Fair Learning et NLP : Modèles de Biais d'interprétation

Hugues Van Assel

Décembre 2019

Résumé

L'article "Bias in Bios : A Case Study of Semantic Representation Bias in a High-Stakes Setting" (De-Arteaga et al.) donne une perspective intéressante sur les dynamiques d'accroissements des inégalités dans le domaine du traitement du langage naturel. Dans ce rapport, nous nous concentrons sur : éclairer les méthodes utilisées mais non explicitées par cet article, et reproduire les algorithmes qui y sont présentés à partir de données disponibles. Nous avons ainsi retrouvé la relation de proportionnalité entre la proportion de femmes dans une profession et le "TPR gender gap" de cette profession. Cette corrélation rend l'utilisation d'algorithmes d'apprentissage problématique puisqu'elle tend à creuser les inégalités.

1 Introduction

Le NLP est un sujet actuellement complexe : même les algorithmes les plus efficaces aujourd'hui peinent à comprendre en profondeur le langage, par exemple le « Question Answering » est un problème sur lequel les algorithmes actuels ont des résultats très médiocres.
Un des problèmes actuels des algorithmes d'IA est qu'ils apprennent à partir de critères qui n'ont pas de relation de cause à effet avec notre problème. Un exemple concernant ce problème est mis en évidence par Tulio Ribeiro en 2016 avec un algorithme de Deep Learning classifiant les images de loups et de huskys. Après examen de l'algorithme, on se rend compte que l'algorithme base son choix uniquement sur la couleur du fond : si le fond est blanc (enneigé), l'algorithme va classifier une image de loup, et sinon une image de husky. Ainsi, selon la même logique, certains algorithmes, par exemple de recrutement, peuvent être amenés à discriminer les femmes en considérant les inégalités actuelles comme ayant une explication rationnelle alors qu'elles sont le fruit de discriminations historiques. Bien sûr, pour résoudre ce problème, il ne suffit pas de supprimer la variable de genre des données, puisque le genre est corrélé à d'autres variables. Nous allons donc voir dans ce rapport comment quantifier ces inégalités causées par des biais d'interprétation et quels modèles nous permettent de limiter au maximum ce genre d'inégalités.

En plus de la base de données Common Crawl, nous disposons dans ce projet d'un jeu de données de 300 000 biographies. Malheureusement, la complexité des algorithmes nous a contraint à nous restreindre à 30 000 biographies afin d'obtenir nos résultats et conclusions dans le temps imparti. Notre dataset nous semble tout de même d'un fort intérêt puisque qu'il est globalement équilibré en terme de proportion homme/femme (Fig 1), il représente une quantité importante de professions différentes (Fig 2) et les sexes sont représentés inégalement au sein de chaque profession (Fig 3).

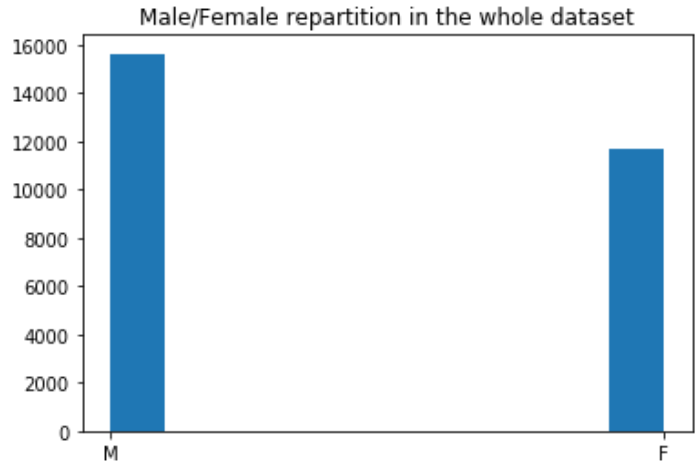

FIGURE 1 – Répartition homme/femme des biographies

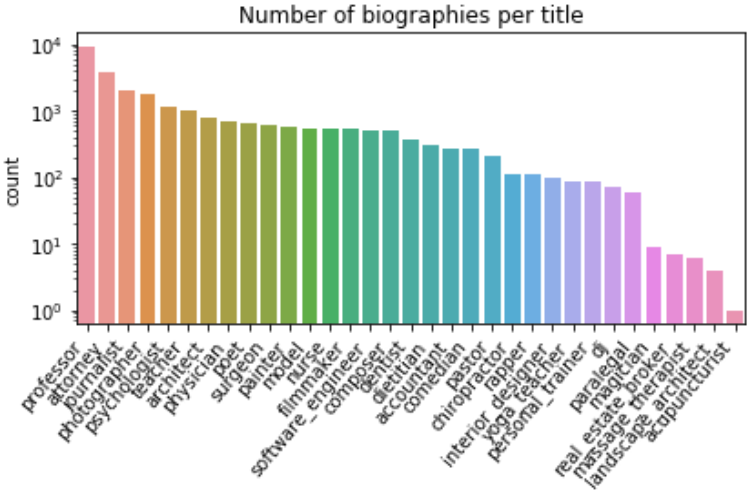

FIGURE 2 – Répartition des biographies par profession

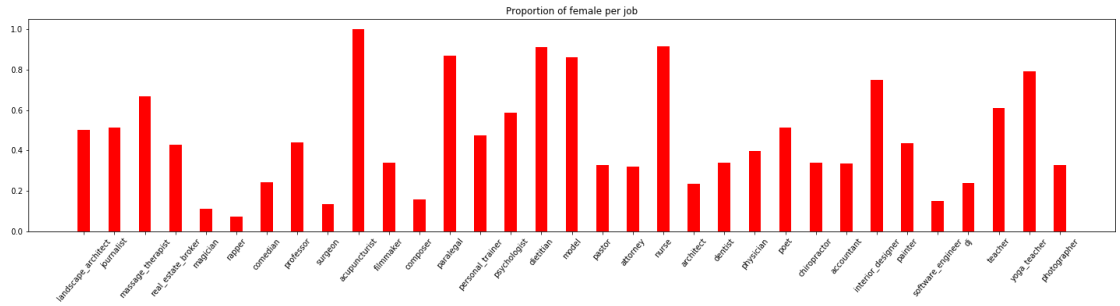

FIGURE 3 – Proportion de femmes pour chaque profession

Notre code est organisé de la manière suivante : traitement des données et visualisations dans Dataprocessing.ipynb, modèles de prédiction du genre dans Genderprediction.ipynb, modèles de prédiction du titre dans Titleprediction.ipynb et le réseau de neurones récurrents dans RNN.ipynb.

2 Outils mathématiques

2.1 Les différentes représentations d'un texte

Le Bag-of-words (BOW) est une méthode de repésentation d'un texte par un vecteur de taille le nombre de mots différents qu'il existe dans l'ensemble des textes que l'on considère. Chaque élément de ce vecteur va représenter un mot, et la valeur de cet élément sera alors 1 si le mot associé est contenu dans le texte considéré, et 0 sinon.

Le word embedding (WE, ou "prolongement lexical" en français) est une autre méthode de représentation d'un texte. La première étape consiste à utiliser un ensemble de données que l'on ne va pas utiliser dans notre problème, par exemple le Common Crawl. On va réaliser des n-grammes de mots à partir de tous les textes de cet ensemble. Un n-gramme est un doublet de mots qui sont situés à une distance plus faible que n. Par exemple, pour la phrase "the quick brown fox jumps over the lazy dog", on obtient les 2-grammes suivant :

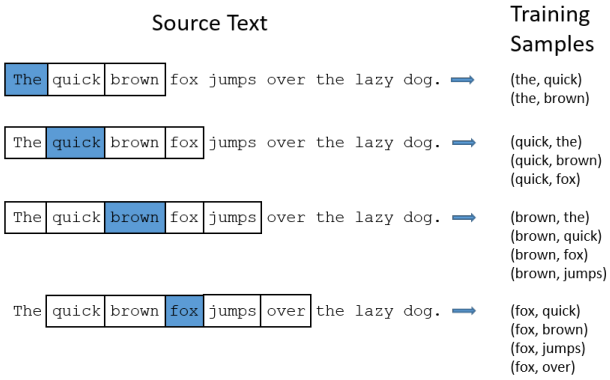

FIGURE 4 – Exemple de n-grammes

Une fois que l'on a l'ensemble des n-grammes de nos textes d'entraînement, on entraîne un réseau de neurones assez simple :

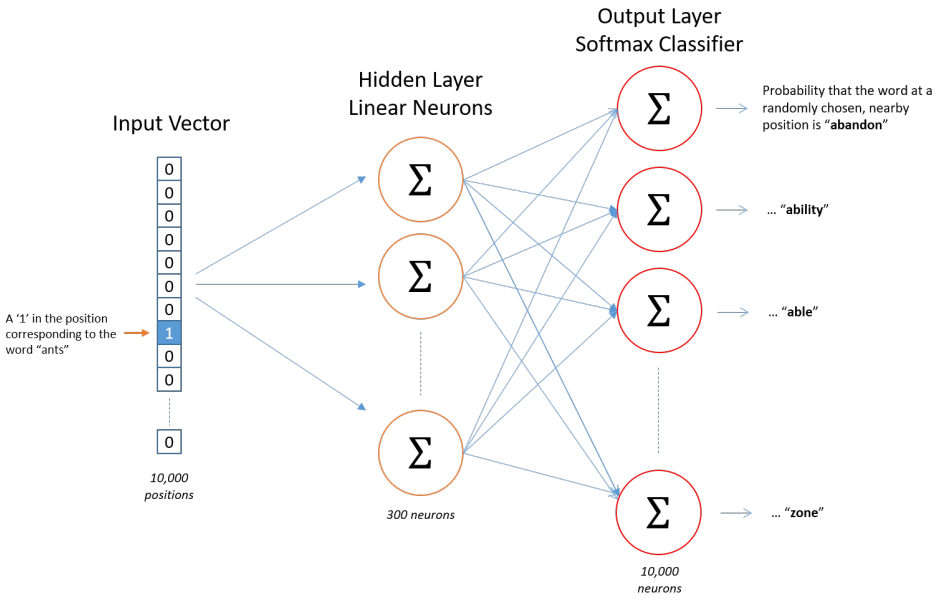

FIGURE 5 – Réseau de neurones à entraîner

Ce réseau de neurones dispose d'une entrée de la forme d'un vecteur de la taille de l'ensemble des mots présents dans le corpus, d'une couche de neurones linéaires de nombre a (où a est un hyperparamètre que l'on pourra optimiser par la suite) et d'une sortie sous la forme d'un vecteur de la même taille que l'entrée. On entraîne alors ce réseau pour qu'il nous prédise l'ensemble des mots susceptibles d'être proches d'un mot donné.
La couche cachée peut donc représentée par une matrice de nombre de colonnes a et de nombre de lignes le nombre de mots considérés. On représente donc chaque mot par le vecteur qui constitue la ligne associée de la couche cachée. Après optimisation de l'hyperparamètre a, on a donc une représentation de chaque mot par un vecteur de taille 300. Le but de cet entraînement était donc seulement d'entraîner la couche cachée pour pouvoir l'exploiter.
Ceci effectué, pour un nouveau texte donné, sa représentation WE est alors la moyenne de la WE de chacun de ses mots.
Toutefois, on se rend très vite compte que ce réseau est énorme et que l'entraîner sur tous nos échantillons va prendre une quantité de temps conséquente. De plus, certains mots vont apparaître de nombreuses fois sans que les nouvelles occurences nous éclairent sur le sens du mot, par exemple "le" qui est présent dans une forte proportion des échantillons considérés. On peut alors appliquer la méthode du subsampling (ou sous-échantillonage en français) qui consiste à associer à chaque mot une probabilité de ne pas l'ajouter à l'ensemble d'entraînement lorsqu'on le rencontre à nouveau, probabilité qui augmente en fonction de sa fréquence dans les textes déjà parcourus.
Ce WE peut être directement utilisé à partir de la bibliothèque fasttext.

Dans ce travail, nous avons utilisé des algorithmes d'apprentissage prenant en entrée des biographies représentées soit par un vecteur Bag of Words soit par la moyenne des Word Embeddings des mots qui la composaient.

Nous allons également utiliser des réseaux de neurones récurrent et plus précisément des réseaux

récurrents bidirectionels, avec des GRU (Gated Recurrent Units). Un réseau récurrent est un réseau de neurones artificiels possédant des connexions récurrentes, c'est-à-dire que celui-ci est formé de neurones interconnectés interagissant non-linéairement et pour lequel il existe au moins un cycle dans la structure. Bidirectionel signifie que les neurones peuvent être influencés par à la fois les états précédents et suivants. Enfin, les GRU sont simplement le mécanisme de déclenchement du réseau, au même titre que LSTM l'est dans d'autres architectures par exemple.

Plus formellement, le réseau utilisé dans l'article prend en entrée la suite des représentations fasttext WE du texte i : $e_i^1, \ldots, e_i^T$. On pose alors :

$$\overrightarrow{h_i^t} = \overrightarrow{GRU}(e_i^t, h_i^{t-1})$$

$$\overleftarrow{h_i^t} = \overleftarrow{GRU}(e_i^t, h_i^{t+1})$$

$$h_i^t = [\overleftarrow{h_i^t}, \overrightarrow{h_i^t}]$$

Ceci effectué, on pose, avec les poids w_a, W_a et b_a :

$$u_i^t = {w_a}^t \tanh(W_a h_i^t + b_a)$$

$$\alpha_i^t = \frac{\exp(u_i^t)}{\sum_{k=1}^{T} \exp(u_i^k)}$$

Ce qui donne la sortie :

$$x_i = \sum_{k=1}^{T} \alpha_i^t h_i^t$$

A partir de laquelle on effectue une prédiction de type :

$$\hat{y}_i = softmax(W_0 x_i + b_0)$$

Les deep recurrent neural networks sont très utiles pour notre problème puisque le langage est un ensemble : on ne peut pas comprendre le sens d'une phrase mot par mot, c'est bien l'agencement de ceux-ci qui nous permet de saisir la signification de la phrase, ainsi on ne peut pas "oublier" les mots, d'où nécessité d'une longue mémoire à court terme, et donc de l'utilisation de ce type de réseaux de neurones. Le caractère bidirectionel nous permet d'augmenter la quantité d'information disponible au sein du réseau. De plus, ce réseau dispose d'un mécanisme d'attention, qui nous permettra a posteriori de déterminer quels éléments de l'entrée ont le plus influencé le choix du métier donné en sortie par le réseau (et donc, quels mots sont responsables du choix). Dans notre cas, nous avons représenté les mots en entrée sous forme de word embedding.

2.2 Quantification des inégalités

Tout d'abord, dans les cas binaires où une situation est préférable à l'autre (comme le recutement), il est évident que l'on peut quantifier les inégalités par ce que les Américains appellent le Disparate Impact :

$$DI = \frac{P(S=1|G=g)}{P(S=1|G=\sim g)}$$

La loi aux Etats-Unis impose $DI > 0.8$, toutefois, cela n'est pas suffisant pour assurer l'égalité des chances. Par exemple, toujours aux Etats-Unis, l'entreprise Northpointe a élaboré un algorithme qui produisait un score représentant le risque de récidive des prisonniers, dans le but de décider si l'on devait accorder au détenu une liberté conditionnelle. Le DI entre Blancs et Afro-Américains était raisonnable, mais il cachait en réalité un autre problème : même s'il n'y avait pas de discrimination collective envers les Afro-Américains, il y avait une discrimination individuelle : les résultats étaient bien plus fiables pour les Blancs que pour les Afro-Américains, ce qui signifie, en caricaturant, que l'algorithme fonctionnait efficacement pour les Blancs, mais jouait à pile ou face pour décider si un Afro-Américain restait dangereux ou non.
Ce problème nous incite donc à introduire ce qu'on appelle le "True positive rate (TPR) gap", par exemple si on revient à notre problème de genre :

$$Gap_{g,y} = TPR_{g,y} - TPR_{\sim g,y}$$

où $TPR_{g,y}$ représente le taux de vrais positifs pour le genre g et la profession y. Le TPR gender gap mesure ainsi la différence d'efficacité de l'algorithme de prédiction en fonction du genre, et donc la discrimination individuelle en fonction du genre.

3 Résultats

3.1 Prédiction du sexe et effacement partiel du genre

Nous avons implémenté les méthodes de prédiction du sexe à partir des représentations BOW et WE par régression logistique. Pour BOW, on obtient 99% de réussite, et pour WE, on obtient 85%, ce qui est logique, puisque WE effectue une moyenne sur tous les mots, et donc dilue l'information. On peut alors afficher pour BOW les mots qui ont le poids le plus fort dans la décision de l'algorithme :

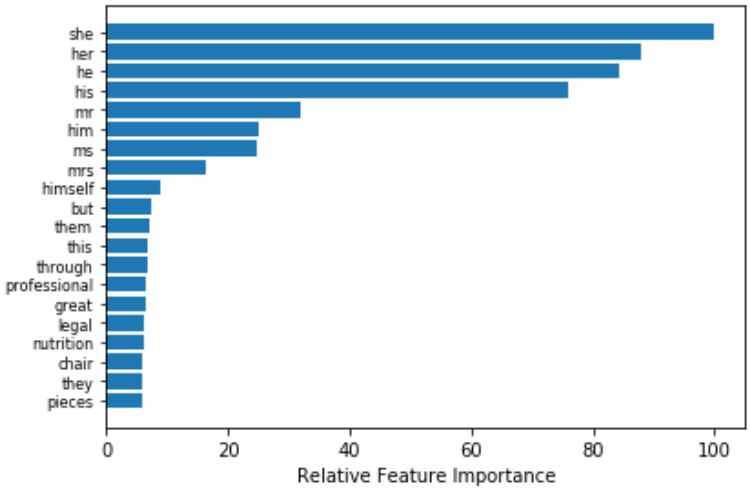

FIGURE 6 – Mots les plus décisifs

Notre résultat semble cohérent puisque l'algorithme base ses choix sur des mots tels que "she", "him" ou "mr" qui sont effectivement des indicateurs du genre. On peut alors enlever ces mots pour tenter de supprimer les informations de genre des biographies. La même régression logistique avec les textes sans les mots les plus genrés donne pour BOW 71% et pour WE 67%. Une des raisons pour laquelle nos résultats ne sont pas plus proches de 50% est que nous ne pouvons pas retirer toutes les informations de genre, notamment les prénoms.

Fasttext nous offre de nombreuses autres possibilités car le vecteur est bien plus petit et moins creux que BOW. Nous avons par exemple testé un algorithme de type arbre, XGBoost, qui nous donne des résultats de l'ordre de 1% meilleur.

3.2 Prédiction de la profession

Nous avons cherché à reproduire les résultats de l'article, à savoir prédire la profession à partir de la biographie. Nous avons ainsi par regression logistique obtenu ces différents résultats pour BOW et pour WE, avec un texte genré ou un texte non genré (où l'on a enlevé les mots les plus genrés comme expliqué dans la partie précédente) :

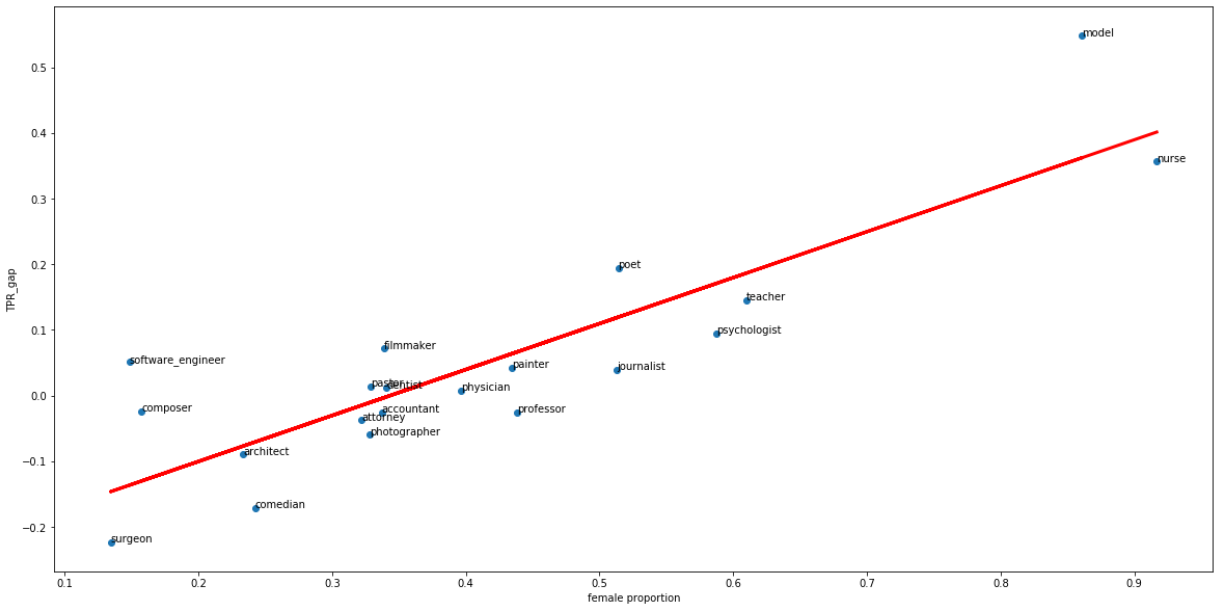

FIGURE 7 – TPR gap en fonction de la proportion de femmes, BOW, texte avec genre

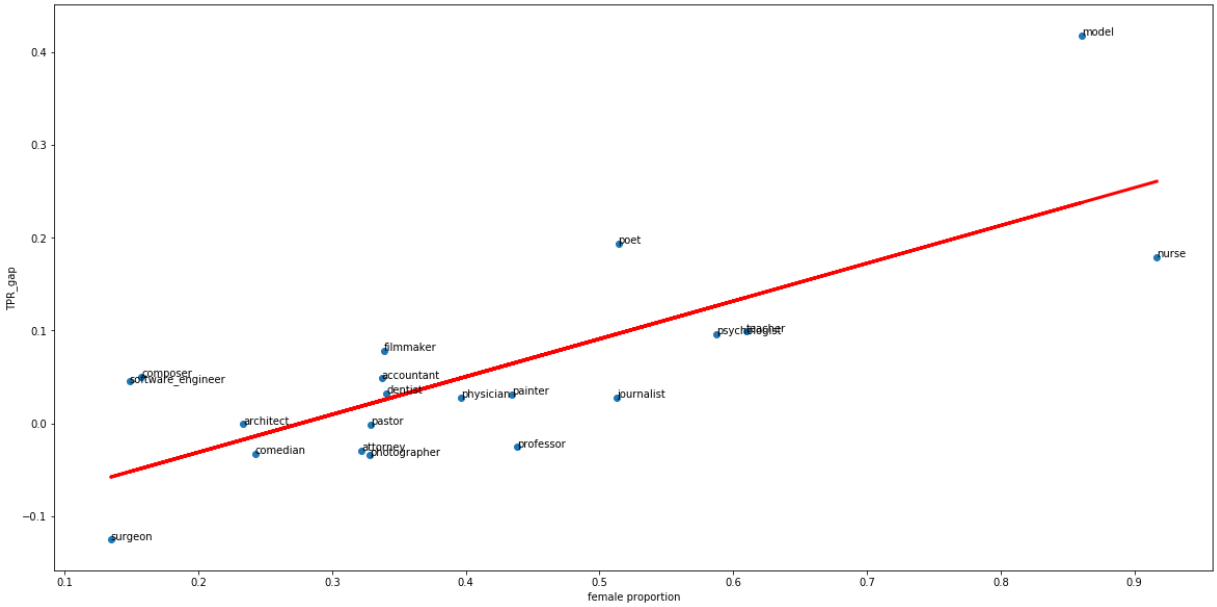

FIGURE 8 – TPR gap en fonction de la proportion de femmes, BOW, texte sans genre

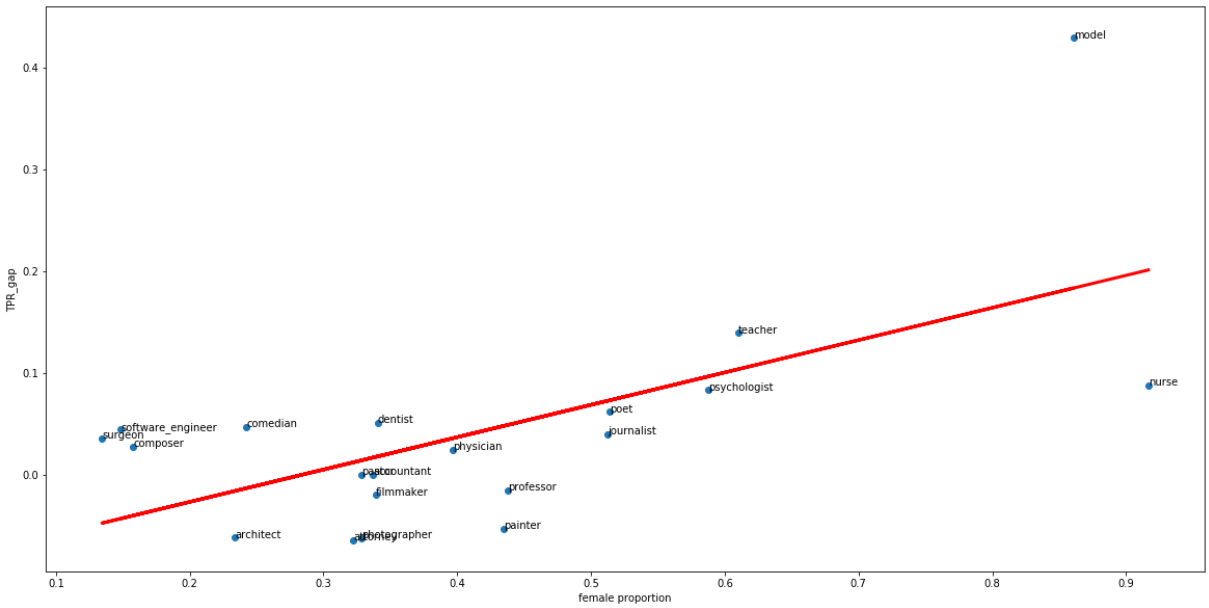

FIGURE 9 – TPR gap en fonction de la proportion de femmes, WE, texte avec genre

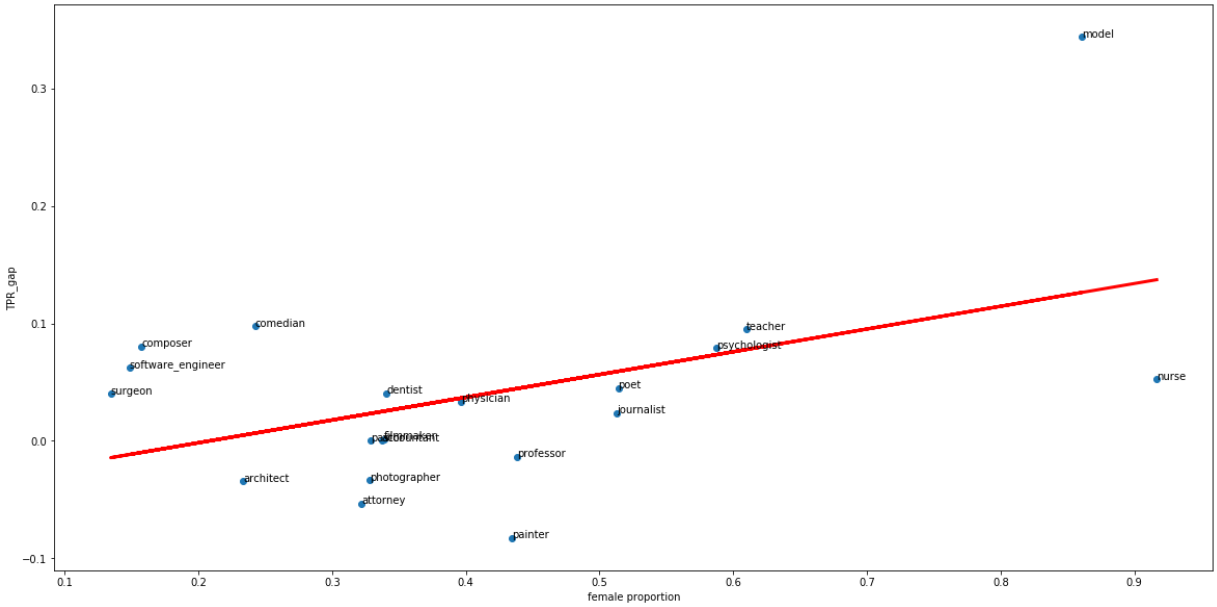

FIGURE 10 – TPR gap en fonction de la proportion de femmes, WE, texte sans genre

On peut regrouper ces résultats sous la forme d'un tableau regroupant les valeurs des coefficients directeurs des droites tracées :

	BOW	WE
Texte genré	0.70	0.32
Texte non genré	0.40	0.19

TABLE 1 – Coefficient directeur de la droite entre TPR gap et la proportion de femmes

Ces résultats semblent être cohérents, puisqu'enlever les mots les plus genrés entraîne une diminution du TPR gap à proportion de femmes fixée, ce qui signifie que les inégalités de genre ont été diminuées. Nos résultats sont toutefois quelque peu différents de ceux de l'article, puisque les auteurs trouvent en effet un coefficient directeur plus faible lorsque le texte est non genré, mais trouvent des résultats similaires pour BOW et WE ce qui n'est pas le cas pour nous où BOW est de l'ordre de deux fois plus discriminant que WE.

Un théorème important énoncé dans l'article est le théorème d'aggravation des inégalités, qui illustre également la pertinence du TPR gender gap pour quantifier les inégalités.

Théorème : Si $\pi_{g,y} < 0.5$ et $Gap_{g,y} < 0$, alors :

$P(G = g|Y = \hat{Y} = y) < \pi_{g,y}$

Démonstration : D'après l'égalité bayésienne :

$$P(G = g|Y = \hat{Y} = y) = \frac{\pi_{g,y} TPR_{g,y}}{P(\hat{Y} = y|Y = y)}$$

Ainsi,

$$\begin{aligned}\frac{P(G = g|Y = \hat{Y} = y)}{P(G =\sim g|Y = \hat{Y} = y)} &= \frac{\pi_{g,y} TPR_{g,y}}{\pi_{\sim g,y} TPR_{\sim g,y}} \\ &< \frac{\pi_{g,y}}{\pi_{\sim g,y}}\end{aligned}$$

D'où :

$$\frac{P(G = g|Y = \hat{Y} = y)}{1 - P(G = g|Y = \hat{Y} = y)} < \frac{\pi_{g,y}}{1 - \pi_{g,y}}$$

Ce qui, par croissance de la fonction $f : x \mapsto \frac{x}{1-x}$ sur $[0, 1[$ donne :

$$P(G = g|Y = \hat{Y} = y) < \pi_{g,y}$$

Concrètement, ce théorème signifie que si le gender gap d'un genre sous-representé est négatif pour une profession, alors ce genre sera encore plus sous-représenté dans l'ensemble des prédictions positives concernant cette profession. En effet, en réécrivant nous-mêmes les algorithmes, on obtient expérimentalement ce résultat : 15.3% des chirurgiens sont des femmes dans notre dataset, et on a effectivement $TPR_{femme,chirurgien} < 0$, et en effet 12.0% des vrais positifs concernant les chirurgiens sont des femmes, ce qui illustre bien le théorème.

4 Conclusion et continuité possible du travail

La problématique du Fair Learning nous apparaît comme très complexe et nécessite de nombreuses recherches afin d'aboutir à des algorithmes justes, d'autant plus que les algorithmes d'IA sont de plus en plus utilisés par des individus n'ayant aucune connaissance dans ce domaine (par exemple des recruteurs), et donc inconscient des dérives dont ces algorithmes sont capables en terme de discrimination.

Pour une utilisation juste, il est primordial de prendre en compte le théorème d'accroissement des inégalités. Le TPR gender gap apparaît comme une bonne métrique pour quantifier l'aspect équitable d'un algorithme. Ce travail a mis en lumière les corrélations entre le TPR gender gap et l'inégale répartition des hommes et des femmes dans les professions. Comme nous l'avons vu avec les coefficients directeurs des modèles linéaires, ces corrélations peuvent être atténuées en retirant les informations de genre dans les entrées. Or, comme nous l'avons montré en tentant de retirer de manière itérative des mots explicitement genrés, cette pratique est souvent soumise à des paramètres implicites ce qui la rend compliquée. Entre autres, notre discussion avec les auteurs du papier nous a appris que les hommes et les femmes avaient tendance à se décrire avec des tons différents ce qui est très dur à quantifier et donc à annihiler.

Une des principales difficultés a été de gérer des datasets très larges (dizaines de milliers de colonnes) avec des ressources en calcul limitées, ce qui nous a limité dans notre choix de modèle. De plus, il n'était pas pertinent de ne garder que les mots qui apparaissaient le plus souvent car

certains mots apparaissant peu jouaient un grand rôle dans la décision (les prénoms par exemple). Nous avons donc dû majoritairement nous restreindre à des modèles simples de type régression logistique capable d'apprendre efficacement sur des jeux de données particulièrement larges. Ce modèle simple a l'avantage d'être transparent et nous avons ainsi utilisé la valeur absolue des poids comme donnée d'importance des features.

L'utilisation des embeddings fasttext a permis de dépasser cette difficulté et donc d'utiliser d'autres algorithmes d'apprentissage. Ainsi, la précision de prédiction du genre avec le jeu de données sans informations sur le genre a augmenté avec l'utilisation de méthodes ensemblistes de type random forest ou XGBoost. Néanmoins, il nous a semblé que prendre la moyenne des embeddings sur la biographie entière réduisait la précision des modèles (en comparaison avec Bag of Words).

Nos ressources limitées en calcul ne nous ont pas permis d'entraîner efficacement le réseau récurrent avec attention et d'obtenir des résultats qui avaient du sens (voir notebook). Ceci est une voie d'amélioration de notre travail, la visualisation des poids d'attention du réseau offre en effet des perspectives d'explicabilité intéressante.

5 Remerciements

Nous tenons à remercier Jean-Michel Loubes pour son aide durant ce projet, que ce soit par son intervention à l'université d'Orsay qui nous a fortement aidé ou ces articles divers qui nous ont éclairé durant ce projet.

6 Bibliographie

Bias in Bios : A Case Study of Semantic Representation Bias in a High-Stakes Setting, Maria De-Arteaga, Alexey Romanov, Hanna Wallach, Jennifer Chayes, Christian Borgs, Alexandra Chouldechova, Sahin Geyik, Krishnaram Kenthapadi, Adam Tauman Kalai

Can Everyday AI be Ethical ? Philippe Besse, Céline Castets-Renard, Aurélien Garivier, Jean-Michel Loubes

http ://mccormickml.com/2016/04/19/word2vec-tutorial-the-skip-gram-model/

http ://mediamining.univ-lyon2.fr/people/guille/word$_e$*mbedding/cbow.html*

https ://github.com/JMLToulouse/Fair-ML-4-Ethical-AI