Détectez les Bad Buzz grâce au Deep Learning

Hourdin Charlène - Octobre 2022

Agenda

Présentation du projet

Pré-traitement des données

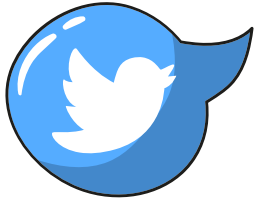
Présentation des trois approches

Choix du modèle et déploiement de l'API

PRÉSENTATION DU PROJET

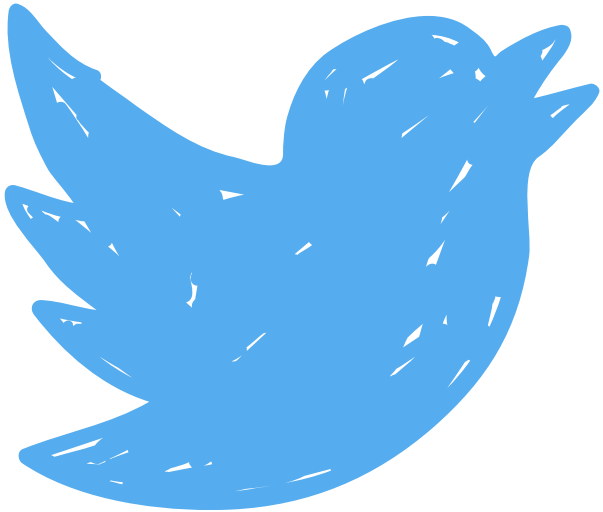
Appel à projet

01	**Le projet**	**Surveiller la réputation d'Air paradis** sur les réseaux sociaux
02	**L'objectif**	**Réaliser un prototype** permettant de prédire le sentiment associé à un tweet
03	**La mission**	**Mettre en production une API :** Le modèle envoie un tweet et récupère la prédiction de sentiment.
04	**La méthode**	Livrer un prototype fonctionnel du modèle



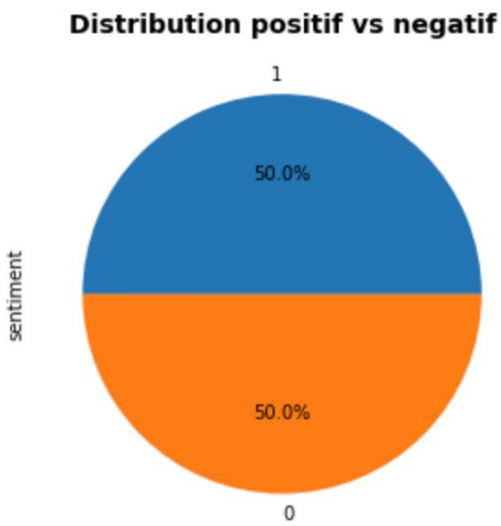
PRÉTRAITEMENT DES DONNÉES

Jeu de données



1 600 000 tweets

Classe equilibré

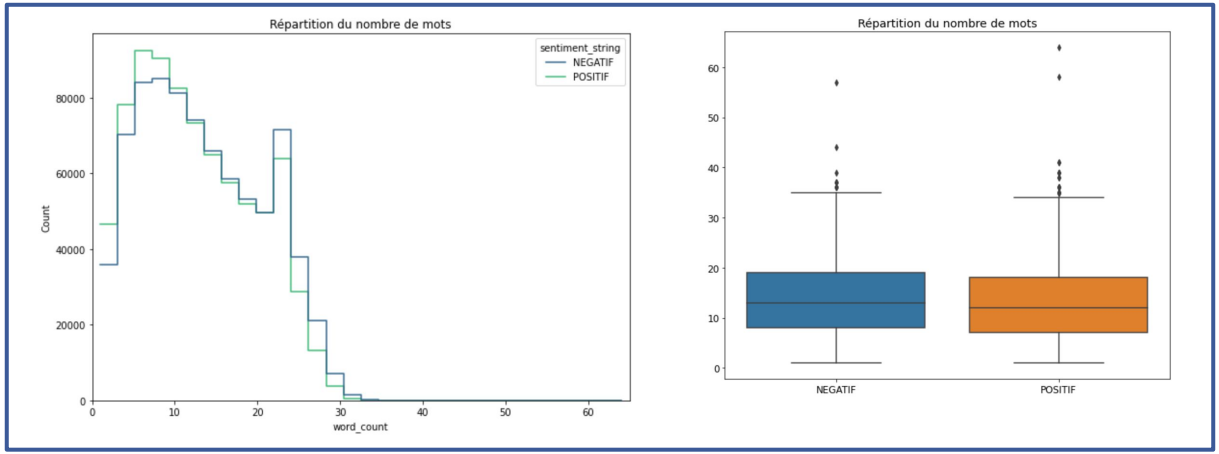


0 = négatif

1 = positif

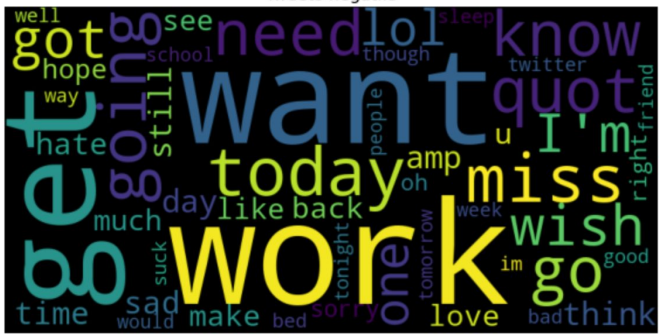
Exploration des données

13/14 mots en moyenne, plus de positif



Tweets positifs

Tweets negatifs



Prétraitement des données

Nettoyage préliminaire : Suppression des doublons sur le sous-ensemble [Utilisateur – Tweet] et des lignes vides

Standard

Normalisation

Mise en minuscule

Suppression de la ponctuations, des caractères spéciaux et des nombres

Traitements

Remplacement mentions, url et hashtags

Remplacement des contractions et des mots abrégés par les séquences de mots correspondantes.

Avancée

Lemmatization

Application de la racine lexicale des mots

Par exemple « **suis** » en **«être»**

Vectorisation

Transformation du texte en nombre.

PRÉSENTATION DES 3 APPROCHES

LES 3 APPROCHES

Apprentissage supervisée

Simple

BernoulliNB
SVC
Logistic regression

Ce sont des méthodes de **classification linéaire** qui apprennent la probabilité qu'un échantillon appartiennent à une certaine classe.

- BernoulliNB : Modèle bayésien naïf
- SVC : Machine à vecteur de support
- Régression logistique : Modèle linéaire

- Simple et rapide à exécuter
- Le modèle LR, offre de bonne performance

Avancé

RNN
LSTM (GlOVE et Word2vec)
BiLSTM (GlOVE et Word2vec)

Modèles de **réseaux de neurones profond,** Ils sont plus coûteux en coût et en temps d'apprentissage. Une **couche LSTM** est une couche avec une mémoire interne appelée **cellule**. Les cellules peuvent maintenir l'état selon les besoins. Cette unité contient des valeurs que le réseau peut contrôler en fonction de la situation.

- Nécessite des profils technique avancée
- Plus coûteux en temps d'apprentissage et en coût
- Meilleur résultats avec une étape de word embedding

BERT

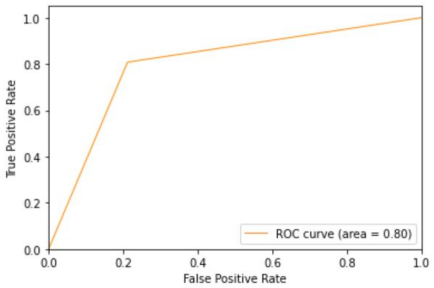
Modèle pré-entraîné HuggingFace

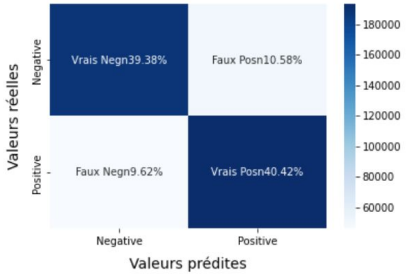
C'est un modèle faisant partie de la classe des **Transformers** de la bibliothèque **HuggingFace**. Ce modèle offre de **très bonnes performances** dans le traitement automatique de la langue.

- Nécessite des ressources GPU importante

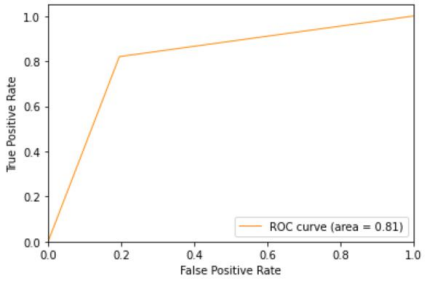
Modèle simple

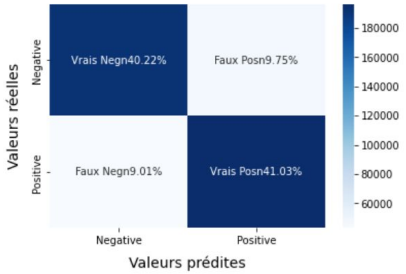
Bernoulli NB



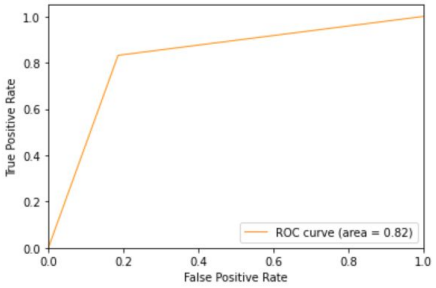


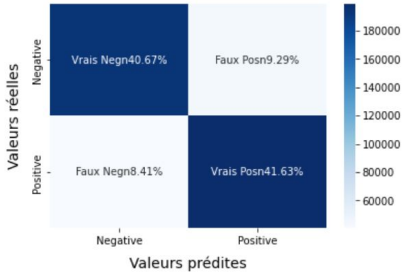
SVC





Logistic regression





	f1 score	fbeta score	accuracy	recall	AUC	Training time	Predict time
BernoulliNB	0.800074	0.795518	0.798008	0.807785	0.798001	0.580399	0.209882
LinearSVC	0.813961	0.810375	0.812449	0.820009	0.812444	24.621607	0.050732
LogisticRegression	0.824657	0.820348	0.822985	0.831939	0.822979	184.211845	0.051044

Modèles avancé

Paramètres

VOCAB_SIZE:	50000
MAX_LEN:	36
EMBEDDING_DIM:	256
DROP_OUT:	0.5
OPTIM_LR:	0.001
REGUL_LR:	0.001
NUM_EPOCHS:	50
BATCH_SIZE:	250

Modèle de base *à améliorer*
RNN (Baseline)

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 36, 256)	12800000
spatial_dropout1d (SpatialD ropout1D)	(None, 36, 256)	0
simple_rnn (SimpleRNN)	(None, 128)	49280
dropout (Dropout)	(None, 128)	0
dense (Dense)	(None, 64)	8256
dropout_1 (Dropout)	(None, 64)	0
dense_1 (Dense)	(None, 1)	65

Total params: 12,857,601
Trainable params: 12,857,601
Non-trainable params: 0

Lecture gauche / droite et droite gauche

bidirectional (Bidirectiona	(None, 256)	439296
lstm_1 (LSTM)	(None, 128)	219648

LSTM avec incorporation de mot
(word embedding pré-entraîné)

Word2vec

embedding_2 (Embedding)	(None, 36, 300)	51211200

GlOVE

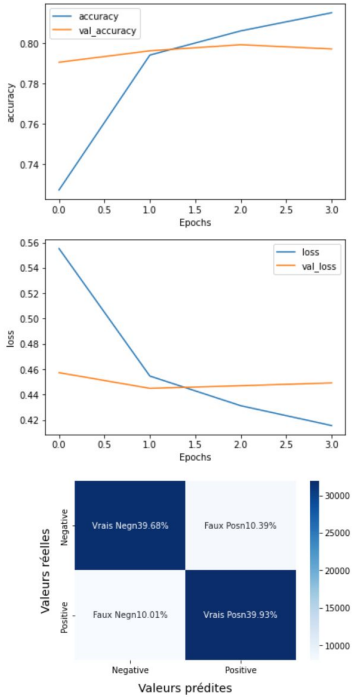
embedding (Embedding)	(None, 36, 200)	34140800

Differents modèle : RNN -> LSTM -> BiLSTM

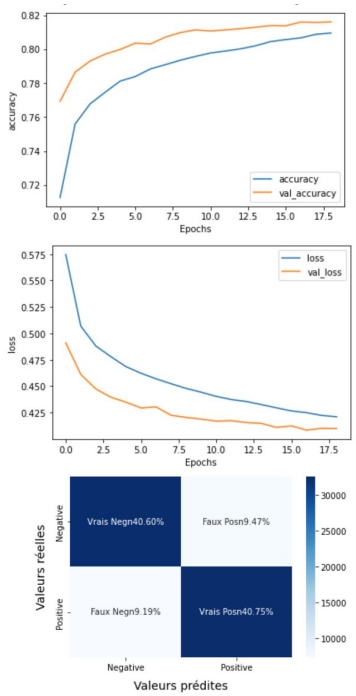
Differents embedding : From scratch -> Word2vec -> GlOVE

Performance des modèles avancé

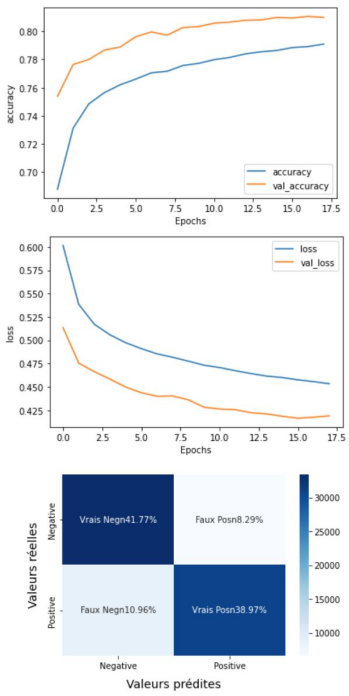
RNN



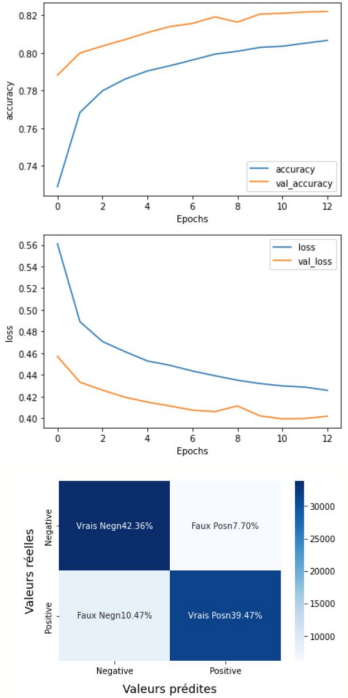
Word2vec LSTM



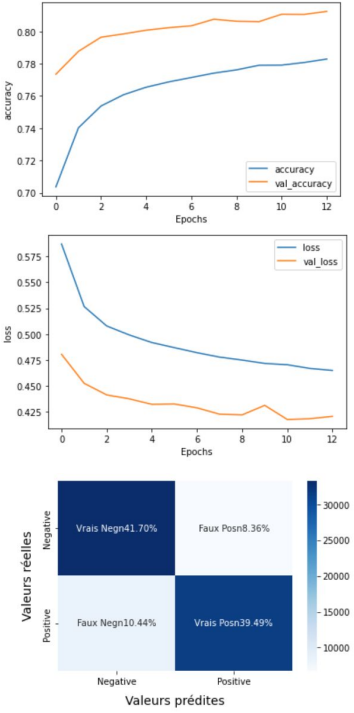
Word2vec BiLSTM



Glove LSTM



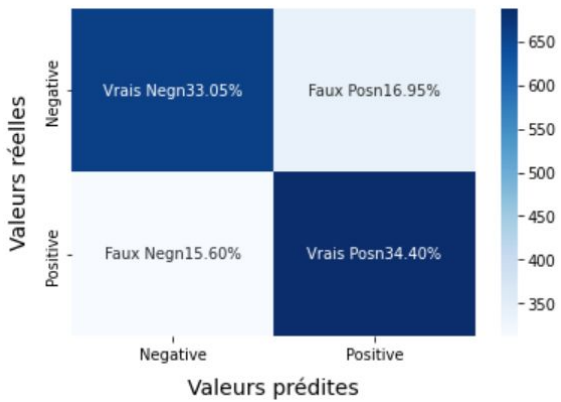
Glove BiLSTM

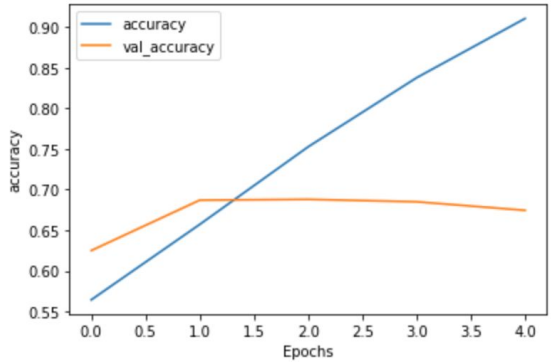


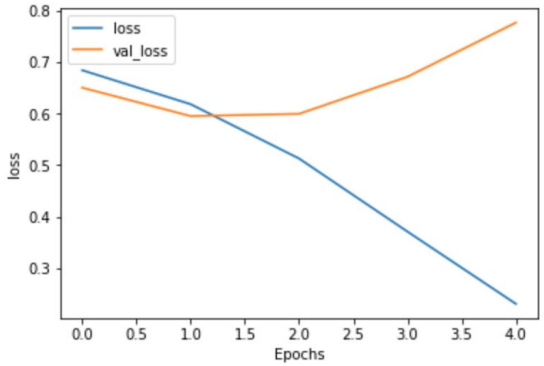
BERT

Layer (type)	Output Shape	Param #
bert (TFBertMainLayer)	multiple	109482240
dropout_75 (Dropout)	multiple	0
classifier (Dense)	multiple	1538

Total params: 109,483,778
Trainable params: 109,483,778
Non-trainable params: 0







COMPARAISON DES MODÈLES

	f1 score	fbeta score	accuracy	recall	AUC	Training time	Predict time
Base_model_RNN_SW	0.775628	0.778244	0.777162	0.771308	0.856695	194.007068	1.950056
Base_model_RNN_lem	0.796533	0.793437	0.795462	0.801747	0.876793	238.396502	2.794122
word2vec_LSTM_SW	0.784376	0.784511	0.784713	0.784150	0.867489	328.366219	1.745733
word2vec_LSTM_lem	0.801054	0.819317	0.808425	0.772360	0.891052	199.644173	1.743551
word2vec_BiLSTM_SW	0.768795	0.791382	0.779575	0.733886	0.865215	641.373311	2.769052
word2vec_BiLSTM_lem	0.806589	0.815326	0.810225	0.792435	0.892453	628.708532	3.403104
GlOVE_LSTM_SW	0.786579	0.798617	0.792075	0.767303	0.875362	268.527660	1.798171
GlOVE_LSTM_lem	0.812716	0.826748	0.818100	0.790358	0.900399	208.350103	1.574794
GlOVE_BiLSTM_SW	0.784605	0.786634	0.785800	0.781246	0.867165	264.062463	2.424098
GlOVE_BiLSTM_lem	0.795422	0.823063	0.806512	0.753260	0.891043	291.084454	2.720933
bert	0.678836	0.673453	0.674500	0.688000	0.753178	1063.036481	23.155614

CHOIX DU MODÈLE ET DÉPLOIEMENT DE L'API

Prédiction

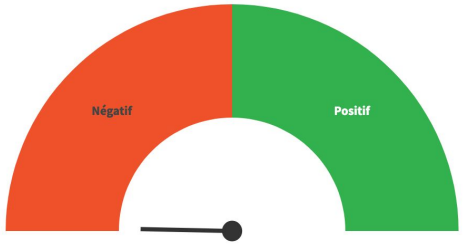
Goodmorning twitterville!!! What oh what to do today?! I'm thinkin it's the day I make an official life plan... So many goals!!!!

Bonjour twitterville !!! Que faire aujourd'hui ? ! Je pense que c'est le jour où je fais un plan de vie officiel... Tellement d'objectifs !!!!

Score:
83

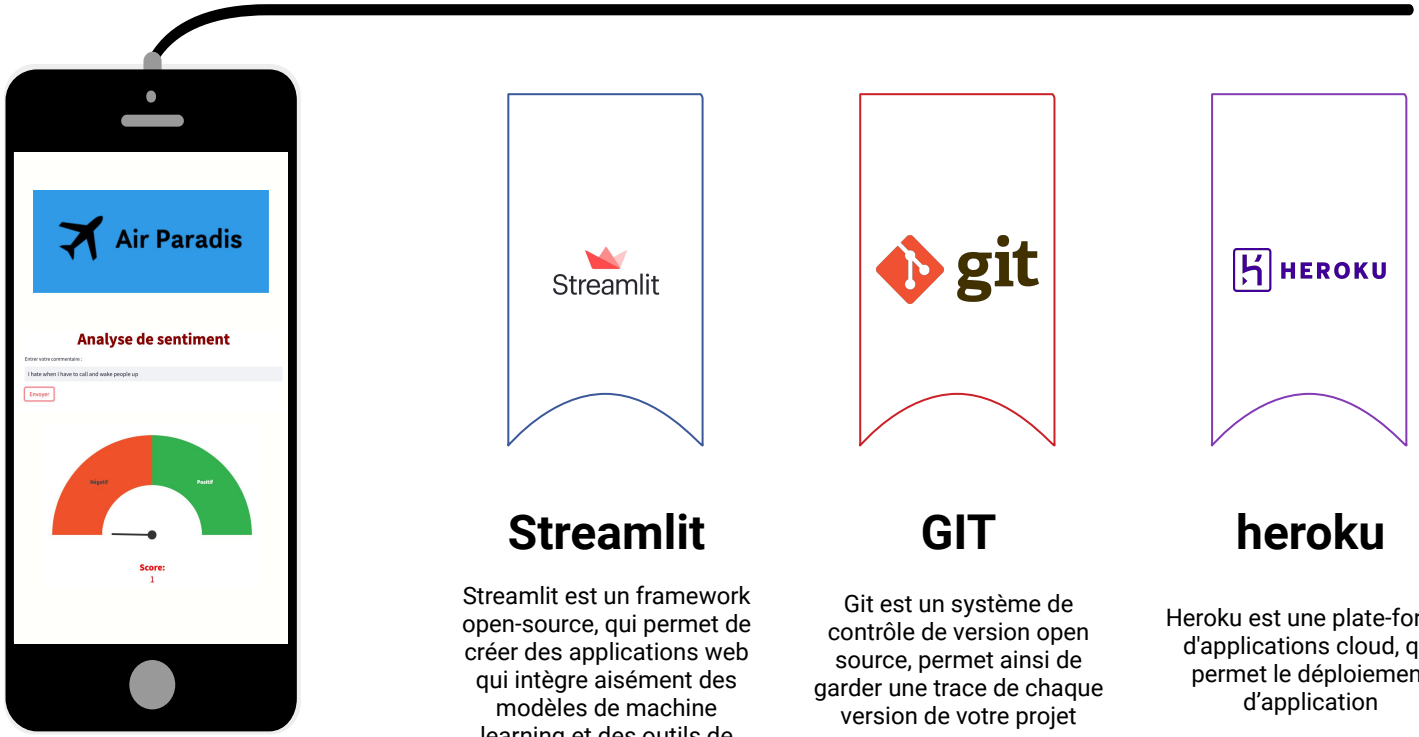
I hate when I have to call and wake people up

Je déteste quand je dois appeler et réveiller les gens



Score:
1

Déploiement du modèle



Streamlit

Streamlit

Streamlit est un framework open-source, qui permet de créer des applications web qui intègre aisément des modèles de machine learning et des outils de visualisation de données.

git

GIT

Git est un système de contrôle de version open source, permet ainsi de garder une trace de chaque version de votre projet

HEROKU

heroku

Heroku est une plate-forme d'applications cloud, qui permet le déploiement d'application

Conclusion

Le meilleur modèle fait certaine **erreur de prédiction** qui peuvent être liées à des erreurs de labellisation, un mix de joie et de peine dans le commentaire, une langue autre que l'anglais, une erreur du modèle

Les **performances** du modèle dépendent principalement :
- de la qualité initiale des données ;
- du prétraitement effectué.

Le **modèle simple** est plus rapide à exécuter et offre de bonnes performances

Le **modèle avancé** obtient de meilleurs résultats, mais cela nécessite des connaissances en Deep Learning et en programmation, ainsi qu'un certain temps de modélisation, d'entraînement et de tests.

*Le **choix d'une approche** dépendra des besoins exprimés par les équipes de l'entreprise et des ressources disponibles*