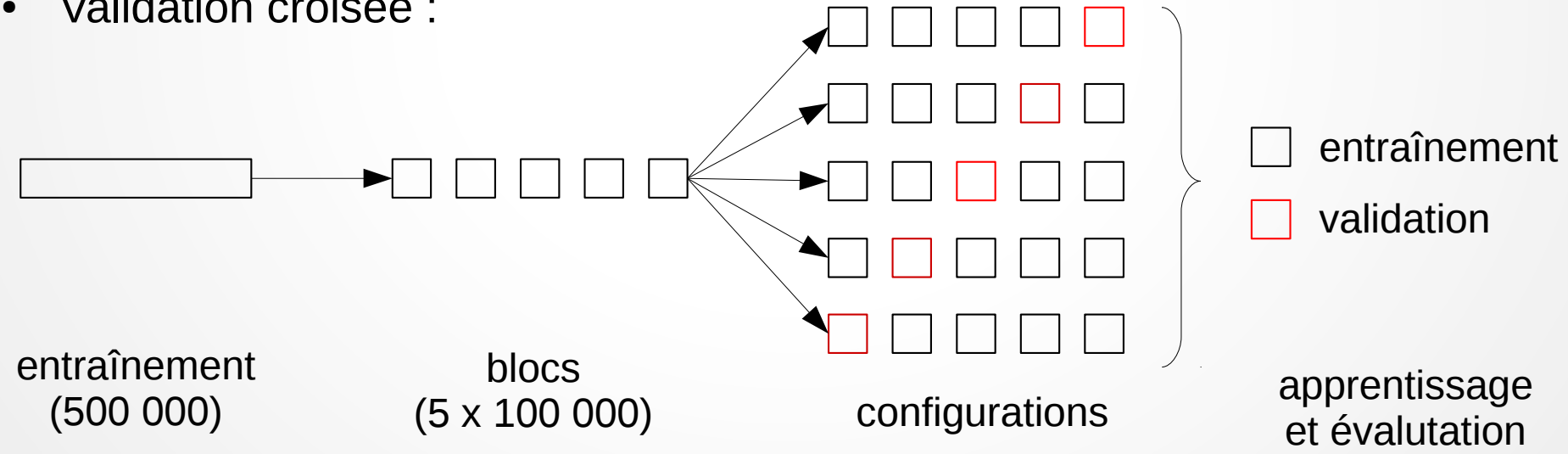
apprentissage supervisé



affinage des modèles

- *Grid search* : recherche du meilleur modèle parmi l'ensemble des modèles définis par une grille de valeurs des (hyper)paramètres testés

- Validation croisée :



modèles utilisés

- régression logistique (modèle linéaire généralisé)
- forêt aléatoire (apprentissage ensembliste)
- perceptron multicouche (réseau de neurones artificiels)

métriques d'évaluation

- Exactitude : part des prédictions exactes (par modalité ou globalement)
- Précision : $P = \frac{VP}{VP + FP}$ (part des cas corrects pour une modalité prédite)
- Rappel : $R = \frac{VP}{VP + FN}$ (part des cas corrects pour une vraie modalité)
- Score F1 : $F1 = \frac{2 \times P \times R}{P + R}$ (moyenne haromnique de la précision et du rappel)

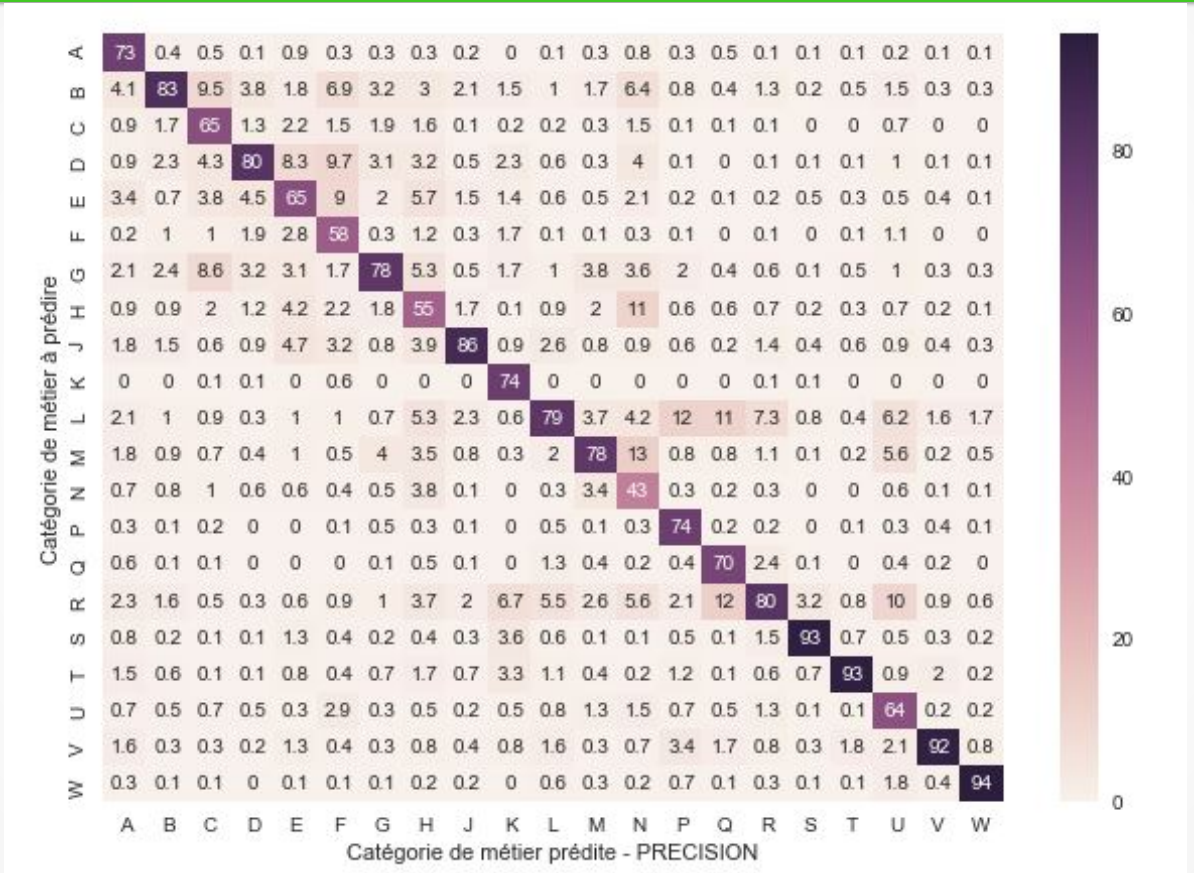
aggrégation : moyenne pondérée par la fréquence de chaque modalité

résultats

Modèle	Exactitude	Précision glob.	Rappel glob.	Score F1 glob.
Régression logistique	78.9 %	78.6 %	79.9 %	79.3 %
Forêt aléatoire	80.1 %	80.3 %	80.1 %	80.2 %
Perceptron multicouche	80.6 %	80.5 %	81.2 %	80.8 %

Note : résultats comparables au Portugal, où trois modèles sont utilisées pour codifier le métier dans la nomenclature ISCO (41 catégories) : SVM (précision globale de 78 %), Régression logistique (77 %), Forêt aléatoire (67 %)

résultats



exemple de matrice de confusion
(précision du perceptron)

pistes d’amélioration

- définition du vocabulaire
- inclusion de variables additionnelles
- choix des données d‘entraînement
- granularité de la variable cible
- spécification des modèles

Merci pour votre attention !