# Nettoyage et enrichissement du fichier contestants.csv dans OpenRefine

## Création du projet

Je créé mon dossier, **M2_DEFI_2023_Guerin_Eurovision,** en important le fichier contestants.csv.

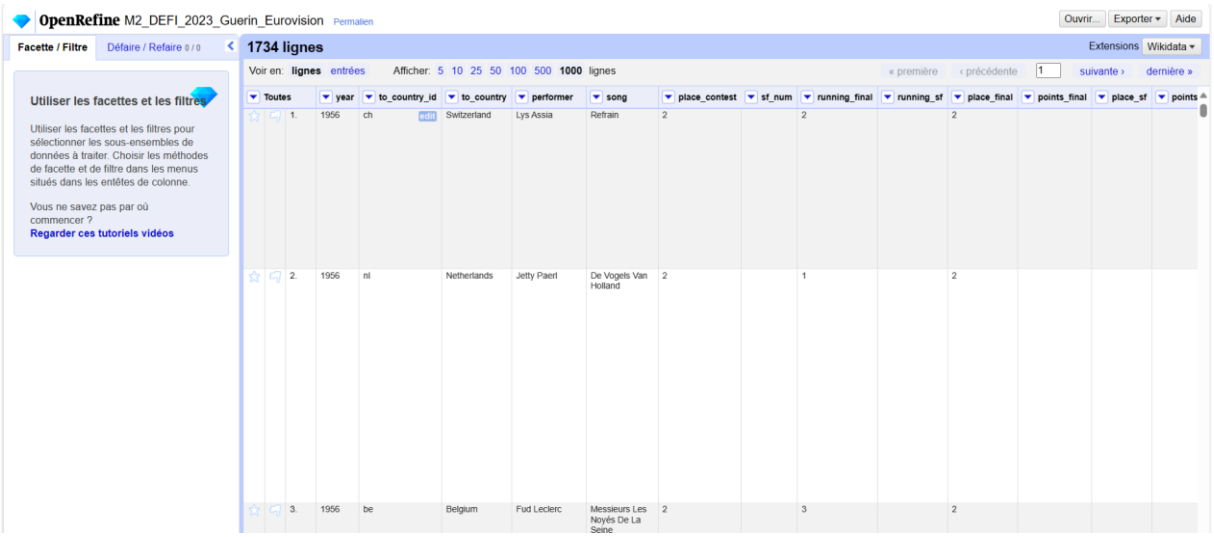


*Figure 1 Création du projet*

## Suppression de colonnes

Suppression de la colonne « to_country_id » pour ne se baser que sur « to_country » dans mon analyse.

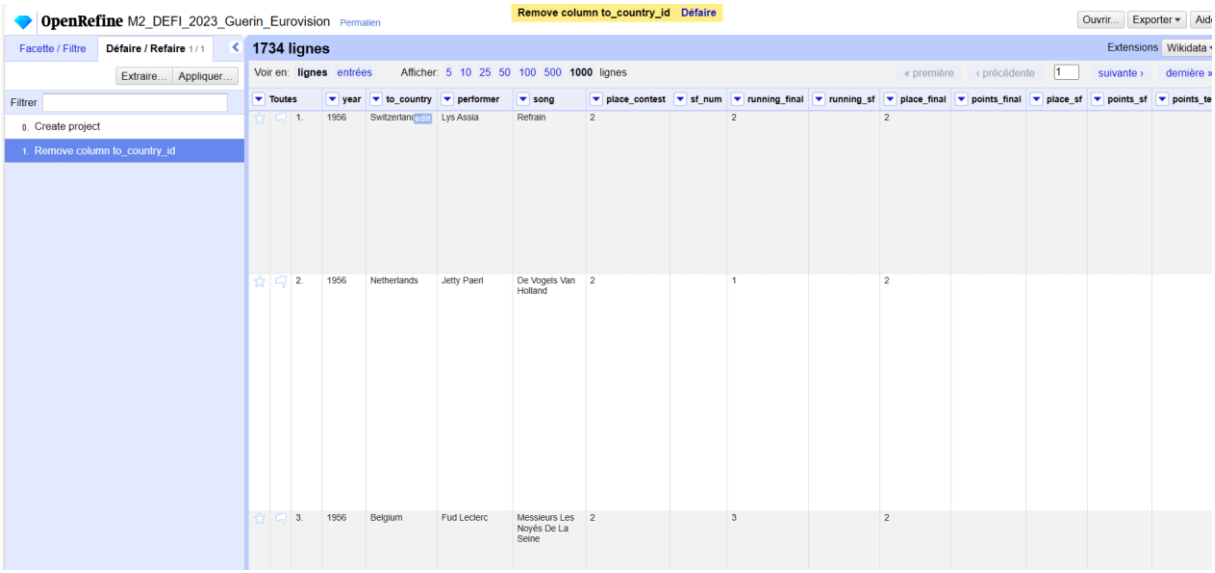


*Figure 2 Suppression de la colonne to_country*

Suppression des colonnes concernant les demi-finales (sf_num, running_sf,place_sf, points_sf, points_tele_sf,points_jury_sf) puisqu'on s'intéresse aux participants des finales (et plus précisément des gagnants). De plus, la France ne participant pas aux demi-finales (membre du Big Five), il n'y a pas d'intérêt.

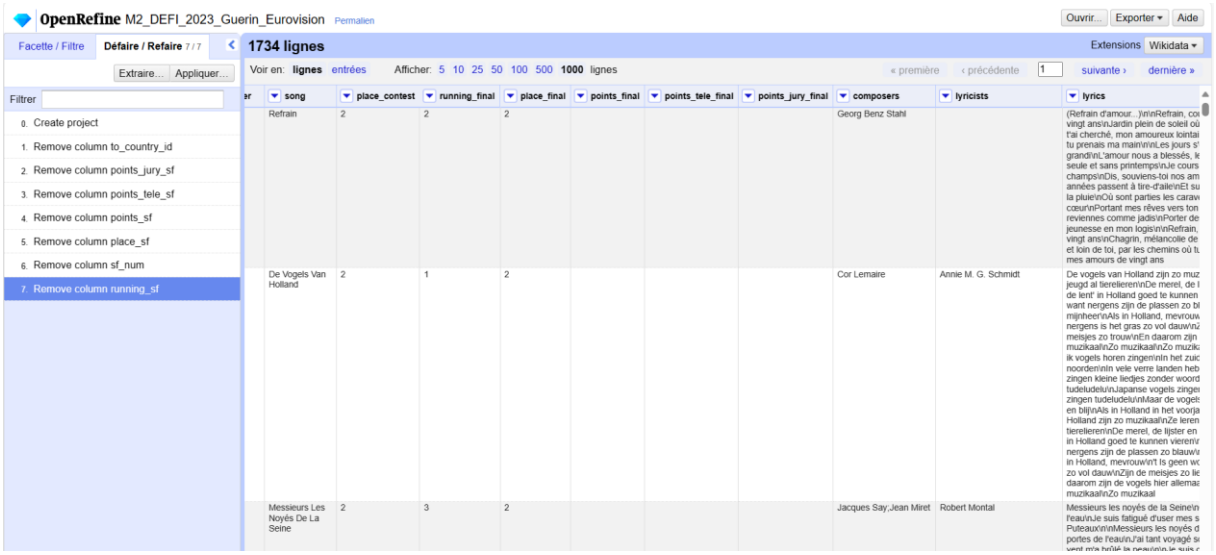


*Figure 3 Suppression des colonnes concernant les demi-finales*

Supprimer les colonnes composers et lyricists, inutiles pour notre recherche.

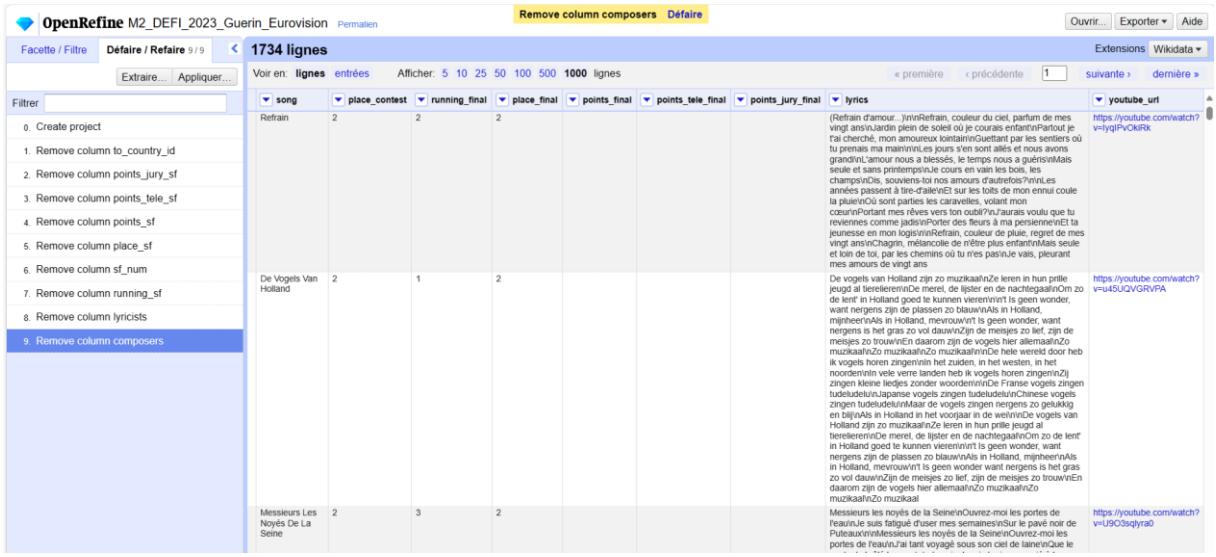


*Figure 4 Suppression des colonnes composers et lyricists*

## Enrichissement des pays via WikiData

Je fais correspondre les pays avec l'entité « country » de Wikidata, pour ensuite pouvoir récupérer les données de géolocalisation. Je m'aperçois néanmoins qu'il faut faire souvent le match manuellement car WikiData propose le pays à travers les siècles (par exemple Royaume de France pour la France). Fort heureusement, j'applique le choix pour chaque pays sur plusieurs lignes à chaque fois.

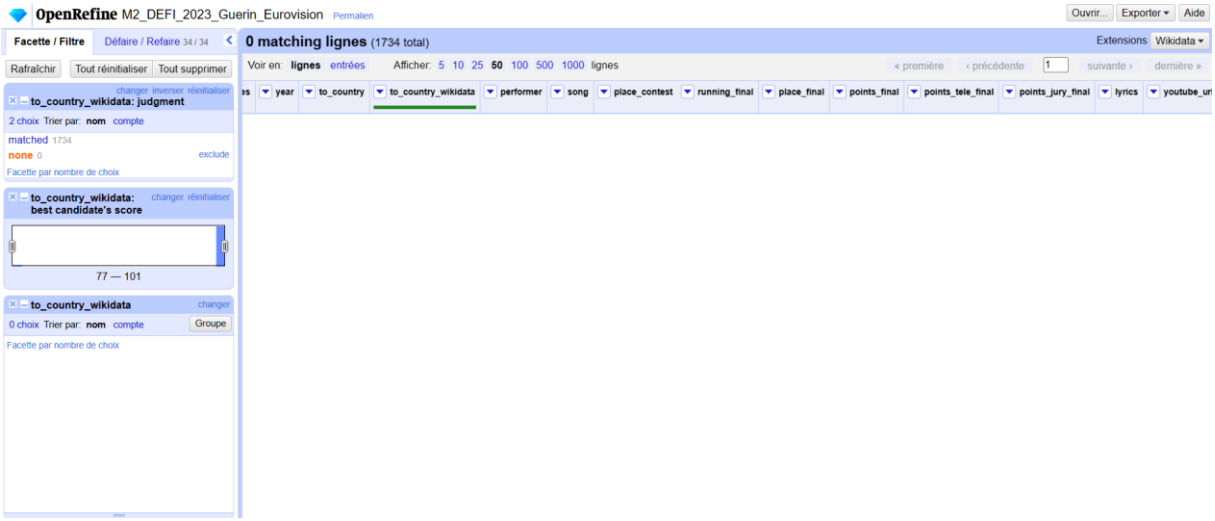


*Figure 5 Enrichissement des pays via Wikidata*

Ajout de la localisation géographique : pour ce faire, je pars de « to_country_wikidata » et je demande à avoir l'entité « coordinate location ».

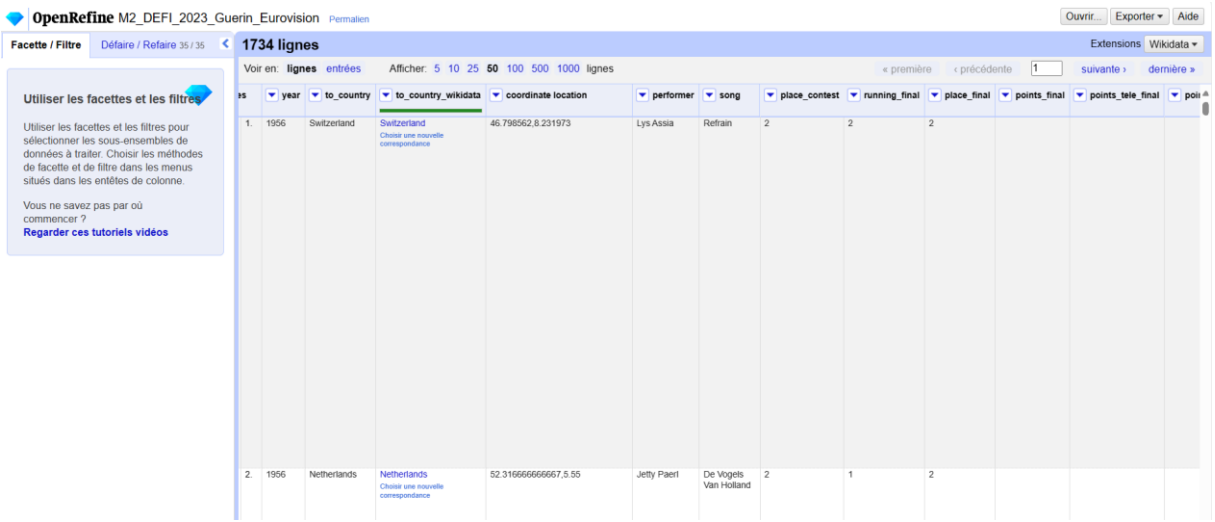


*Figure 6 Ajout de la localisation géographique de chaque pays ayant participé à l'Eurovision*

## Formatage de l'année

Je créé une colonne « year_formatUS » pour avoir le bon format pour les outils de visualisation. Je procède de la manière suivante : je créé une colonne (« year_formatUs ») en me basant sur year, que je transforme en date. Puis, je transforme ces dates de nouveau en texte. Je fais un chercher/remplacer pour enlever le bout d'heure.

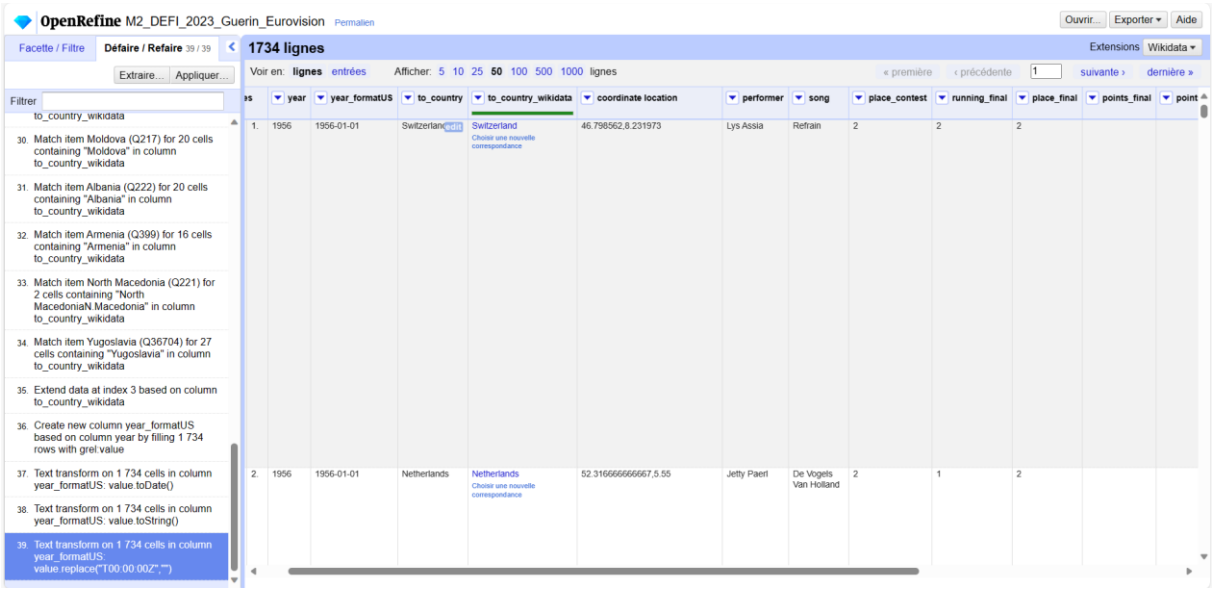


*Figure 7 Création d'une colonne au format américain*

## Enrichissement des artistes

Je lie les artistes (« performer ») à leur entité « human » de Wikidata. Pour ce faire, je duplique ma colonne « performer » en « performer_wikidata ». Je créé une colonne « edition » qui est la concaténation de « Eurovision Song Contest » et de la valeur année (year). Le but est d'utiliser cette colonne pour avoir un meilleur taux de match sur la colonne performer_wikidata puisqu'il existe l'entité « participant in » sur les fiches des artistes. Après cette première opération il y a 589 not matched. Je tente de réconcilier les not match avec musical group : il reste 284 not matched. Puis je tente sur musical duo : il reste 278 not match. Je décide de les laisser pour l'instant de côté.

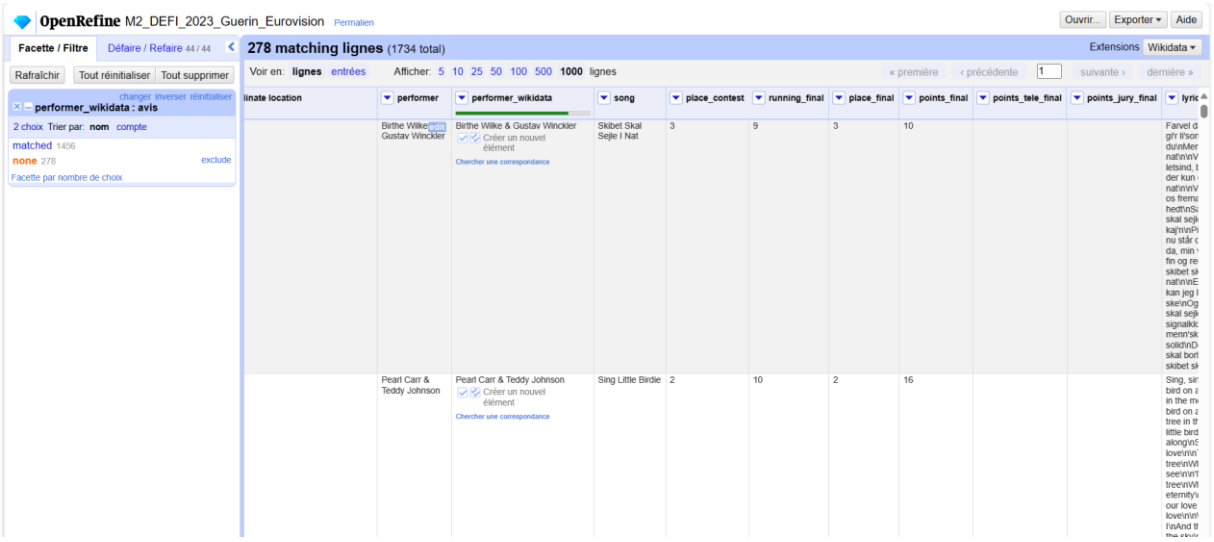


*Figure 8 API WikiData pour les artistes (human, musical duo, musical group)*

Pour les match (1456 lignes), je cherche le genre (homme/femme) via Wikidata avec la propriété gender. Une fois ces informations récoltées, je passe ces informations en texte en supprimant le lien avec WikiData.

Sur ces 1456 lignes d'artistes identifiées, il y a 311 cellules vides. Pour ces cellules vides, je remplis « non applicable » car ce sont les groupes / duos qui restent et qui ne peuvent pas avoir de genre.

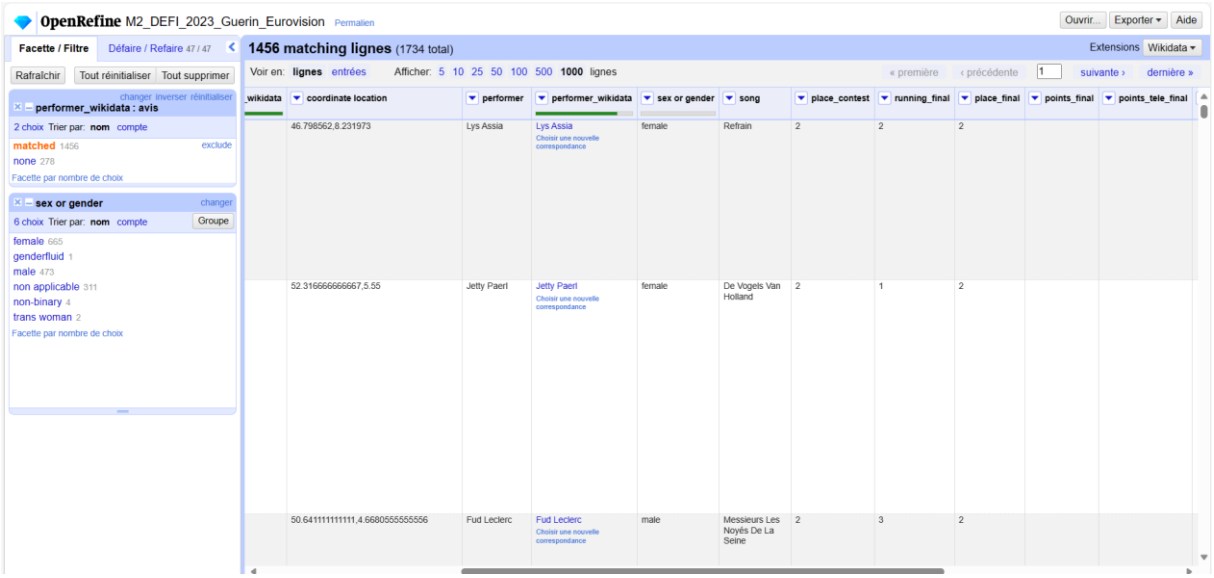


*Figure 9 Genre de l'artiste pour les artistes identifiés*

Je créé une colonne « performer_type » à partir de celle concernant le genre. Quand la case indique non applicable, je remplace par groupe. Si elle indique male / female / etc, j'indique solo.

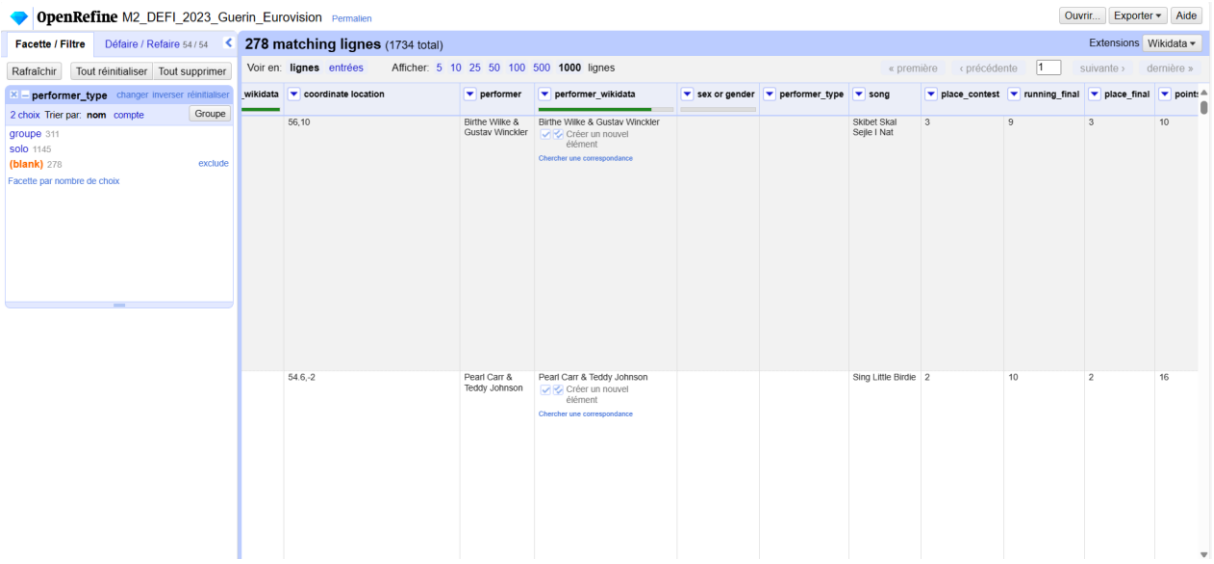


*Figure 10 Création des colonnes gender et performer_type*

Je créé une colonne test pour les 278 lignes des artistes non trouvés. Ensuite, je fais une recherche textuelle de « & » et « and » pour remplir les cases sur le genre (non applicable : ce sont des groupes) et type de performance en groupe.

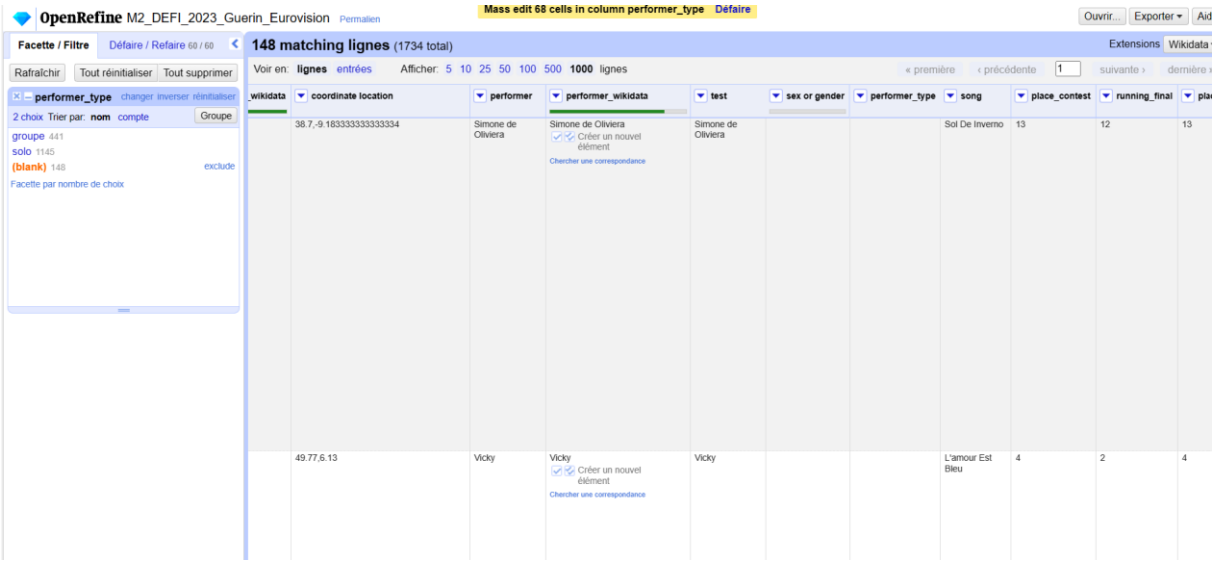


*Figure 11 Remplissage des groupes en & et and*

Je passe à 148 lignes encore incomplètes, que je complèterai après export. Je supprime la colonne « test » et « performer_wikidata » car le genre et le type d'artistes étaient les seules choses qui m'intéressaient. Je garde finalement la colonne « youtube_url » car elle va me permettre de voir pour ces 148 lignes directement si c'est un groupe, une femme, un homme  etc.

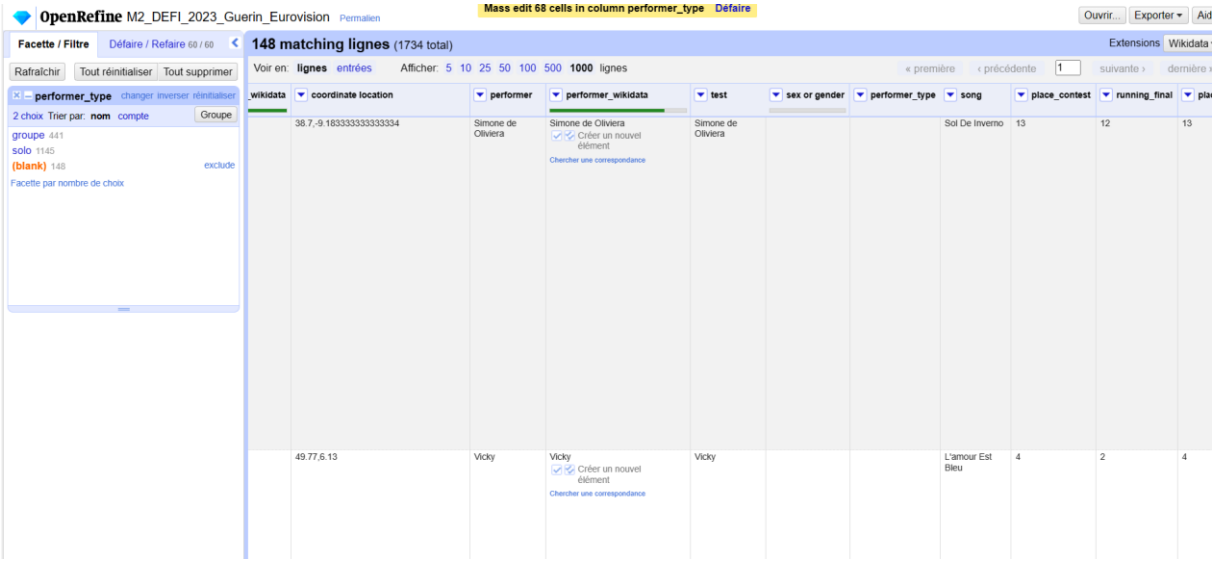


*Figure 12 Les 148 artistes qu'il reste à traiter après export*

## Autres suppressions de colonne et uniformisation sur place_contest

Je supprime les colonnes « points_final », « points_tele_final », « points_jury_final » car c'est la place obtenue (« place_contest ») qui m'intéressent. De plus, les évolutions du concours font que ce n'est pas pertinent de prendre ces paramètres en compte.

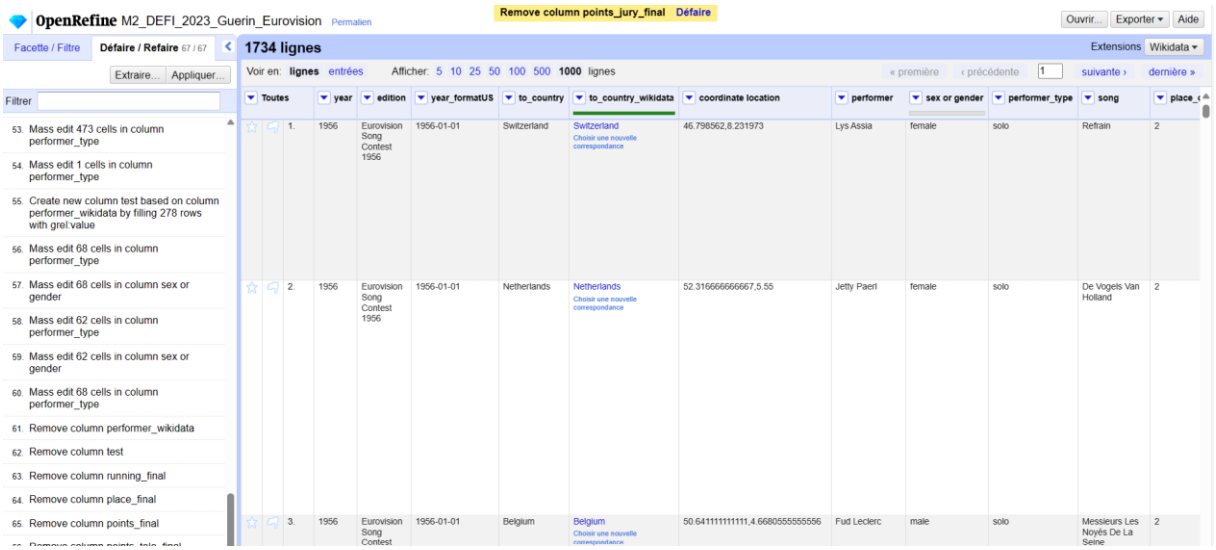


*Figure 13 Suppression des colonnes*

Pour la colonne place_contest, j'uniformise les 1.0 par exemple en 1. Pour ce faire j'ai cherché dans la colonne tous les « .0 » puis j'ai fait un chercher/remplacer.

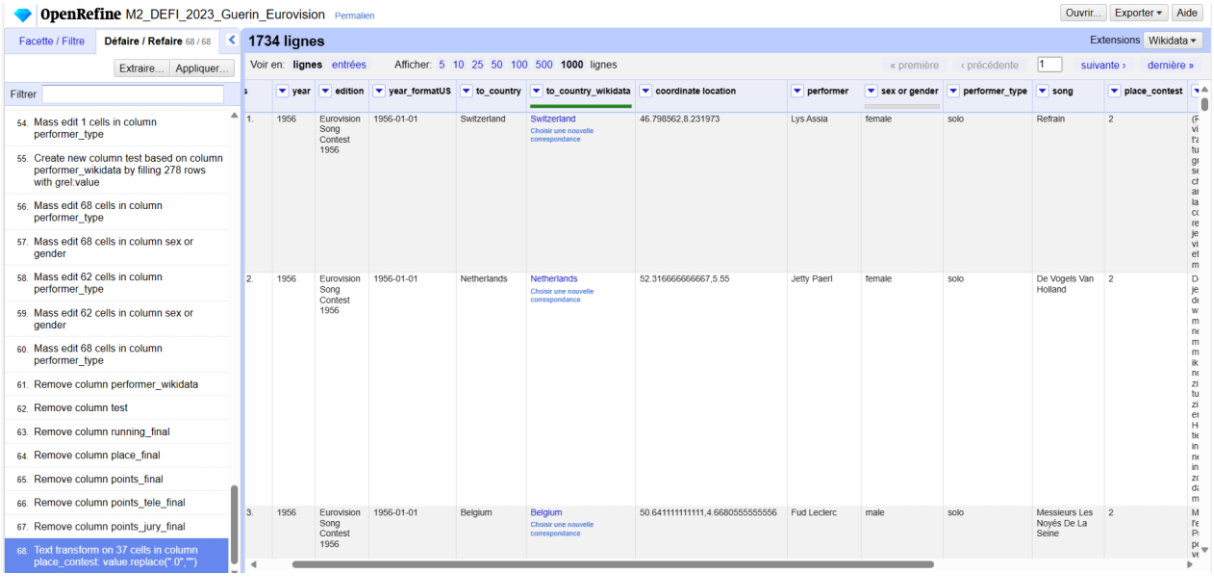


*Figure 14 Uniformisation de place_contest*

Il reste 56 cellules blanches.  En affinant avec l'année, je me rends compte qu'il y a les 40 chansons pour l'année 2020 qui a été annulée. Je me positionne sur l'année 2020 et je mets dans la colonne place_contest « non applicable ». Il reste 15 chansons pour l'année 2023. Je les remplirai une fois l'export fait.

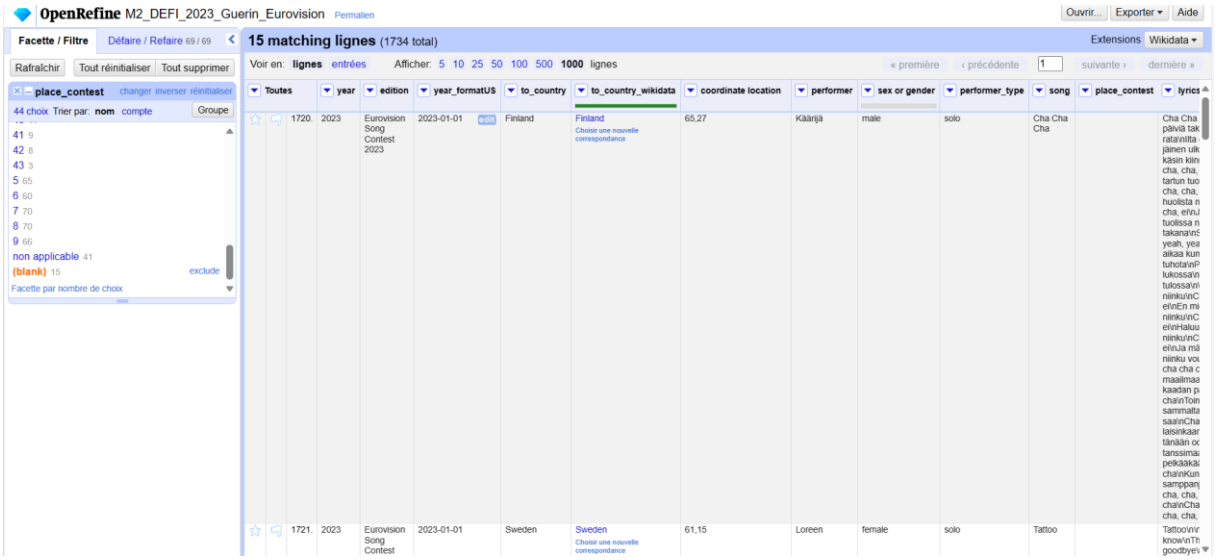


*Figure 15 Les 15 chansons à positionner après export*

## Enrichissement des chansons

Je voulais aller chercher la langue de chaque chanson, mais le résultat n'était pas concluant via WikiData (284 match). Je me dis que je pourrais partir des lyrics et trouver un outil qui ferait le match automatiquement. Pour ce faire je créé un script python (voir langue.py) et j'exporte le résultat dans liste_langue.csv.

Je veux joindre ma nouvelle colonne de langue dans mon projet OpenRefine original. Pour ce faire, je créé un nouveau projet avec langue.csv qui s'appelle ajout_langue. Puis, je me place dans mon projet original et je fais la requête suivante :

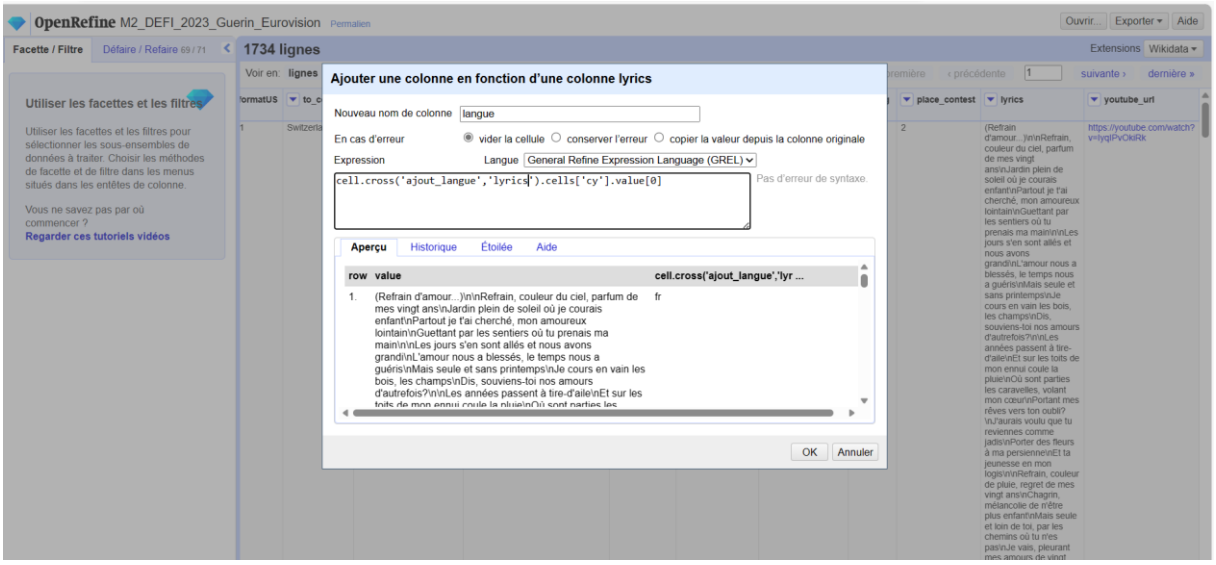


*Figure 16 Ajout de la colonne des langues*

Ma colonne langue est créé avec les valeurs récupérées. Il n'y a que douze valeurs blanches, que je traiterai après export.

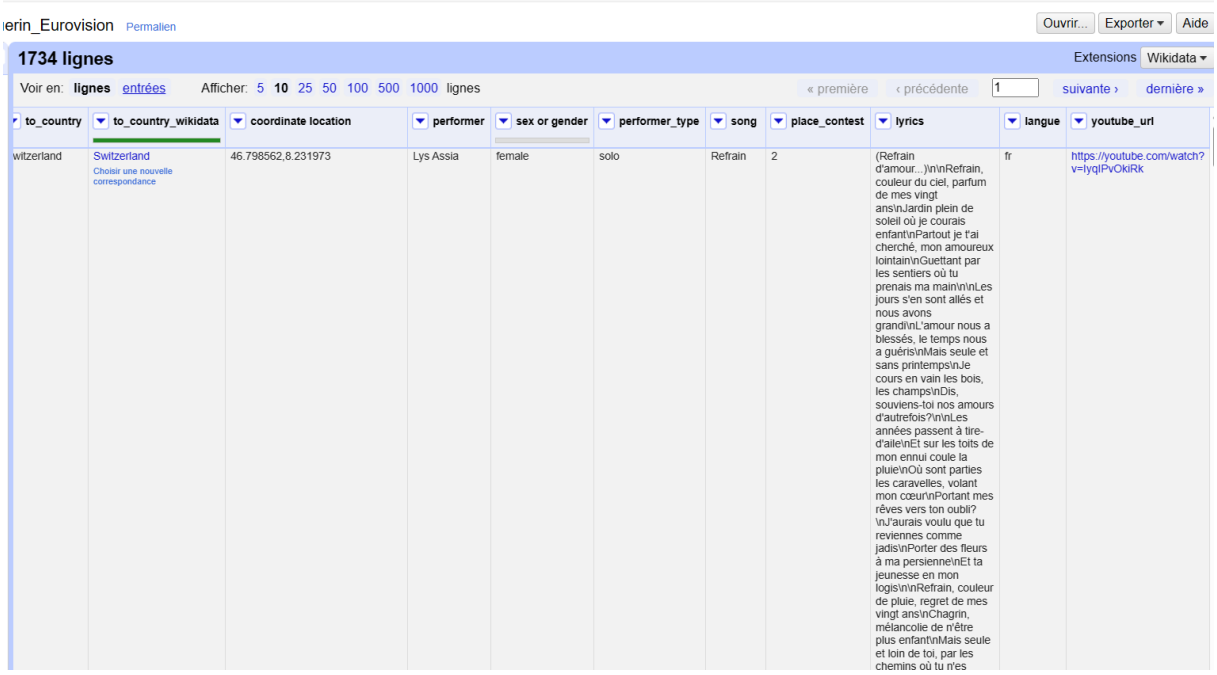


*Figure 17 Résultat final dans OpenRefine*

Je peux supprimer mon projet ajout_langue.

## Export de l'historique et du fichier

J'exporte mon historique, ainsi que mon fichier au format csv.

Historique : M2_DEFI_2023_Guerin_Eurovision_Historique.json

Export au format CSV : M2_DEFI_2023_Guerin_Eurovision_ExportOpenRefine.csv

Export au format EXCEL : M2_DEFI_2023_Guerin_Eurovision_ExportOpenRefine.xlsx

## Traitement Post OpenRefine

Pour rappel je dois compléter :

- 148 lignes pour le genre et le type d'artistes
- 15 places au concours
- 12 chansons avec langues manquantes.

Je duplique mon fichier et je travaille sur ce fichier :
M2_DEFI_2023_Guerin_Eurovision_PostOpenRefine.csv

Je commence par les places au concours (place_contest). Je me rends compte qu'il y a des doublons : Loreen (gagnante de 2023) apparait deux fois. Je pense qu'il y a eu un petit bug (dans le scrapping ou moi en faisant une manipulation). Je supprime du coup mes 12 lignes en trop.

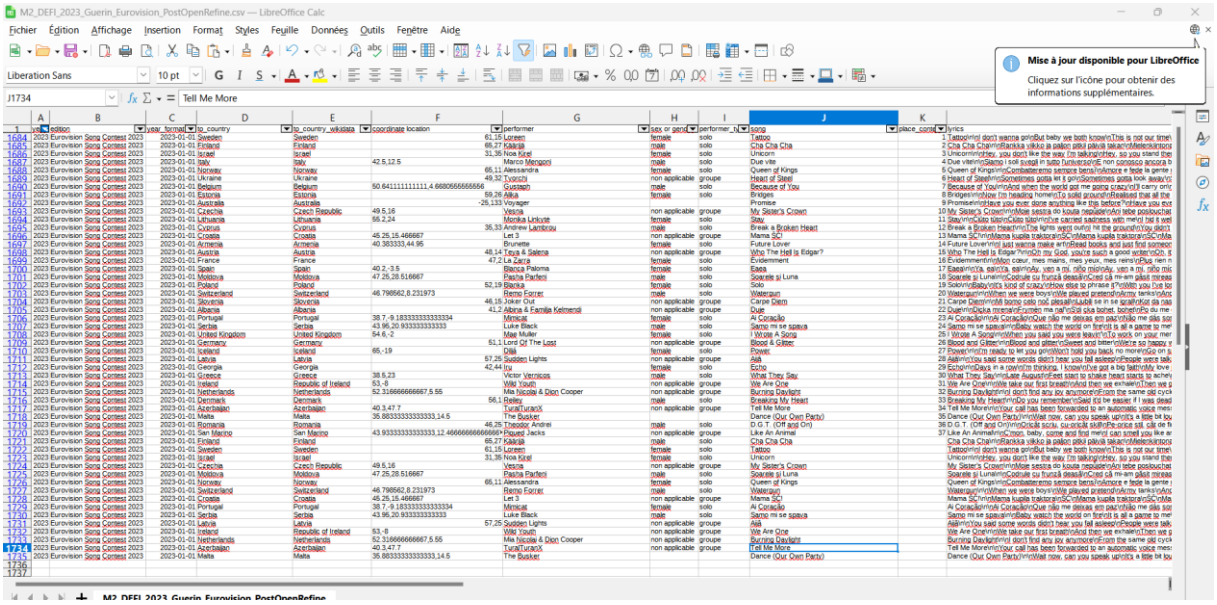

*Figure 18 Lignes en double*

Pour les 12 chansons sans langue, je vois qu'il y a eu un décalage sur une ligne : un bout des lyrics est dans la case langue et donc, la langue est seule dans 3 colonnes. Je fais les corrections et je supprime les trois colonnes créés (peut-être un souci dans mon script python ?).

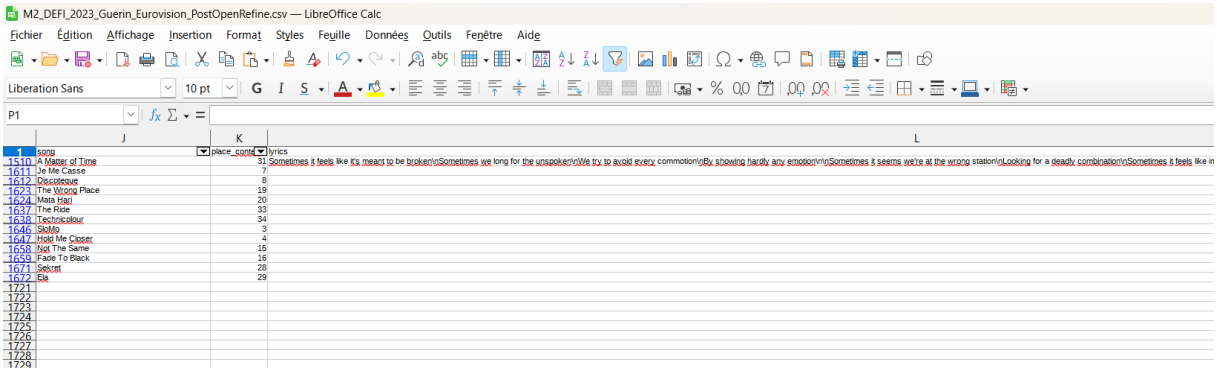

*Figure 19 Langues manquantes*

Il me reste 147 artistes à traiter, que je traite manuellement. Ensuite, il restait deux chansons sans titre. Une fois cette dernière opération, mon fichier est prêt pour utilisation.

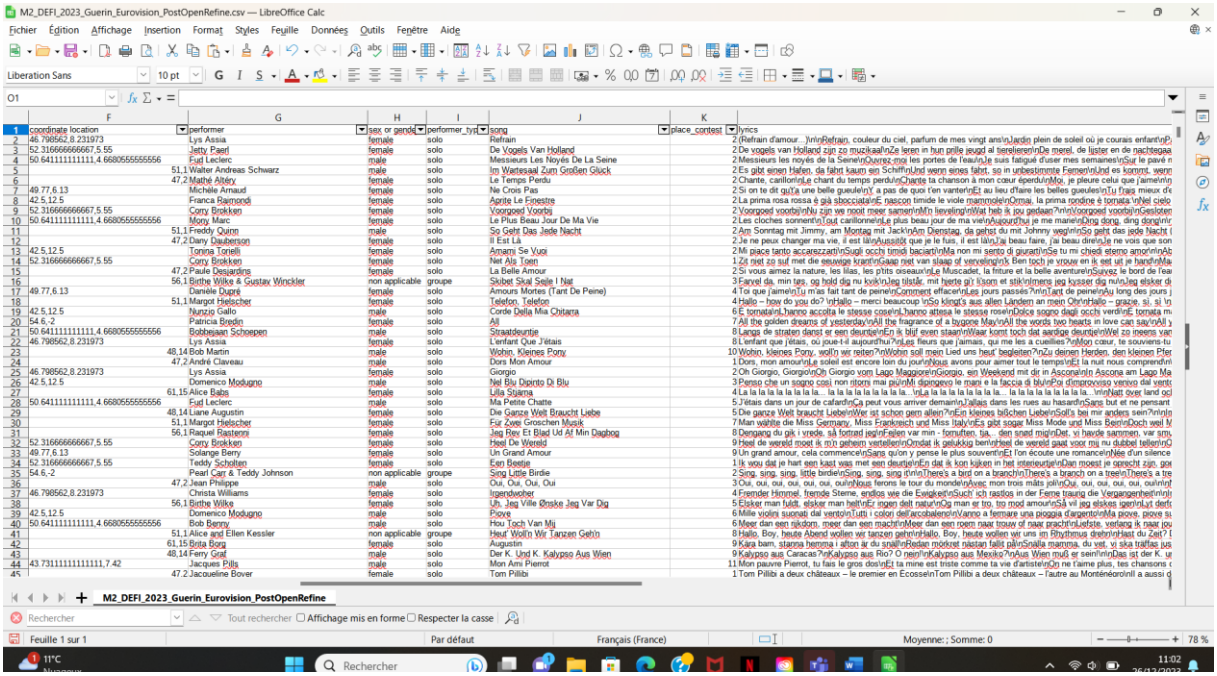


*Figure 20 Fichier final*