BAD BUZZ

AIR PARADIS : DÉTECTER UN BAD BUZZ GRÂCE À LE DEEP LEARNING

Contexte

Enjeux

« Air paradis » cherche à mettre en place un outil de detection de « **bad buzz** »

Pour cela, l'equipe IA doit:

- **Detecter** les tweets « negatifs »

Objectifs

Nous allons chercher à tester differentes approaches :

- **Modele sur mesure simples**: déploiement d'une API sur un service Cloud
- **Modeles sur etagere**
- **Modele sur mesure avances :**
    - **Racinisation:**
        - Steamming
        - Lemmatization
    - **Word emmbeding:**
        - Glove
        - Word2vec
        - FastText
    - **Modele de Deep learning:**
        - LSTM
        - BERT
- **Deploiement en production du meilleur modele**

Méthode

Nous utilisons jeu de données **Sentiment140 dataset**

- Tweets: donnees de **1.6 million tweets**.

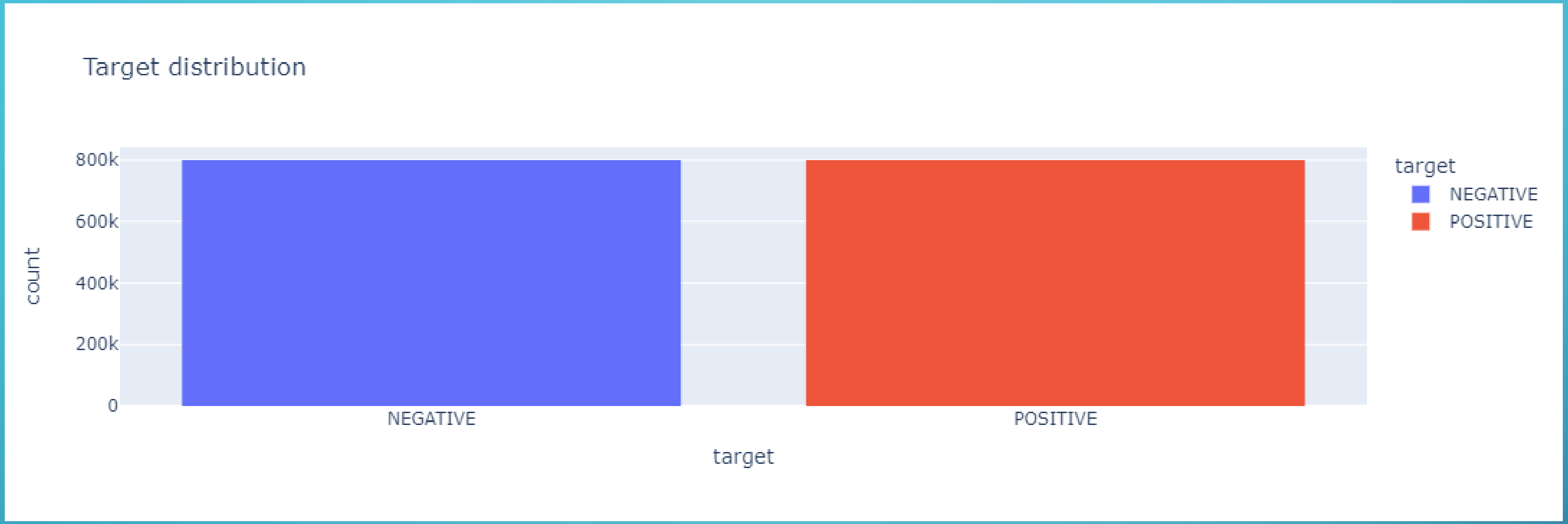
Nous sommes face a une probleme de Natural Language Processing (NLP):

  - Tester different models pour predire le sentiment du tweets

Nous allons:

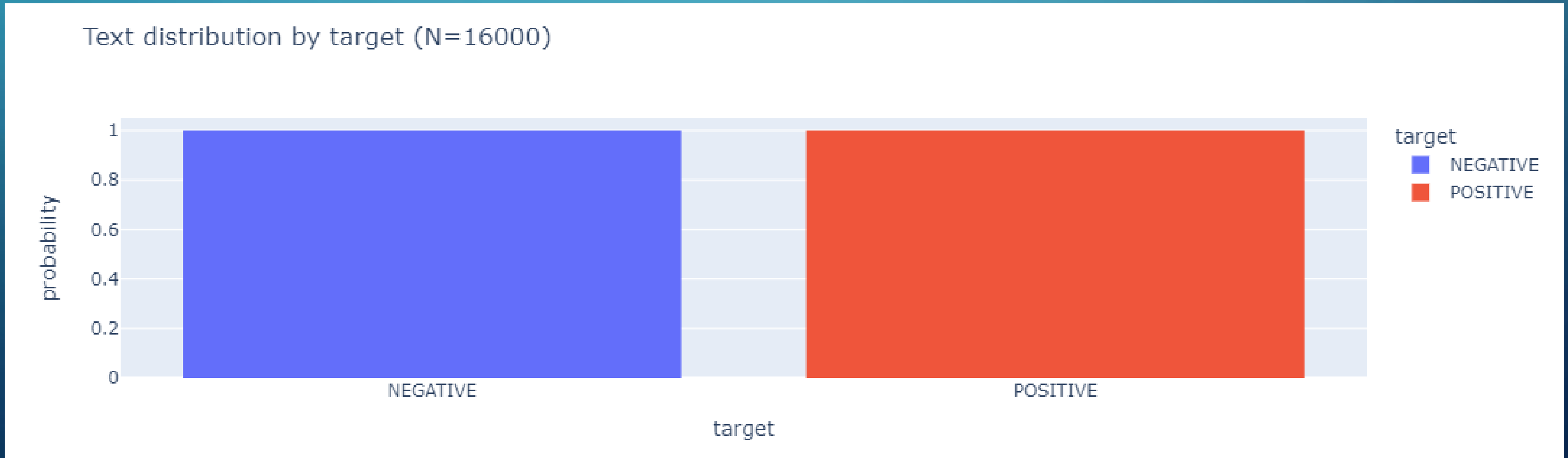
- Lancer un EDA
- Pre-traitement du text
- Extraires les **features:** « **Tokens** » => groupe de mots reduits a leur forme la plus simple
- Tester different approaches

Natural language processing (NLP): Exploratory data analysis

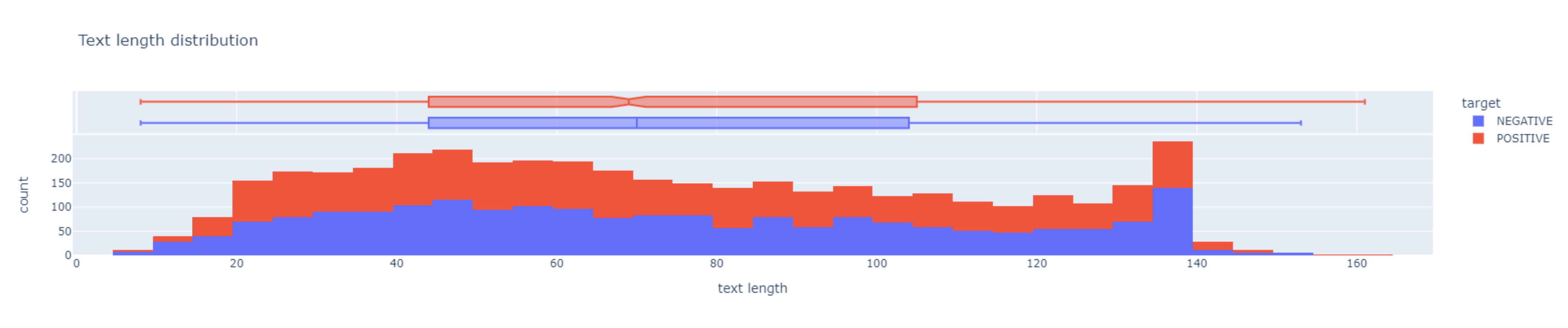


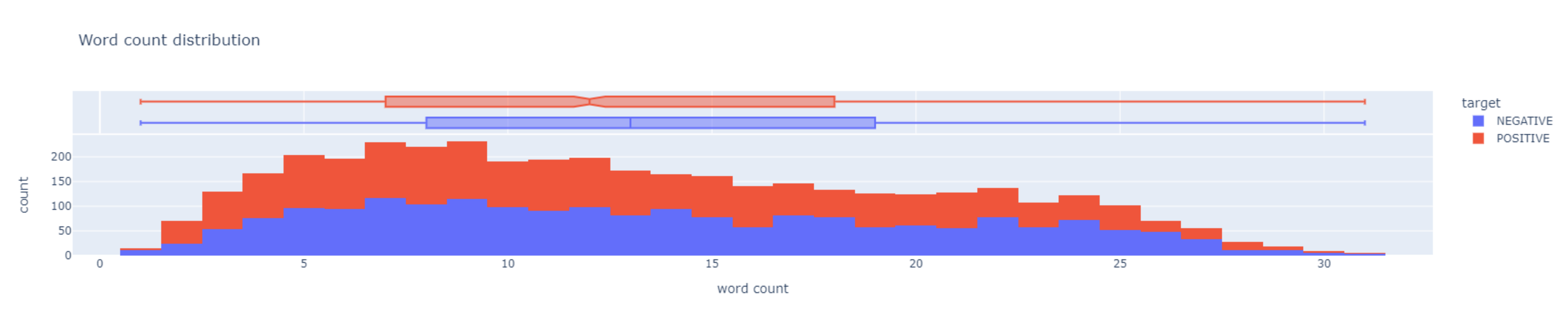
Statistiques :

- Sampling dataset a 16000 commentaires



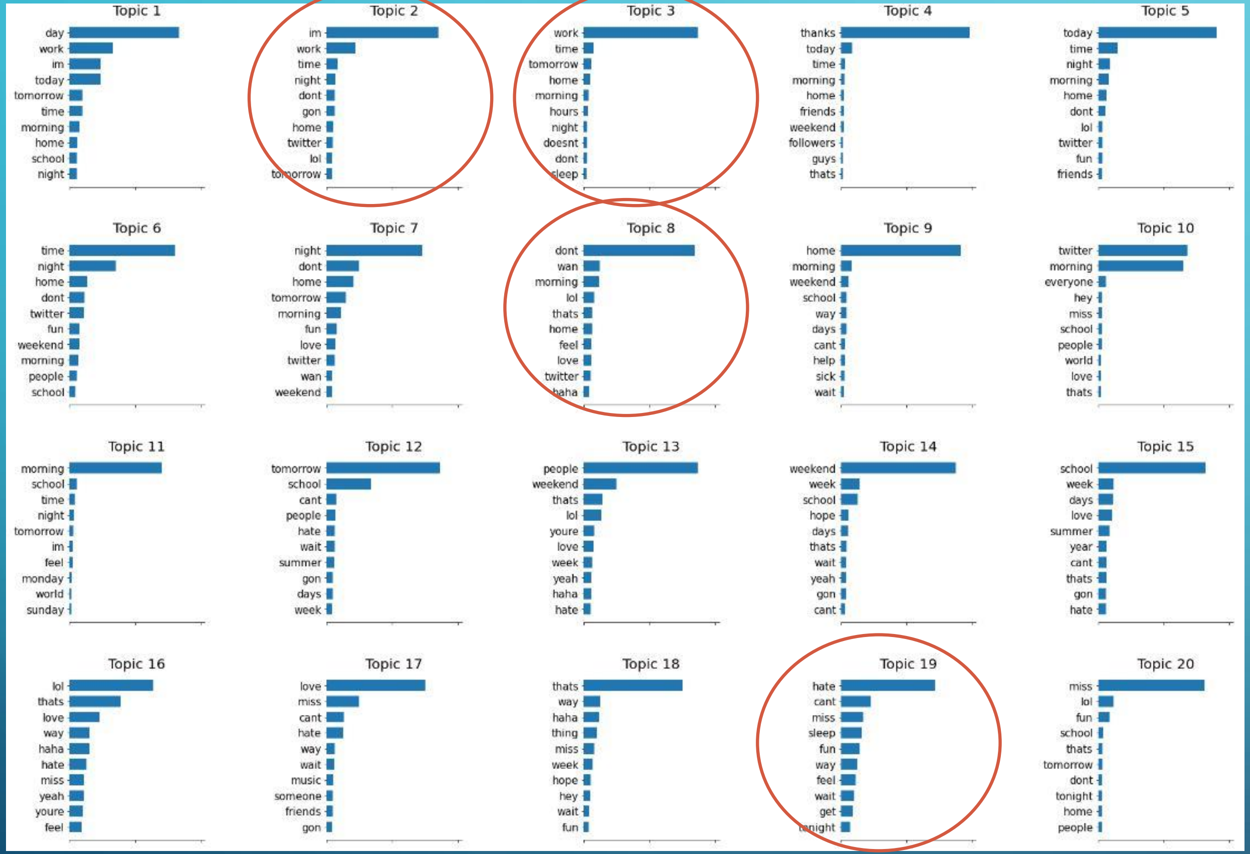
Exploratory data analysis: Text

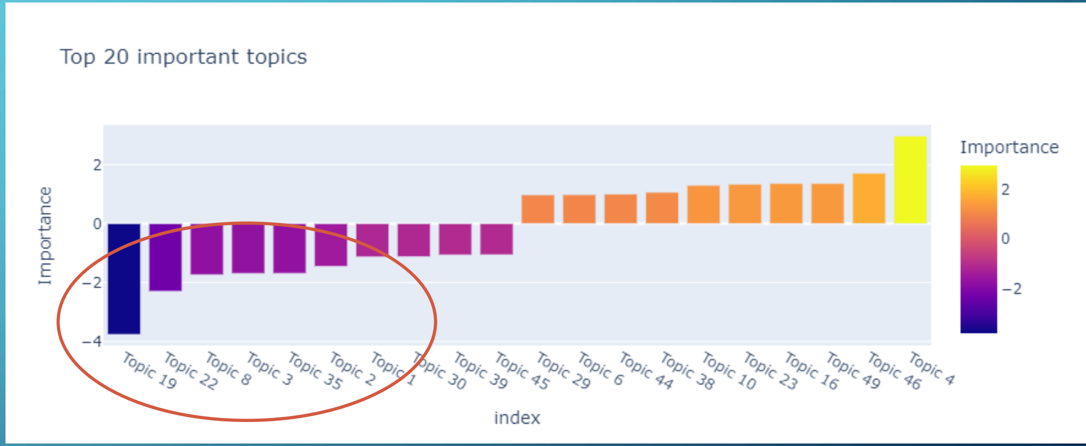




Exploratory data analysis: Text topics

LSA Topics / n_components = 50





Azure Cognitive Services: Text Analytics API

Model: sentiment analysis

```
Sentence: Not only the service is excellent, although the quality of the champagne should be improved, but also the compact but spacious space in first class.
Sentence sentiment: positive
Sentence score:
Positive=0.99
Neutral=0.00
Negative=0.01

......'positive' target 'service'
......Target score:
......Positive=1.00
......Negative=0.00

......'positive' assessment 'excellent'
......Assessment score:
......Positive=1.00
......Negative=0.00

......'positive' target 'space'
......Target score:
......Positive=0.99
......Negative=0.01

......'positive' assessment 'compact'
......Assessment score:
......Positive=0.99
......Negative=0.01

......'positive' assessment 'spacious'
......Assessment score:
......Positive=1.00
......Negative=0.00

......'positive' assessment 'first class'
......Assessment score:
......Positive=0.99
......Negative=0.01
```

```
Document Sentiment: positive
Overall scores: positive=0.82; neutral=0.11; negative=0.07

Sentence: this airline blows your mind.
Sentence sentiment: positive
Sentence score:
Positive=0.65
Neutral=0.21
Negative=0.13

......'positive' target 'airline'
......Target score:
......Positive=0.91
......Negative=0.09

......'positive' assessment 'blows your mind'
......Assessment score:
......Positive=0.91
......Negative=0.09
```

Modèle sur mesure simple: Logistic regression

Text pre-processing:

- **Lemmatization:** NLTK
- **Vectorization:** Tf-Idf

Dimension reduction: LSA

Model: Logistic regression

	precision	recall	f1-score	support
NEGATIVE	0.70	0.44	0.54	643
POSITIVE	0.59	0.81	0.68	637
accuracy			0.62	1280
macro avg	0.64	0.62	0.61	1280
weighted avg	0.64	0.62	0.61	1280

ROC AUC score : 0.624

Average Precision score : 0.571

Confusion matrix

Modèle sur mesure avancé : Neural Networks avec Keras

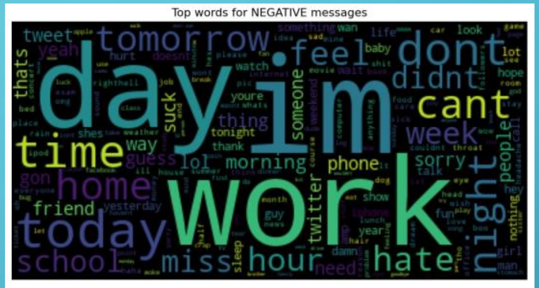
Model: LSTM DNN

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_3 (Embedding)     (None, 300, 300)          1967400

 bidirectional (Bidirectiona  (None, 300, 42)          54096
 l)

 global_max_pooling1d (Globa  (None, 42)               0
 lMaxPooling1D)

 batch_normalization (BatchN  (None, 42)               168
 ormalization)

 dropout (Dropout)           (None, 42)                0

 dense (Dense)               (None, 21)                903

 dropout_1 (Dropout)         (None, 21)                0

 dense_1 (Dense)             (None, 21)                462

 dropout_2 (Dropout)         (None, 21)                0

 dense_2 (Dense)             (None, 1)                 22

=================================================================
Total params: 2,023,051
Trainable params: 2,022,967
Non-trainable params: 84
```

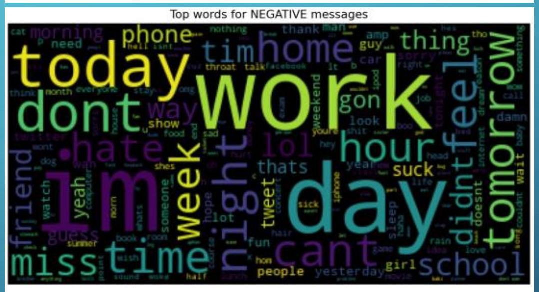
On a teste cette DNN avec 2 approach differentes de racinitation:

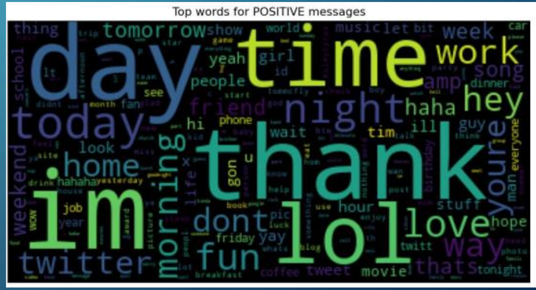
- Lemmatization
- Steamming

Lem



Stm





On a teste cette DNN avec 3 approach differentes de Word Enbedding:

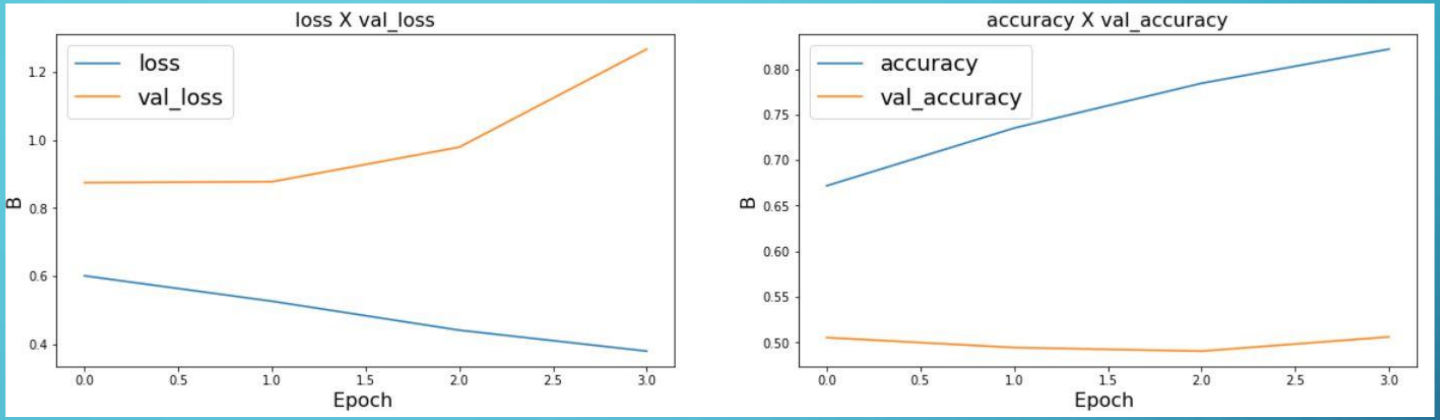
- Glove
- Word2Vec
- FastText

Modèle sur mesure avancé : Neural Networks avec Keras

Text pre-processing:

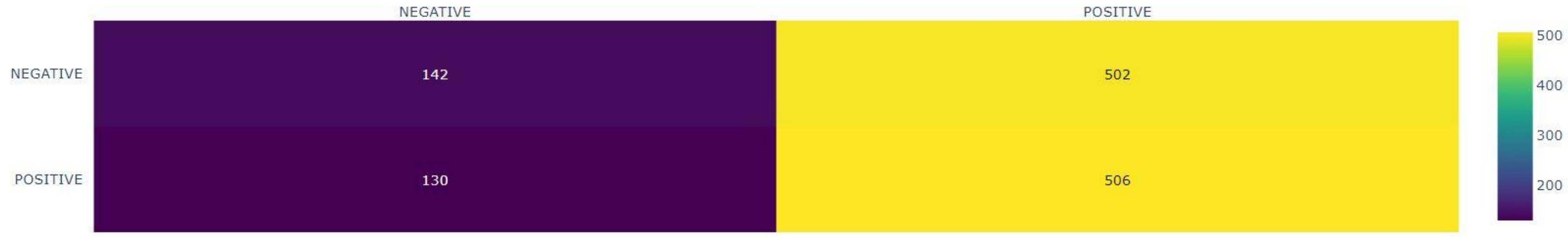
- **Steamming:** NLTK
- **Word Embedding: Glove**

Model: LSTM DNN




```
              precision    recall  f1-score   support

    NEGATIVE       0.52      0.22      0.31       644
    POSITIVE       0.50      0.80      0.62       636

    accuracy                           0.51      1280
   macro avg       0.51      0.51      0.46      1280
weighted avg       0.51      0.51      0.46      1280

ROC AUC score :  0.508
Average Precision score :  0.501
```

Confusion matrix

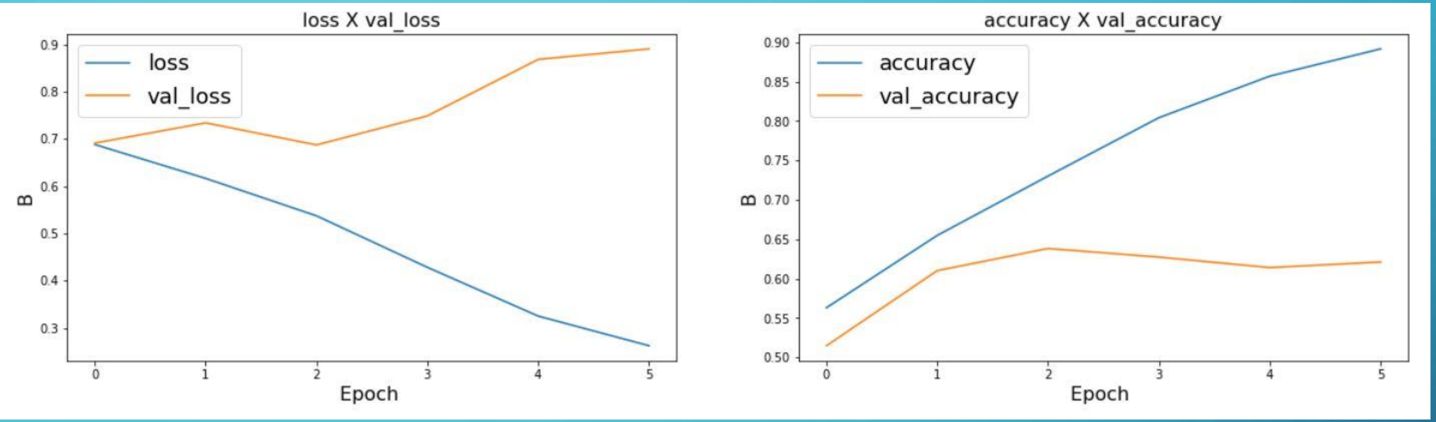


Modèle sur mesure avancé : Neural Networks avec Keras

Text pre-processing:

- **Stemming :** NLTK
- **Word Embedding: Word2Vec**

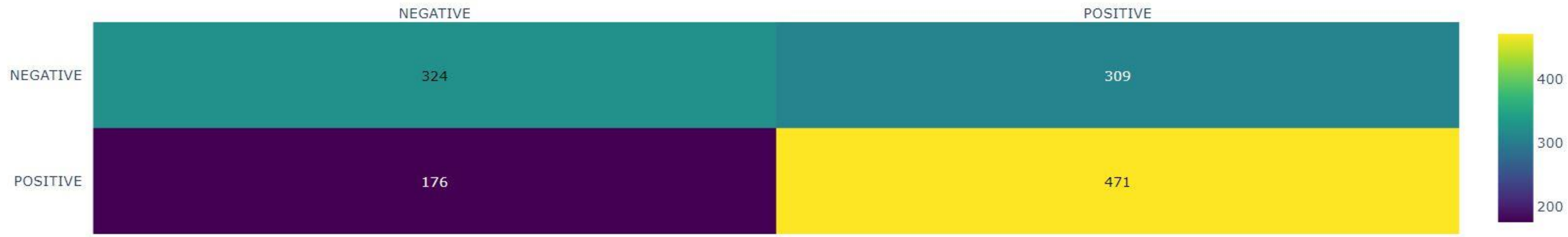
Model: LSTM DNN



	precision	recall	f1-score	support
NEGATIVE	0.65	0.51	0.57	633
POSITIVE	0.60	0.73	0.66	647
accuracy			0.62	1280
macro avg	0.63	0.62	0.62	1280
weighted avg	0.63	0.62	0.62	1280

ROC AUC score : 0.62
Average Precision score : 0.577

Confusion matrix

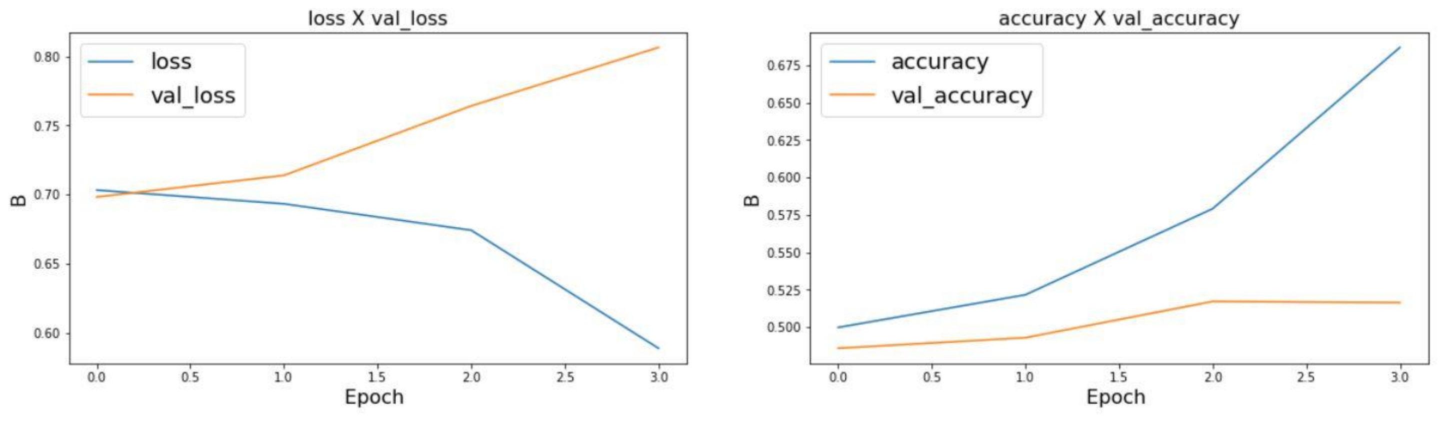


Modèle sur mesure avancé : Neural Networks avec Keras

Text pre-processing:

- **Lemmatization:** NLTK
- **Word Embedding: FastText**

Model: LSTM DNN

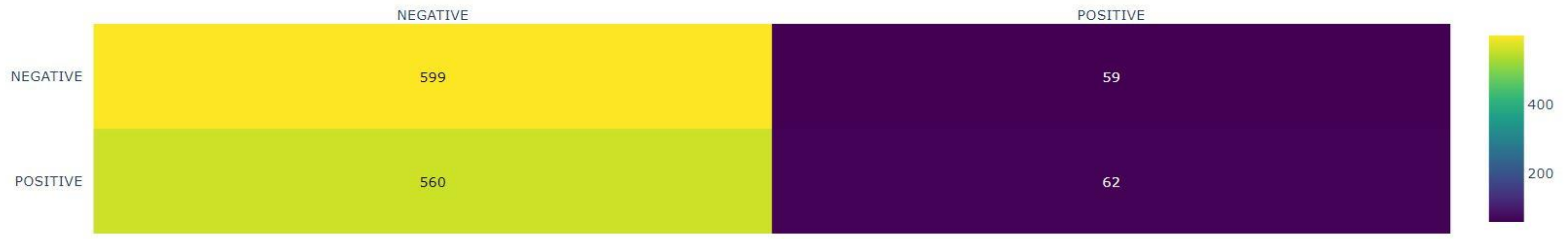


	precision	recall	f1-score	support
NEGATIVE	0.52	0.91	0.66	658
POSITIVE	0.51	0.10	0.17	622
accuracy			0.52	1280
macro avg	0.51	0.51	0.41	1280
weighted avg	0.51	0.52	0.42	1280

ROC AUC score : 0.505

Average Precision score : 0.489

Confusion matrix



HaggingFace: BERT fine-tuning

Model: Vanilla BERT model : bert-base-uncased

```
Model: "tf_bert_for_sequence_classification"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
bert (TFBertMainLayer)       multiple                  109482240
_________________________________________________________________
dropout_37 (Dropout)         multiple                  0
_________________________________________________________________
classifier (Dense)           multiple                  1538
=================================================================
Total params: 109,483,778
Trainable params: 109,483,778
Non-trainable params: 0
_________________________________________________________________
None
```

```
              precision    recall  f1-score   support

    NEGATIVE       0.30      0.38      0.33       650
    POSITIVE       0.11      0.08      0.10       630

    accuracy                           0.23      1280
   macro avg       0.21      0.23      0.21      1280
weighted avg       0.21      0.23      0.22      1280

ROC AUC score :  0.229
Average Precision score :  0.461
```

Confusion matrix

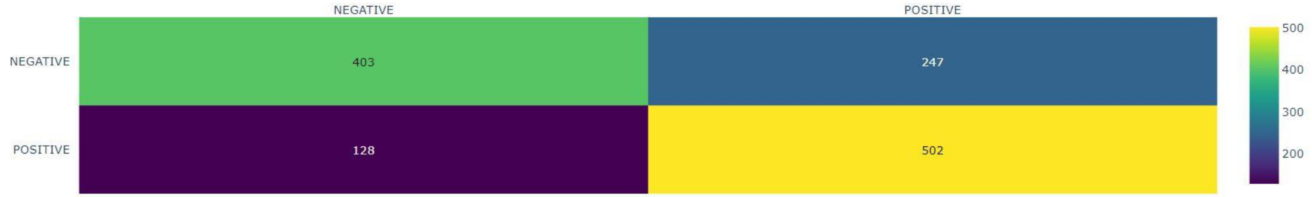

```
Text :  Or am I just wishing for the impossible... lmao
True sentiment :  1
Predicted sentiment :  0.581
```

Model: English tweets adapted model : vinai/bertweet-base

```
Model: "tf_roberta_for_sequence_classification"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
roberta (TFRobertaMainLayer) multiple                  134309376
_________________________________________________________________
classifier (TFRobertaClassif multiple                  592130
=================================================================
Total params: 134,901,506
Trainable params: 134,901,506
Non-trainable params: 0
_________________________________________________________________
None
```

```
              precision    recall  f1-score   support

    NEGATIVE       0.76      0.62      0.68       650
    POSITIVE       0.67      0.80      0.73       630

    accuracy                           0.71      1280
   macro avg       0.71      0.71      0.71      1280
weighted avg       0.72      0.71      0.70      1280

ROC AUC score :  0.708
Average Precision score :  0.634
```

Confusion matrix



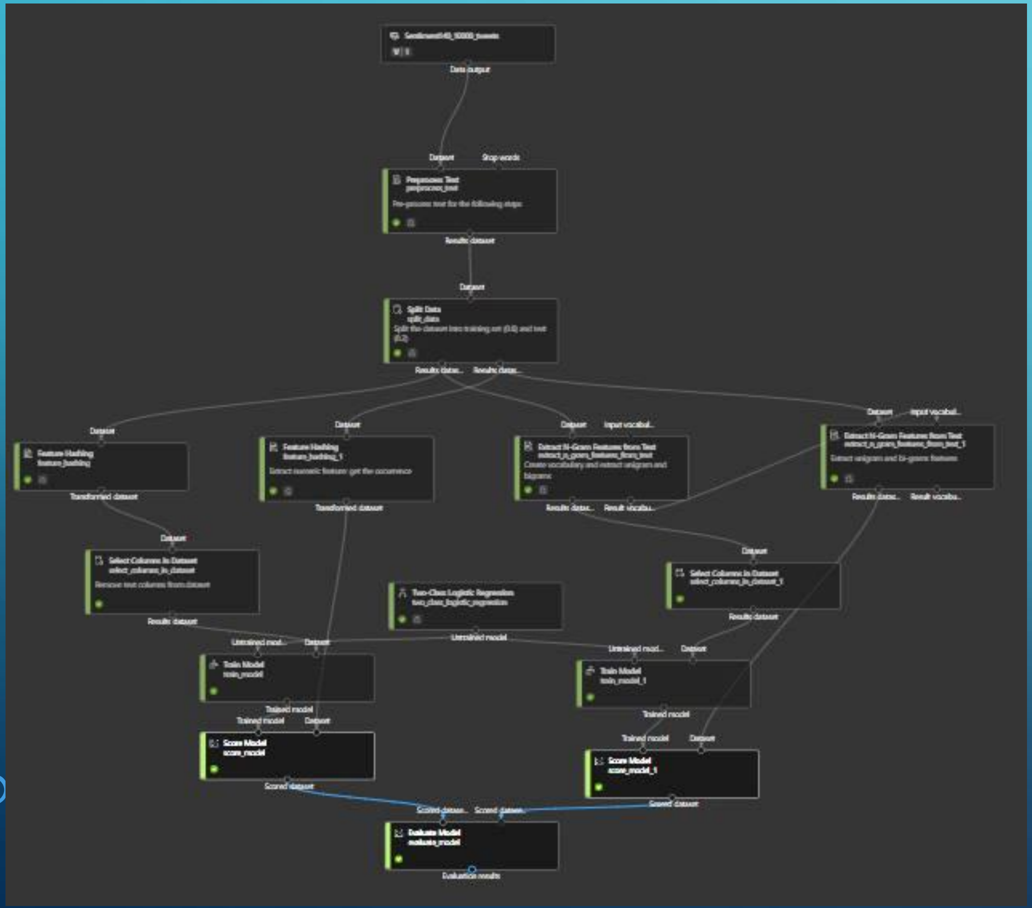

```
Text :  @angelmalfoy Init. Bella or is it Belle? (can't member lol) is wayyy too depressed. It's just like, GO AWAY TWILIGHT! aha
True sentiment :  1
Predicted sentiment :  0.505
```

AzureML Studio: Designer

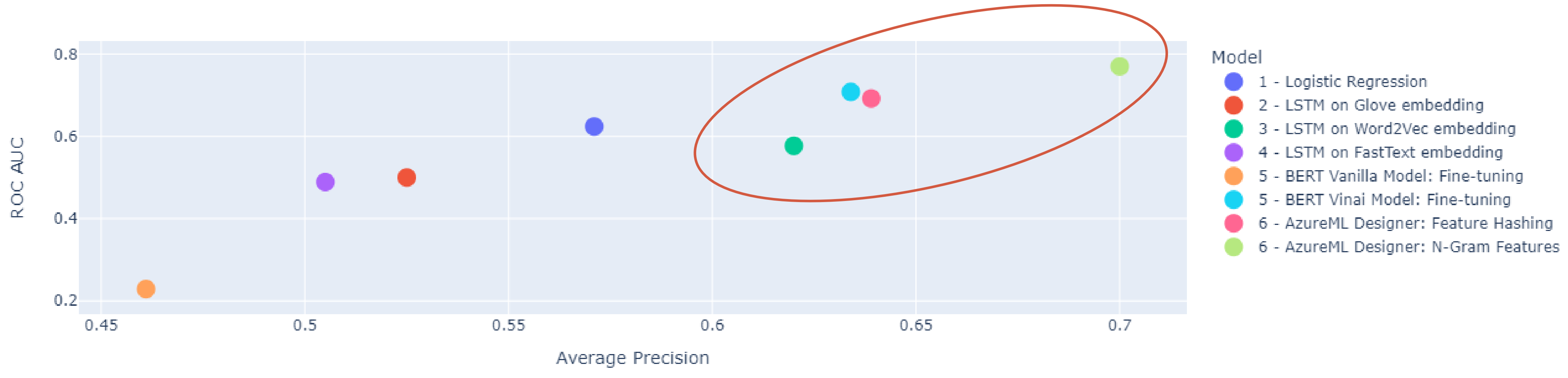
Text pre-processing :

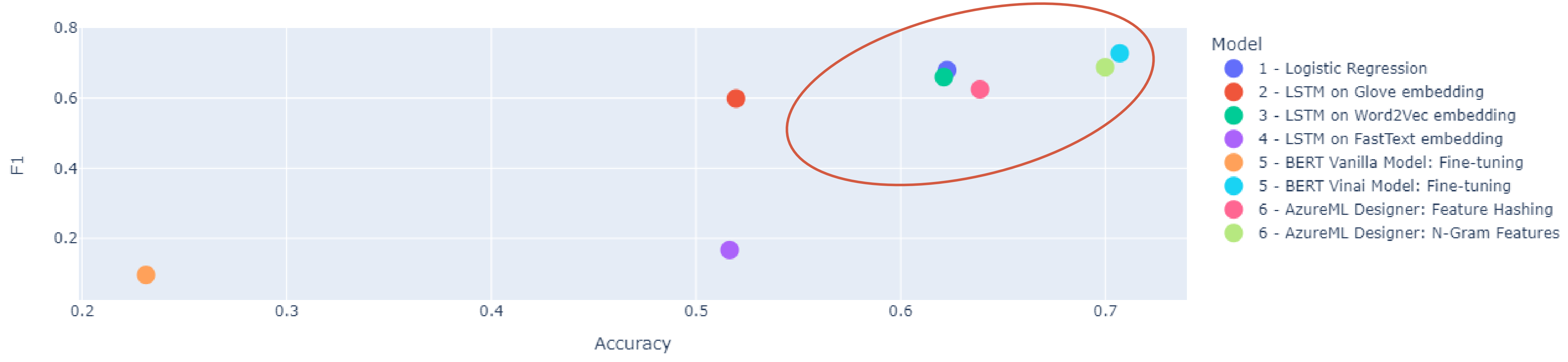
- Feature Hashing
- N-Gram Features

Model: Logistic regression



Comparaison des résultats des modèles





Conclusions...

La meilleure solution est de construire un modèle parmi les candidats suivants à déployer en production :

- AzureML Designer
- BERT Vinai Model-Fine tuning
- LSTM + word2vec + steamming

Déployement local vers Streamlit:

Nous avons déployé le modèle LSTM localement grâce à Streamlit