Wrangle ACT

Table des matières

INTRODUCTION

L'ensemble des données à étudier dans le cadre de ce projet est l'archive de tweets de l'utilisateur de Twitter @dog_rates, également connu sous le nom de **WeRateDogs**. Le but de ce document est d'évaluer les problèmes de qualité observés parmi les différents tableaux qui ont été évalués. Ainsi il s'agit pour nous ici dans un premier temps de documenter **les problèmes de qualités et d'ordre observés(I)** et ensuite présenter **les plans de nettoyage de ces données(II)** afin de les rendre plus digeste a la visualisation.

I- Problèmes de qualité et d'ordre

Les problèmes de qualité font références a des problèmes de contenu comme des données inexactes, des données manquantes, dupliquées ou incorrectes. Les problèmes d'ordre font plutôt référence à la structure même des données qui ont été étudiés. Ainsi nous allons recenser dans un premier temps les problèmes de **qualités**(1) observés dans chacun des tableaux de façon individuel et dans un second plan s'attarder aux problèmes **d'ordre** (2**)**.

1- Les problèmes de qualité :

Nous les avons recensés suivant les différents tableaux que nous avons eu à étudier. Nous avons eu à utiliser des abréviations pour faciliter leurs manipulations. Nous avons recensé ces abréviations dans le tableau suivant:

N	Dénomination	Abréviation
1	twitter-archive-enhanced.csv	Tableau_1
2	image_predictions.tsv	Tableau_2
3	tweet_json.txt	Tableau_3

De plus nous allons classifier les problèmes de qualité en trois grandes catégories Les problèmes de **complétude**, et **de cohérence**. Toutes nos observations sont résumées dans le tableau suivant :

N	Libelle	Tab	Colonnes	Catégorie	Observation
1	Valeurs manquantes	1	in_reply_to_status_id	Complétude	Nous avons 2356 entrées mais juste 78 valeurs pour cette colonne
2	Valeurs manquantes	1	in_reply_to_user_id	Complétude	Nous avons 2356 entrées mais juste 78 valeurs pour cette colonne
3	Valeurs manquantes	1	retweeted_status_id	Complétude	Nous avons 2356 entrées mais juste 78 valeurs pour cette colonne
4	Valeurs manquantes	1	retweeted_status_user_id	Complétude	Nous avons 2356 entrées mais juste 78 valeurs pour cette colonne
5	Valeurs manquantes	1	retweeted_status_timestamp	Complétude	Nous avons 2356 entrées mais juste 181 valeurs pour cette colonne
6	Valeurs manquantes	1	expanded_urls	Complétude	Nous avons 2356 entrées mais juste 2297 valeurs pour cette colonne
7	Format de données erroné	1	timestamp	Cohérence	Le type est object au lieu de time date
8	Format des valeurs manquantes	1	doggo, floofer ,pupper, puppo	Cohérence	Les entrées avec les valeurs manquantes sont nulles
9	Format de noms differents les uns des autres	1	name	Cohérence	Certains noms commencent par une majuscule d'autres par une minuscule
10	Format erroné	2	P1,P2,P3	Cohérence	Tous les noms des chiens ont un tiret comme des colonnes
11	Valeurs erronées	1	rating_numerator,rating_denominator	Exactitude	Les données collectées sont erronées pour certains

2- Problèmes d'ordre :

Il suit la même structure que celle de la qualité nous allons résumer nos observations par un tableau récapitulatif de ce que nous avons remarqué :

N	Type de problème	Tab	libéllé
12	Structure	1	*Les colonnes* **rating_numerator** *et* **rating_denominator** *du tableau_1 violent la première règle de rangement qui veut que chaque variable forment une seule colonne en effet ces colonnes sont en fait des éléments d'une seule colonne* **rating** *qui est la note attribuée à chacune des observations effectuée*
13	Structure	1	Les dénominations doggo,floofer,pupper,puppo sont des labels de chiens et devraient être supprimés pour ne faire place qu'a une seule colonne appelée label qui répertorie toutes ces labels.
14	Structure	2	Créer une colonne dans image_predictions qui va donner le résultat des prédictions sur les différentes races de chiens
15	Structure	1	Regrouper le tableau1,2,3 en un seul tableau

II- Nettoyage des données :

Pour chacun des problèmes de qualité et d'ordre mentionné ci-dessus nous allons dans un tableau de synthèse. Nous aurons ainsi les étapes pour résoudre le problème et le code utilisé pour le faire. Pour cela Nous repartirons les solutions en deux grandes parties : le nettoyage lié à la complétude (problème de données manquantes), ensuite nous allons nous pencher sur les autres problèmes de qualité et enfin nous allons résoudre les problèmes liés à l'ordre.

a- Nettoyage des problèmes de qualité :

Pb	Colonnes	Définition	Code
P1	in_reply_to_status_id	-Supprimer les valeurs de cette colonne avec la méthode drop.	drop('in_reply_to_status_id',axis=1,inplace=True)
P2	in_reply_to_user_id	- Supprimer les valeurs de cette colonne avec la méthode drop	drop('in_reply_to_user_id',axis=1,inplace=True)
P3	retweeted_status_id	- Supprimer les valeurs de cette colonne avec la méthode drop	drop('retweeted_status_id',axis=1,inplace=True
P4	retweeted_status_user_id	- Supprimer les valeurs de cette colonne avec la méthode drop	drop('retweeted_status_user_id',axis=1,inplace=True)
P5	retweeted_status_timestamp	- Supprimer les valeurs de cette colonne avec la méthode drop	drop('retweeted_status_timesstamp',axis=1,inplace=True)
P6	expanded_urls	- Supprimer les valeurs de cette colonne avec la méthode drop	drop('expanded_urls',axis=1,inplace=True)
P7	timestamp	Modifier grâce à la fonction to_datetime modifié le type d'object a dat_time de la colonne timestamp - Si les valeurs sont pour la plupart nulle supprimer simplement la colonne avec un drop	df1_copy['timestamp']=pd.to_datetime(df1_copy['timestamp'])

P8	Format des données manquantes	Pour chacunes des valeurs présentent avec un None utiliser replace pour mettre un Nan	df1.replace(None,Nan,inplace=True)
P9	Format de noms différents les uns des autres sur name de tab1	-Appliquer sur chaque valeurs de la colonne name la méthode title().	df1_copy['name'].str.title( )
P10	Format des noms dans P1,P2,P3	-Selectionner chaque colonne -Utiliser str pour le transformer en string -Utilser la fonction replace pour modifier '_' en ' '	df_2['p1'] =df_2['p1'].str.replace('_', ' ') df_2['p2'] =df_2['p2'].str.replace('_', ' ') df_2['p3'] =df_2['p3'].str.replace('_', ' ')
P11	Correction des données de rating_denominatio n et rating_numerator	-Afficher les lignes avec des dénominateurs different de 0 avec query -Selectionner le texte correspondant avec iloc -Lire la bonne note et modifier le rating_denominator et rating_numerator avec at	df1_copy.query("rating_denominator!=10") df1_copy['text'].iloc[313] df1_copy.at[313, 'rating_numerator'] = 13 df1_copy.at[313, 'rating_denominator'] = 10
Nettoyage des données désordonnées			
P12	Création d'une colonne rating	- Diviser la colonne rating_denominator et diviser par rating_numerator - Supprimer les colonnes rating_denominator, rating_numerator avec drop	df1_copy['rating']=(rating_denominator/rating_numerator) drop(['rating_denominator','rating_numerator],axis=1,inplace=True)
P13	Regroupement des stades des chiens doggo,flooper,pupper,puppo	-Utiliser melt pour passer d'un alignement vertical a horizontal des stades -Nettoyer le tableau résultant -Le reintegrer dans la base de données intiale **twitter-archive-enhanced.csv**	df_regroup= pd.melt(df1_copy,id_vars = 'tweet_id',value_name='dog_stage',value_vars =['doggo', 'floofer', 'pupper','puppo'])

P14	Créer une colonne prediction_breed_dog qui resume le race du chien obtenu à partir des prédictions	-Créer une colonne contenant des valeurs nulle dans la dataframe df2_copy avec la fonction repeat de numpy
		-Faire une itération sur les probabilités et les images et remplir le tableau au fur et à mesure
			-df1_copy['prediction_breed_dog'] = np.nan
			for i in range(len(image_predictions_df_clean)):
			if df2_copy['p1_dog'][i] == 1:
			df2_copy['prediction_breed_dog'][i] =df2_copy['p1'][i]
			elif df2_copy['p2_dog'][i] == 1:
			df2_copy['prediction_breed_dog'][i] = df2_copy['p2'][i]
			elif df2_copy['p3_dog'][i] == 1:
			df2_copy['prediction_breed_dog'][i] = df2_copy['p3'][i]

P15	Regrouper les tableau 1,2,3	-Regrouper grace a la methode merge les tab 1,2,3	df1_copy.merge(df_2.copy,how=inner,on=tweet_id) df1_copy.merge(df_3.copy,how=inner,on=tweet_id)

CONCLUSION

En définitive nous avons dans le cadre du projet weratedog pu ressortir dans grâce à l'évaluation visuelle et programmique les dans un premier temps les problèmes de qualité évaluée a environ 11 et les problèmes d'ordre évalue a 3.Grace a des méthodes pandas nous avons pu pallier à ces différents problèmes et conserver un seul fichier global twitter_archive_master.csv que nous allons analyser grâce à la visualisation.