Université de Lille

ILIS

RAPPORT DE PROJET

Classification de conseils médicaux

Enseignant : Naima Oubenali

01/06/2024

Nene sidibe BAKARY
Imane ELMISSAOUI
Ezéchiel DJOHI

APERÇU DU PROJET

Classification des conseils médicaux trouvés sur internet

Absence de conseil

faible conseil

Fort conseil

Objectif

Développer un modèle qui évalue la qualité des conseils trouvés en ligne

ÉTAPES

Collecte de données et preprocessing

Fine tuning du modèle Bert

Evaluation et validation du modèle

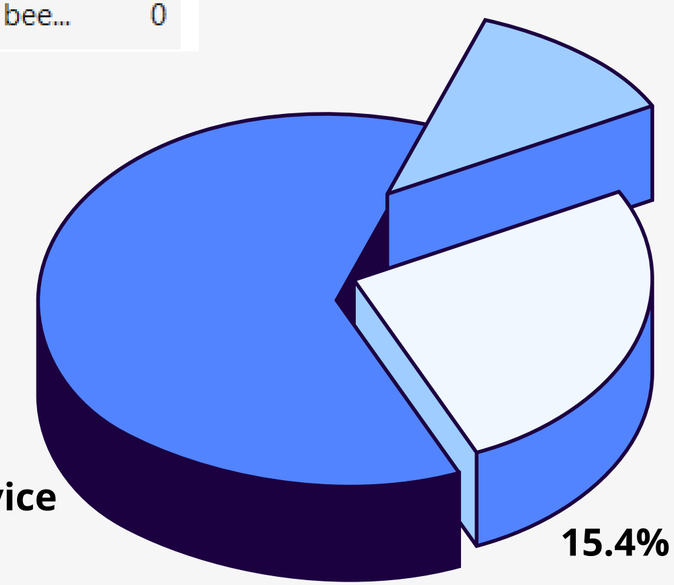
Développement d’une application web de test

DONNÉES UTILISÉES

Hugging Face

	instruction	output	input	label
0	Question: is this a 2) strong advice, 1) weak ...	This is no advice	As we have previously shown an additional effe...	0
1	Question: is this a 2) strong advice, 1) weak ...	This is no advice	Furthermore, gut microbiota analysis in mice t...	0
2	Question: is this a 2) strong advice, 1) weak ...	This is no advice	Further research is recommended that may be he...	0
3	Question: is this a 2) strong advice, 1) weak ...	This is no advice	Further study will be necessary to test if pos...	0
4	Question: is this a 2) strong advice, 1) weak ...	This is no advice	On the other hand, rheumatoid factor and Epste...	0
5	Question: is this a 2) strong advice, 1) weak ...	This is no advice	Interestingly, within this context, it has bee...	0

9,9% strong advice



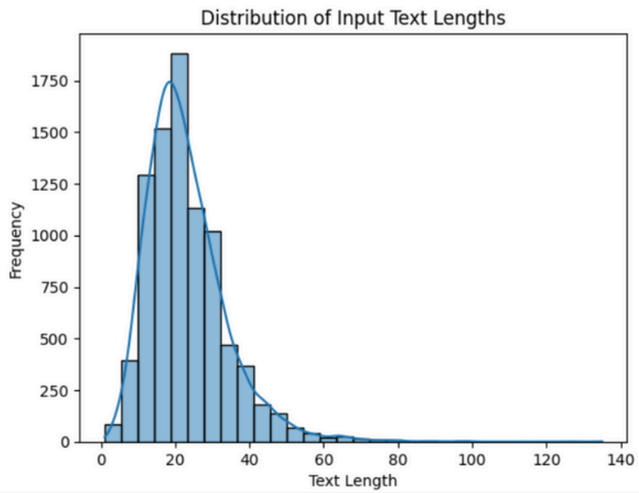
74,7% no advice

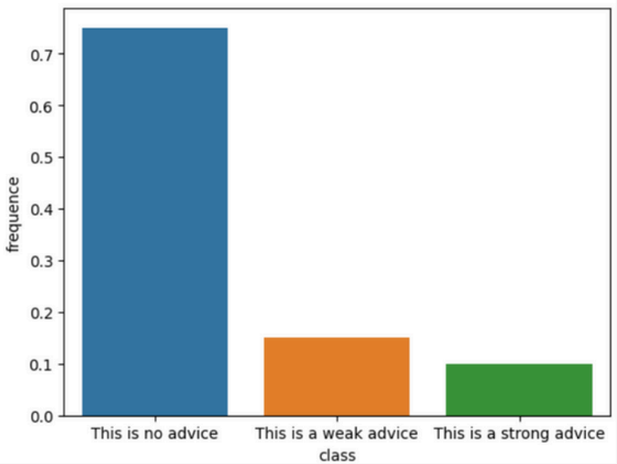
15.4% weak advice

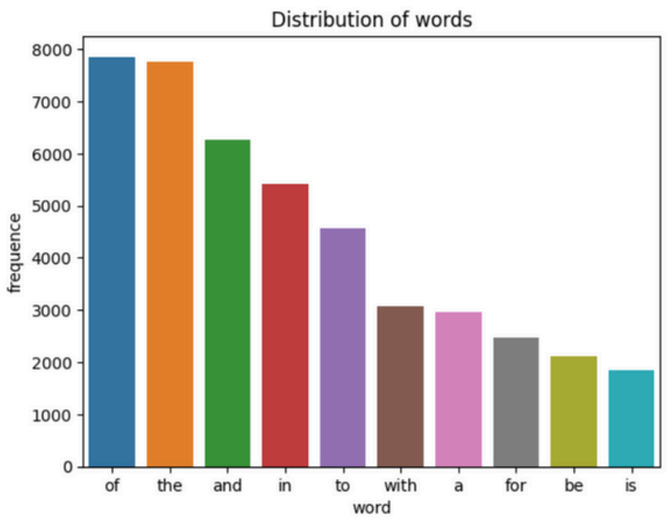
Etapes du Prétraitement

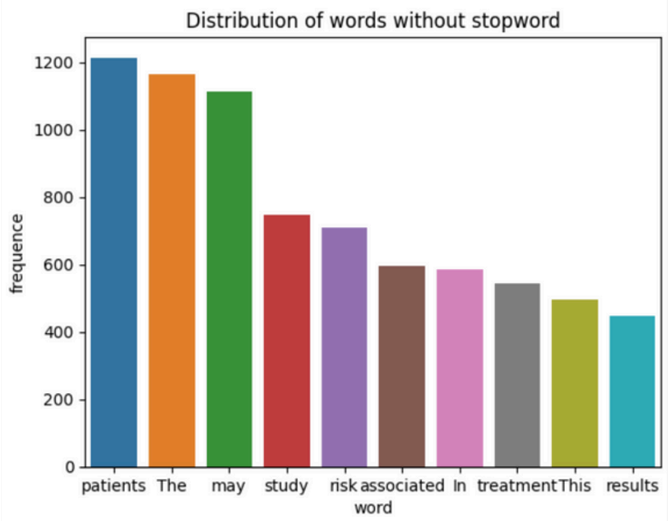
- **Suppression des liens**
- **Suppression des emojis**
- **Remplacement de certains caractères**

ANALYSE EXPLORATOIRE DES DONNÉES :









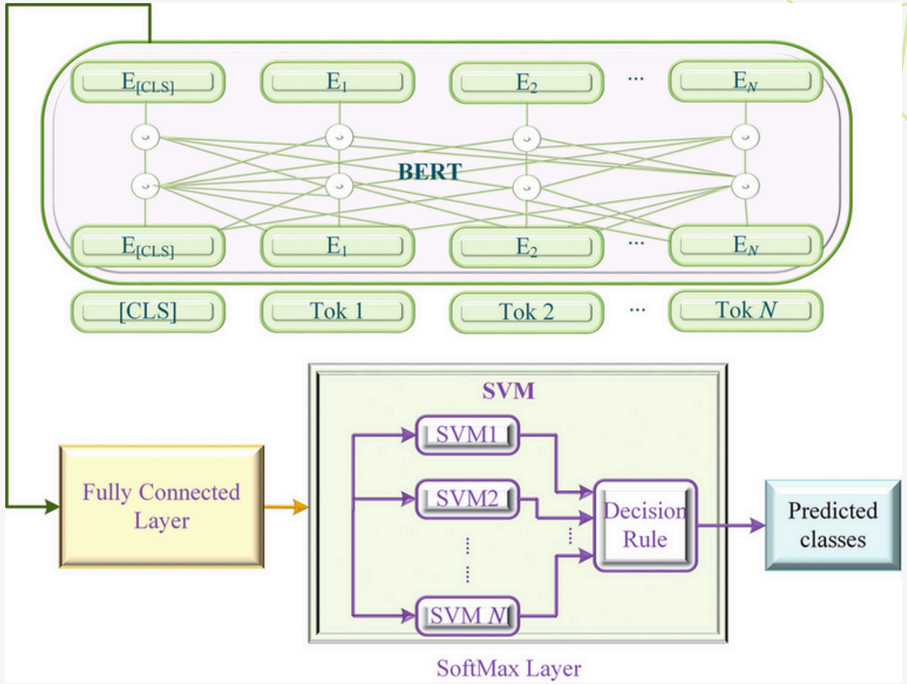
Données entraînés :
medical_meadow_health_advice
de *Hugging face*

Indicateurs	Valeurs
nb_ligne	8676
nb_class	3
nb_token_total	199711
nb__token_unique	26111
short_token_len	1
long_token_len	55
long_text_len	135
short_text_len	1

MODÈLE DE CLASSIFICATION :

Il existe différents types de classification multiclass

- **Support Vector Machines (SVM)**
- **Long Short-Term Memory (LSTM)**
- **Random Forest**
- **Logistic Regression**
- **Transformers :Bert,Roberta,etc**



Architecture de notre modèle de classification

Implémentation du modèle Hybride Bert-SVM(Transfer-Learning)

Prétraitement des données

Tokénization des données textuelles à l'aide d'un tokenizer BERT

Extraction des embeddings de Bert

Utilisation d'un modèle BERT pré-entraîné pour extraire les embeddings.

Entraînement d'un classificateur SVM

Utilisation de ces embeddings comme caractéristiques d'entrée pour le classificateur SVM.

Évaluation du modèle

Test du modèle sur un ensemble de données de test et évaluez ses performances.

ENVIRONNEMENT

3.8 +

NVIDIA
CUDA

12.4

Projet NLP /

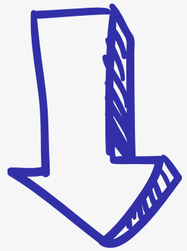
- Data /
  - medical_meadow_health_advice.json
  - Tain.csv
  - Test.csv
  - Validation.csv
- Documentation/
  - Models_description.txt
  - SVM/
- Models/
  - BERT/
  - SVM/
- Function_utiles/
  - exploration.py
  - preprocessing.py
- App.py
- example.py
- README.md

FINE TUNING DU MODELE BERT

Hugging Face

Entrainement avec le modèle Bert_base_uncased

google-bert/**bert-base-uncased**



A Nécessité des ressources de calcul très importantes

Modèle prajjwal1/bert-tiny de Hugging face

prajjwal1/**bert-tiny**

Choix de ce modèle qui a pu tourner sur le GPU disponible

RÉSULTATS

EVALUATION DU MODELE :

Test set classification report:

	precision	recall	f1-score	support
0	0.80	0.98	0.88	644
1	0.67	0.32	0.44	145
2	0.75	0.08	0.14	79
accuracy			0.79	868
macro avg	0.74	0.46	0.48	868
weighted avg	0.77	0.79	0.74	868

Validation set classification report:

	precision	recall	f1-score	support
0	0.78	0.98	0.86	629
1	0.64	0.29	0.40	146
2	0.70	0.08	0.14	93
accuracy			0.76	868
macro avg	0.70	0.45	0.47	868
weighted avg	0.74	0.76	0.71	868

nepochs=5

Test set classification report:

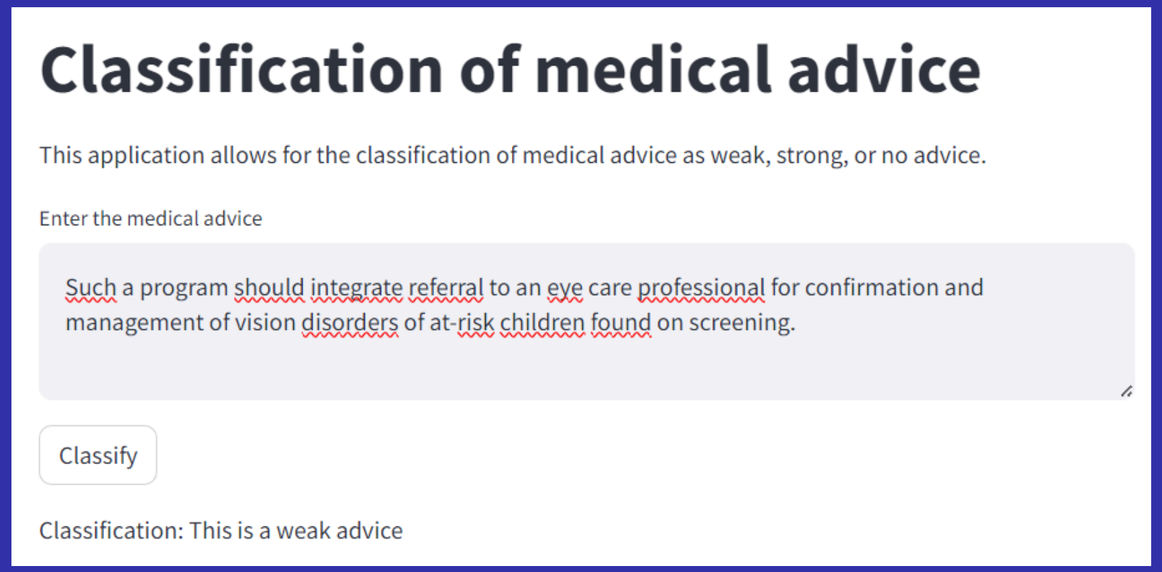
	precision	recall	f1-score	support
0	0.81	0.96	0.88	644
1	0.68	0.33	0.44	145
2	0.44	0.19	0.27	79
accuracy			0.78	868
macro avg	0.64	0.49	0.53	868
weighted avg	0.75	0.78	0.75	868

Validation set classification report:

	precision	recall	f1-score	support
0	0.81	0.96	0.88	629
1	0.64	0.35	0.45	146
2	0.55	0.24	0.33	93
accuracy			0.78	868
macro avg	0.67	0.52	0.55	868
weighted avg	0.75	0.78	0.75	868

nepochs=20

DÉPLOIEMENT DU MODÈLE :



Streamlit de python

Streamlit

DISCUSSION

- Performance du modèle à améliorer
- Distribution déséquilibrée des classes des labels

Nombre d’épochs, batch_size, et autres paramètres à changer pour optimizer le modèle

Re-sampling

Répétition des exemples existants (dans les classes minoritaires) ou l'utilisation de techniques comme SMOTE (Synthetic Minority Over-sampling Technique)

Class Weights

Ajustement des poids des classes dans la fonction de perte du modèle

MERCI !