Traitement automatique de la langue : tâche classification du sexe

Etudiant : Léopold MAURICE

Tous les codes sont disponibles sur mon github LeoMaurice/nlp-lab-project.

Description de la méthode

Description des fichiers de données

On dispose de deux fichiers : *firstname_with_sex.csv* qui répertorie la fréquence des prénoms selon le sexe *transcriptions_with_sex.csv* qui contient chaque personne dont on doit classifier le genre avec comme informations : la transcription manuelle du texte de l'état civil, la prédiction de l'écriture visuelle de ce texte, une variable de sexe annotée manuelle. Cette dernière variable peut prendre 3 valeurs : homme, femme, ambigu. On reviendra sur les cas présentant cette dernière valeur. Les informations disponibles dans la transcription de l'état civil sont assez larges, inclut quasi toujours un prénom et un nom, parfois une fonction ou une relation au chef de famille. On espère qu'au-delà du prénom, les informations disponibles sont genrées, comme le terme chef de famille qui réfère plus fréquemment à des hommes, on reviendra sur ce point.

Pré-traitement des données

On travaille à partir de la prédiction du texte écrit et non à partir de la transcription manuelle, car le but est bien de construire in fine un pipeline qui se connecte à la prédiction ce qui n'est pas sans conséquence. A partir de cette prédiction, on récupère le prénom de la personne grâce des expressions régulières (REGEX), cela nous permet d'associer à chaque personne, la fréquence d'apparition de son prénom dans la population masculine à partir de la base *firstname_with_sex.csv*. Si le prénom n'apparait pas dans cette base, on utilise notre base d'entrainement pour compléter la base de fréquence, en excluant les personnes dont le genre est classé comme ambigu. Pour les personnes classées comme ambigües ou dont le prénom ne se trouve ni dans notre base de fréquence genrée des prénoms, ni dans notre base complétée, on leur associe une fréquence d'apparition masculine de 0.5 car on n'a pas d'information apriori permettant de pencher d'un côté ou de l'autre. Toutes les variables textuelles sont mises en minuscule par simplicité.

Description des données

On veut voir sur la Figure 1 que les données sont assez équilibrées en termes de représentation hommes, femmes, avec une légère surreprésentation des hommes, peut-être lié à leur statut traditionnel de chef de famille. On a seulement 7 personnes comme ambigües.

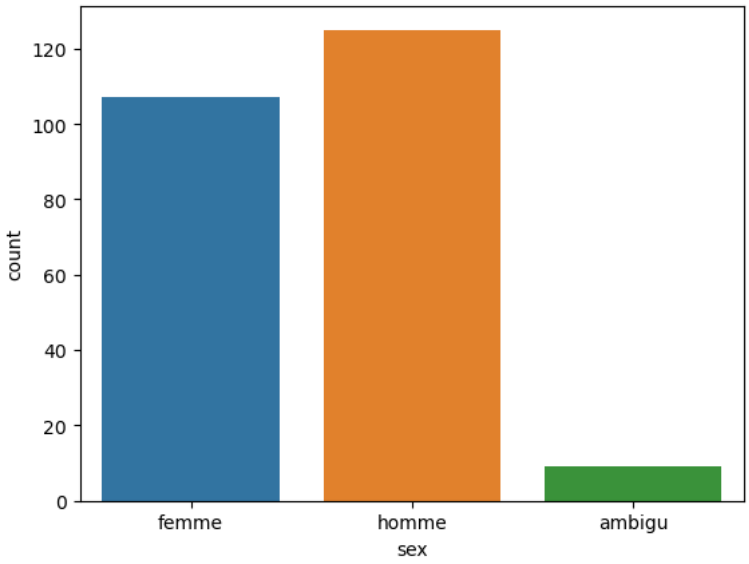

Figure 1: Barplot du nombre de personnes dans les données d'entrainement selon leur sexe annoté. Lecture : Il y a 125 hommes, 107 femmes et 7 personnes que les annotateurs n'ont pas pu classifier avec les données du recensement.

Pour se faire une première idée des données, on peut représenter pour nos trois catégories de sexe les prénoms les plus fréquents, sous la forme d'un wordcloud, présentée sur la Figure 2.

Figure 2: Wordcloud des prénoms en fonction du sexe. Lecture : Jean est le 1er prénom masculine, marie le premier prénom féminin.

On voit que les prénoms sont très genrées :

- Pour les hommes Jean, Louis, Jacques, Joseph, Jacques ressortent le plus
- Pour les femmes Marie, Marguerite, Jeanne, Madeleine, Maria ressortent le plus
- Pour les ambigües on a des prénoms très genrées comme Marie, Jeanne ou Claude et Emile. Cependant on a aussi des prénoms difficilement classables comme Vaude, et Entonie. On peut donc s'attendre à pourvoir réassigner le sexe de certains individus classifier en ambigu.

On peut aussi réaliser un wordcloud sur les informations de l'état civil, présenté sur la Figure 3. On a supprimé les mots de catégories comme « prénom : » pour mener une analyse des mots-clés présents.

Figure 3: Wordcloud des informations d'état civil retranscrites (prediction) en fonction du sexe. Lecture : les prénoms occupent une large place mais aussi certaines informations sont genrées comme chef, fils pour les hommes, femme pour les femmes.

On voit qu'au-delà des prénoms genrés par catégorie, on trouve des mots clés comme chef pour les hommes ou femme pour les femmes. On peut espérer qu'un modèle qui combine les informations de fréquence genrée de prénoms et les mots clés présents dans le recensement (soit avec un modèle simple avec une matrice de fréquence, soit avec un embedding) arrive à prédire le sexe.

Catégorie ambigüe

Dans les personnes classées comme « ambigu » on a des prénoms très genrées avec des prénoms donnés à plus de 98% à des hommes, et de prénoms donnés à 99% à des femmes. On va donc reclassifier la catégorie 'ambigu' avec l'information de fréquence genrée du prénom, suivant un critère simple : si les informations sont ambigües et que le prénom est porté en majorité par des hommes, on réattribue le sexe à 'homme', de façon symétrique pour des femmes. On garde le caractère ambigu si on a pas d'information (fréquence d'apparition du prénom de 0.5). C'est le cas pour deux personnes avec des prénoms bizarres nommées Vaude et Entonie. En réalité, si on regarde la groundtruth, Vaude s'appelle en fait Claude et Entonie s'appelle en réalité Antonie. On considère qu'on a ici une erreur de notre modèle de prédiction et on peut supprimer ces deux cas.

Modèle proposé

Comme on a pu le voir avec les nuages de mots, l'information de prénoms est déjà très genrée mais d'autres informations disponibles comme l'emploi ou le fait d'être désigné chef ou femme sont aussi très genrées. Je propose donc de construire un modèle qui accole une représentation vectorielle du texte prédit (`prediction`) et le ratio d'apparition du prénom pour les hommes (avec 0.5 comme valeur par défaut quand le prénom ou la fréquence n'est pas disponible comme expliqué ci-dessus). Plusieurs représentations vectorielles sont possibles, on en essayera deux :

- Une approche bag-of-words en mesurant pour chaque terme la fréquence d'apparition (document frequency matrix DFM). La matrice personne terme fréquence sera accolé aux vecteurs du ratio d'apparition du prénom pour les hommes en population général.

- Une approche par embeddings : on accolera pour chaque texte `prediction` le vecteur moyen de **Fast Text**, le vecteur de la phrase en entier généré par **Sentence-BERT** (SBERT), et le ratio d'apparition pour les hommes comme dans l'approche bag-of-word.

Dans le cas de l'approche bag-of-words, on filtrera les mots catégoriels comme "prénom:" ou "date_naissance:", car dans une approche bag-of-words, le modèle ne pourra relier un sens à ces termes, leurs fréquences d'apparition viendra juste parasiter et écraser les fréquences d'apparition des mots avec du sens comme "chef" ou "femme.

Dans l'approche par embedding, on va garder au contraire ces termes car on peut espérer notamment que l'embedding de BERT soit capable de garder sémantiquement que "profession: cuisinière" contient une information genrée.

Pour chacune des matrices ainsi construites, on va construire un modèle à partir de *LinearSVC* (bibliothèque *sklearn*, encapsulation de *Liblinear*) pour ensuite prédire le sexe en fonction de la matrice. Le choix de cette méthode de classification a été fait en testant plusieurs méthodes dont un bayésien naïf.

Approche bag-of-word, accolée aux ratios d'apparition du prénom chez les hommes en population générale

Le F1-score obtenu sur le test set (20% choisies aléatoirement sur l'ensemble des données) on obtient une précision de 0.98. Les différentes métriques sont présentées dans le tableau suivant :

	precision	recall	f1-score	support
femme	0.96	1.00	0.98	22
homme	1.00	0.96	0.98	26
accuracy			0.98	48
macro avg	0.98	0.98	0.98	48
weighted avg	0.98	0.98	0.98	48

On peut voir sur notre matrice de confusion Figure 4 qu'un seul homme est mal classifé comme femme. En regardant au plus près on se rend compte que cet homme est dénommé Simone dans *prediction* et Simon dans *groundtruth*, son vrai métier est cultivateur mais marqué comme cuisinière dans *prediction*. On comprend donc que l'erreur vient du modèle de prédiction des transcriptions et non de notre modèle de classification du sexe qui avait des informations typées féminines.

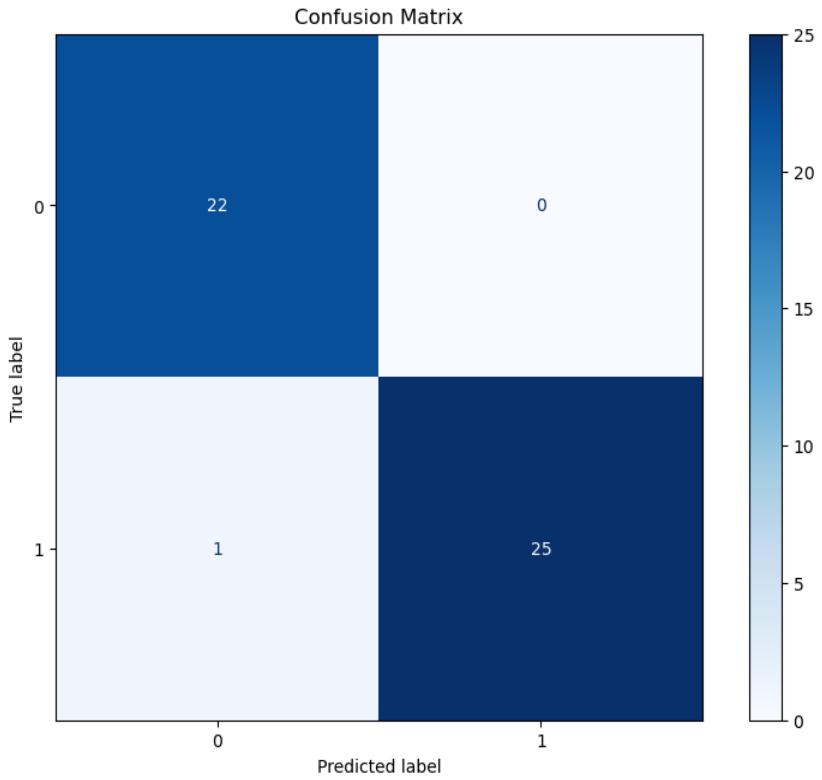

Figure 4: Matrice de confusion du modèle par bag-of-words. Lecture: 1 = homme, 0 = femme. Un homme est classifée par notre modèle comme femme.

Approche embedding, accolée aux ratios d'apparition des prénoms chez les hommes

L'approche par embedding donne des résultats similaires comme montrer sur le tableau de résultat suivant :

```
Classification Report:
              precision    recall  f1-score   support

       femme       0.96      1.00      0.98        22
       homme       1.00      0.96      0.98        26

    accuracy                           0.98        48
   macro avg       0.98      0.98      0.98        48
weighted avg       0.98      0.98      0.98        48
```

On a la même erreur avec Simon/Simone dans notre test set. Par contre, le temps de calcul est bien plus élevé pour l'embedding de l'ordre de plusieurs minutes, pour quelques secondes pour l'approche bag-of-words. Avec une cross-validation, on obtient un F1-score moyen de 0.98 ce qui est très correcte.

Tests avec le groundtruth pour vérifier l'effet des erreurs de transcription

En utilisant l'un de nos deux modèles, ici celui par embedding, sur la variable groundtruth (transcription manuelle des informations d'état civil). On voit que le cas Simon/Simone

marche bien Simon est bien classifié comme homme, comme on peut le voir sur la Figure 5. Un autre cas bizarre apparait dans le train set. Une personne dénommée Marie dans la groundtruth et dans la prédiction est annotée comme homme, mais notre modèle la classe comme femme car sa profession est cuisinière. On ne peut pas vraiment savoir si l'erreur vient des annotations ou de notre modèle.

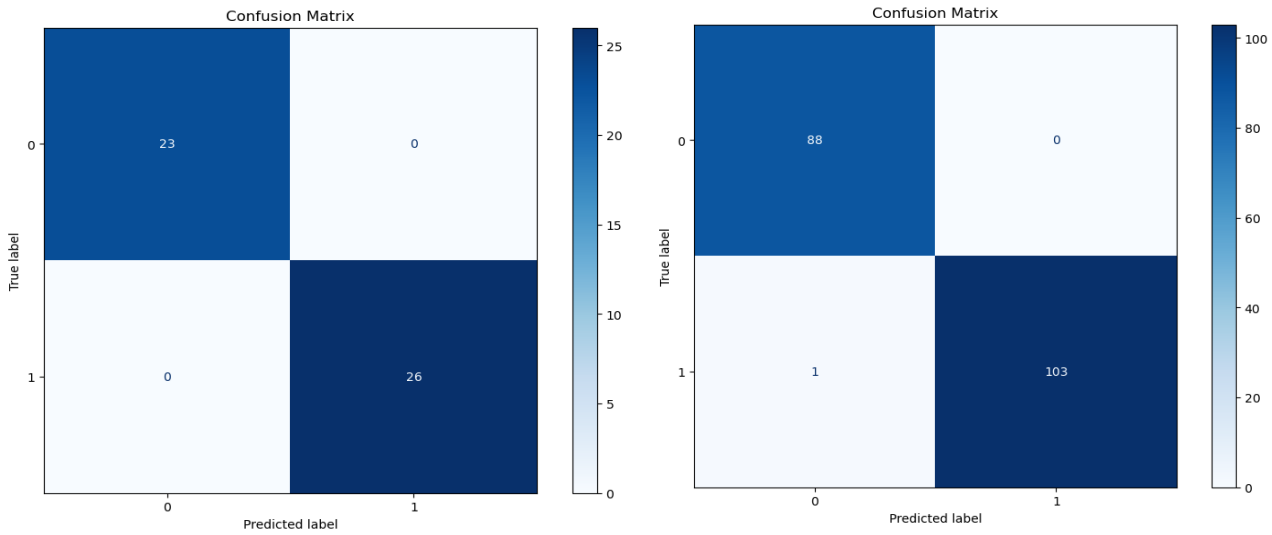

Figure 5: A gauche, matrice de confusion de la prédiction sur groundtruth avec le test set. Droite: prédiction sur la groundtruth du train set. Le modèle par embedding a été utilisé.

Implémentation sur l'ensemble du corpus

Comme on a pu le voir, les performances des deux modèles sont très équivalentes. Les prénoms sont déjà très genrées et il est probable qu'un modèle simple utilisant simplement la fréquence du prénom en population générale suffisent. Cependant, l'ajout d'information par mots-clés comme chef, ou femme parait pertinentes : les seules erreurs réalisées par nos modèles sont en réalité des incohérences entre la groundtruth, la prédiction de la transcription, et l'annotation manuelle du sexe. Pour finir, comme l'embeddings prend beaucoup plus de temps et ne rajoute pas de gain de performance, il me parait que le modèle le plus adapté a une implémentation réelle est celui par l'approche bag-of-words, donc le pipeline suivant :

1. Extraire de la prédiction le prénom grâce à des expressions régulières, récupérer le ratio d'apparition du prénom chez les hommes en population générale. Utiliser la valeur 0.5 par défaut (prénom non existant)

2. Calculer la matrice de fréquences de termes de la prédiction de la transcription en supprimant les termes comme « prénom : ». Agrandir cette matrice de fréquence par le vecteur des fréquences d'apparition du prénom chez les hommes en population générale.

3. Prédire le sexe à partir de notre modèle entrainé sur données annotées.