M2ISE M1Info

Université Paris 8 - Vincennes à Saint-Denis

Master Informatique des Systèmes Embarqués

Apprentissage automatique

Analyse de sentiment

Nour ZEROUALI

Lisa Medrouk – Université Paris VIII

Analyse de sentiment et réseaux sociaux

Les réseaux sociaux c'est une puissante source de communication entre les gens pour partager leurs sentiments sous la forme d'opinion et de points de vue sur n'importe quel sujet ou article, ce qui entraîne une énorme quantité d'informations non structurées. Pour analyser ces sentiments, diverses approches d'apprentissage automatique et basées sur le traitement du langage naturel ont été utilisées dans le passé. Cependant, les méthodes d'apprentissage profond deviennent très populaires en raison de leurs performances élevées ces derniers temps j'utilise colab notebook.

L'analyse des sentiments est un processus consistant à identifier et à classer par ordinateur les opinions exprimées dans un morceau de texte, en particulier pour déterminer si l'attitude de l'écrivain envers un sujet, un produit, etc. est positive, négative ou neutre.
Pour mon apprentissage automatique je crée un réseau neuronal récurrent. Mon objectif était de différencier uniquement les tweets positifs et négatifs. Après cela, je filtre les tweets afin qu'il ne reste que des textes et des mots valides. Ensuite, je définis le nombre de fonctionnalités max à 2000 et j'utilise Tokenizer pour vectoriser et convertir le texte en séquences afin que le réseau puisse le traiter en entrée. Ensuite, je compose le réseau LSTM avec Les variables EMBED_DIM, LST_OUT, BATCH_SIZE, DROUPOUT_X qui sont des hyperparamètres, leurs valeurs sont intuitives, elle sont utilisées pour obtenir de bons résultats. J'utilise également softmax comme fonction d'activation. Le réseau utilise une entropie croisée catégorielle, et je pense que softmax est la bonne méthode d'activation pour cela.

Je passe ensuite a la déclaration du dataset d'entraînement et celui du test.

	id	candidate	candidate_confidence	relevant_yn	relevant_yn_confidence	sentiment	sentiment_confidence	subject_ma
0	1	No candidate mentioned	1.0	yes	1.0	Neutral	0.6578	None of the above
1	2	Scott Walker	1.0	yes	1.0	Positive	0.6333	None of the above
2	3	No candidate mentioned	1.0	yes	1.0	Neutral	0.6629	None of the above
3	4	No candidate mentioned	1.0	yes	1.0	Positive	1.0000	None of the above
4	5	Donald Trump	1.0	yes	1.0	Positive	0.7045	None of the above

FIGURE 1 – Dataset.head

Ici on recupere les deux colonne text et sentiments, dans les sentiments on prend traite que les sentiments positifs et negatifs on ignorant les sentiments neutres.

Le model est construit d'une couche LSTM avec 196 unités de mémoire (neurones intelligents). Enfin, comme il s'agit d'un problème de classification, nous utilisons une couche de sortie dense avec deux neurones et une fonction d'activation softmax pour faire des prédictions 0 ou 1 pour les deux classes negatif et postif.

Parce qu'il s'agit d'un problème de classification binaire, la perte de journal est utilisée comme fonction de perte (binary_crossentropy dans Keras). L'algorithme d'optimisation ADAM efficace est utilisé.

Model ¶

```
embed_dim = 128
lstm_out = 196

model = Sequential()
model.add(Embedding(max_fatures, embed_dim,input_length = X.shape[1]))
model.add(SpatialDropout1D(0.4))
model.add(LSTM(lstm_out, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(2,activation='softmax'))
model.compile(optimizer = 'adam',loss = 'binary_crossentropy', metrics=['accuracy'])
print(model.summary())
```

```
Model: "sequential_13"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_12 (Embedding)     (None, 28, 128)           256000
_________________________________________________________________
spatial_dropout1d_8 (Spatial (None, 28, 128)           0
_________________________________________________________________
lstm_12 (LSTM)               (None, 196)               254800
_________________________________________________________________
dense_12 (Dense)             (None, 2)                 394
=================================================================
Total params: 511,194
Trainable params: 511,194
Non-trainable params: 0
_________________________________________________________________
None
```

FIGURE 2 – Le model

Il aurait fallut exécuter plus que 7 époques, mais je devrais attendre indéfiniment pour kaggle, le nombre d'époques est donc fixer a 7 pour l'instant. Dans le bute de mesurer le score et l'accuracy on extrait un ensemble de validation.

Enfin, on mesure le nombre de suppositions correctes. On remarque la détection des tweets négatifs par le réseau est parfaite, mais décider si c'est positif ne l'est pas vraiment. Je pense que c'est a cause du dataset car l'ensemble d'entraînement positif est beaucoup plus petit que le négatif, il faudra donc agrandir l'ensemble des tweets positifs pour améliorer les résultat et avoir un dataset d'entraînement équilibré.

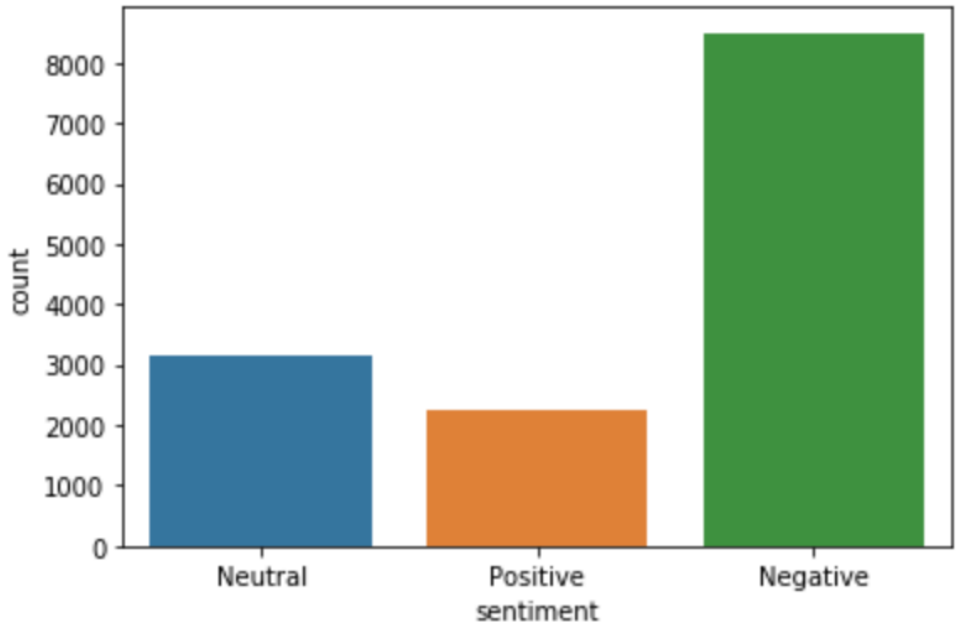

FIGURE 3 – L'ensemble de dataset