



REALISEZ UN DASHBOARD ET ASSUREZ UNE VEILLE TECHNIQUE





SOMMAIRE

Contexte

**I.
Dashboard**

**II. Etat de
l'Art**

**III.
Résultats**



Contexte du Projet 7 : Dashboard

Conception, Dashboard itératif de crédit scoring

Contexte :

Une société spécialisée dans les crédits à la consommation, y compris pour les personnes ayant peu ou pas d'historique de prêt, cherche à développer un outil de scoring. Cet outil doit permettre de déterminer la probabilité qu'un client rembourse son crédit. En outre, la société souhaite garantir la transparence du processus de scoring afin que les clients puissent comprendre les critères utilisés pour évaluer leur solvabilité.

Objectif :

Développer et mettre en place un outil de scoring crédit transparent qui évalue la probabilité de remboursement des prêts, en tenant compte des clients ayant peu ou pas d'historique de crédit.

Mission :

Construire un modèle de scoring pour prédire automatiquement la probabilité de faillite d'un client. Analyser les features les plus contributives, tant au niveau global que local, pour garantir la transparence du score. Mettre en production le modèle via une API avec une interface de test, et adopter une approche MLOps complète, incluant le suivi des expérimentations et l'analyse du data drift en production.



□ Stockage du modèle ?

- Mlflow

□ Accessibilité du modèle ?

- Locale : <http://127.0.0.1:5000/api/>
- CLOUD : <https://predictions-app-projet7-f3b0b3d90518.herokuapp.com/api/>

□ Test du modèle + Dashboard ?

- Streamlit Cloud

Création d'un Dashboard StreamLit qui utilise le modèle entraîné stocké dans Mlflow accessible depuis **Heroku**.





□ Différents tests de prédiction effectués avec MLFlow, ainsi qu'à travers l'API Flask en local .

```
C:\WINDOWS\system32>cd C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\Notebook
C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\Notebook>python server_local.py
C:\Python39\lib\site-packages\sklearn\base.py:376: InconsistentVersionWarning: Trying to unpickle estimator LabelEncoder from version 1.5.0 when using version 1.5.1. This might lead to breaking code
or invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
    warnings.warn(
        * Serving Flask app 'server_local'
        * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
    * Running on all addresses (0.0.0.0)
    * Running on http://127.0.0.1:3000
    * Running on http://192.168.1.85:3000
Press CTRL+C to quit
127.0.0.1 - - [27/Aug/2024 21:05:07] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:05:23] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:05:36] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:07:52] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:08:17] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:08:26] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:08:26] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:09:10] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:09:14] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 21:59:26] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:21:27] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:21:50] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:21:55] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:29:29] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:29:44] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:29:53] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:00] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:06] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:14] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:22] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:27] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:33] "POST /api/ HTTP/1.1" 200 -
127.0.0.1 - - [27/Aug/2024 22:30:41] "POST /api/ HTTP/1.1" 200 -
```



Réponse brute du serveur : [[0.63895802188395, 0.36104197811605]]

Prédiction : 36

- Réponse Brute du Serveur : [[0.63895802188395, 0.36104197811605]]
- Classe 0 (probabilité = 63.89%) : Selon notre modèle, il y a 63.89% de chances que l'observation appartienne à la classe 0.
- Classe 1 (probabilité = 36.10%) : Il y a 36.10% de chances que l'observation appartienne à la classe 1.



□ Développement d'une API Flask en Python, hébergée sur Heroku, pour la prédition des demandes de prêts.

Les différentes démarches

Créations d'un Bucket S3 pour héberger le modèle sur AWS

```
C:\WINDOWS\system32>aws configure
AWS Access Key ID [None]: AKIAST6S7IRG2LV3ZVOI
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]: json

C:\WINDOWS\system32>aws s3 ls

C:\WINDOWS\system32>aws s3 mb s3://mon-bucket-test
make_bucket failed: s3://mon-bucket-test An error occurred (BucketAlreadyExists) when calling the CreateBucket operation: The requested bucket name is not available. The bucket namespace is shared by all users of the system. Please select a different name and try again.

C:\WINDOWS\system32>aws configure
AWS Access Key ID [*****ZVOI]:
AWS Secret Access Key [*****xf60]:
Default region name [us-east-1]: eu-west-3
Default output format [json]:

C:\WINDOWS\system32>aws s3 ls

C:\WINDOWS\system32>aws s3 mb s3://votre-nom-de-bucket --region eu-west-3
make_bucket: votre-nom-de-bucket

C:\WINDOWS\system32>aws s3 ls
2024-08-28 23:47:06 votre-nom-de-bucket

C:\WINDOWS\system32>aws s3 cp "C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model" s3://votre-nom-de-bucket/model/ --recursive
upload: ..\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model\MLmodel to s3://votre-nom-de-bucket/model/MLmodel
upload: ..\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model\python_env.yaml to s3://votre-nom-de-bucket/model/python_env.yaml
upload: ..\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model\conda.yaml to s3://votre-nom-de-bucket/model/conda.yaml
upload: ..\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model\registered_model_meta to s3://votre-nom-de-bucket/model/registered_model_meta
upload: ..\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model\requirements.txt to s3://votre-nom-de-bucket/model/requirements.txt
upload: ..\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\mlflow_runs\290555362347125930\1e46374402274ffe9572106d93203ef9\artifacts\model\model.pkl to s3://votre-nom-de-bucket/model/model.pkl
```

```
C:\WINDOWS\system32>aws s3 ls s3://votre-nom-de-bucket/model/
2024-08-28 23:52:42      549 MLmodel
2024-08-28 23:52:42      459 conda.yaml
2024-08-28 23:52:42   3654221 model.pkl
2024-08-28 23:52:42     129 python_env.yaml
2024-08-28 23:52:42      62 registered_model_meta
2024-08-28 23:52:42   274 requirements.txt

C:\WINDOWS\system32>aws s3 ls
2024-08-28 23:47:06 votre-nom-de-bucket

C:\WINDOWS\system32>aws s3 mb s3://mon-projet-ml --region eu-west-3
make_bucket: mon-projet-ml

C:\WINDOWS\system32>aws s3 cp s3://votre-nom-de-bucket/ s3://mon-projet-ml/ --recursive
copy: s3://votre-nom-de-bucket/model/conda.yaml to s3://mon-projet-ml/model/conda.yaml
copy: s3://votre-nom-de-bucket/model/python_env.yaml to s3://mon-projet-ml/model/python_env.yaml
copy: s3://votre-nom-de-bucket/model/MLmodel to s3://mon-projet-ml/model/MLmodel
copy: s3://votre-nom-de-bucket/model/registered_model_meta to s3://mon-projet-ml/model/registered_model_meta
copy: s3://votre-nom-de-bucket/model/requirements.txt to s3://mon-projet-ml/model/requirements.txt
copy: s3://votre-nom-de-bucket/model/model.pkl to s3://mon-projet-ml/model/model.pkl

C:\WINDOWS\system32>aws s3 ls s3://mon-projet-ml/
PRE model

C:\WINDOWS\system32>aws s3 rb s3://votre-nom-de-bucket --force
delete: s3://votre-nom-de-bucket/model/conda.yaml
delete: s3://votre-nom-de-bucket/model/python_env.yaml
delete: s3://votre-nom-de-bucket/model/model.pkl
delete: s3://votre-nom-de-bucket/model/registered_model_meta
delete: s3://votre-nom-de-bucket/model/MLmodel
delete: s3://votre-nom-de-bucket/model/requirements.txt
remove_bucket: votre-nom-de-bucket

C:\WINDOWS\system32>aws s3 ls
2024-08-29 00:03:28 mon-projet-ml

C:\WINDOWS\system32>
```





Création d'une nouvelle application sur Heroku

```
C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.herokuapp.com/auth/cli/browser/69ce3647-24e5-4272-ae61-bb8155b41946?requestor=SFMyNTY,g2gDbQAAAA05NC4yMzkuMTEuMTU2bgYACOAbm5EBYgABUYA,7YG8Ndbj5SmudtZnTkvpzh4ESDw4ChmQ
olxqTj-2i-U
Logging in... done
Logged in as angekhoty2@gmail.com

C:\Users\Infogene\Documents\Khoty Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku\heroku create predictions-app-projet7
Creating @ predictions-app-projet7... done
https://predictions-app-projet7-f3b0b3d90518.herokuapp.com/ | https://git.heroku.com/predictions-app-projet7.git

C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku>cd C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\Notebook
C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\Notebook>python server_distant.py
C:\Python39\lib\site-packages\sklearn\base.py:376: InconsistentVersionWarning: Trying to unpickle estimator LabelEncoder from version 1.5.0 when using version 1.5.1. This might lead to breaking code
or invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
    warnings.warn(
        "Serving Flask app 'server_distant'"
    )
    # Debug mode: off
    # WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
    # Running on all addresses (0.0.0.0)
    # Running on http://127.0.0.1:5000
    # Running on http://192.168.1.85:5000
Press CTRL+C to quit

Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git init
Reinitialized existing Git repository in C:/Users/Infogene/Documents/Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku/.git/
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git add mlflow_runs/
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git add server_distant.py
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git add Procfile
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git add requirements.txt
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git commit -m "Updated server_distant with s3 model loading"
[master e3c38c5] Updated server_distant with s3 model loading
 3 files changed, 16 insertions(+), 9 deletions(-)
```



```
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git remote remove heroku
```

```
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git remote add heroku https://git.heroku.com/predictions-app-projet7.git
```

```
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ git push heroku master
Enumerating objects: 674, done.
Counting objects: 100% (674/674), done.
Delta compression using up to 4 threads
Compressing objects: 100% (372/372), done.
Writing objects: 100% (674/674), 1.24 MiB | 7.89 MiB/s, done.
Total 674 (delta 143), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (143/143), done.
remote: Updated 1253 paths from 79cb5f9
remote: Compressing source files... done.
remote: Building source:
remote:
```

```
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé\DOSSIER FORMATION
DATA SCIENTIST\PROJET 7 ML\API_Heroku (master)
$ heroku logs --tail --app predictions-app-projet7
2024-08-28T23:04:26.576708+00:00 app[web.1]: ^^^^^^^^^^^^^^
2024-08-28T23:04:26.576708+00:00 app[web.1]: File "/app/.heroku/python/lib/python3.12/site-packages/botocore/signers.py", line 105, in handler
2024-08-28T23:04:26.576708+00:00 app[web.1]: return self.sign(operation_name, re
```



```
Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé/DOSSIER FORMATION
DATA SCIENTIST/PROJET 7 ML/API_Heroku (master)
$ heroku config:set AWS_ACCESS_KEY_ID=AKIAST6S7IRG2LV3ZVOI
Setting AWS_ACCESS_KEY_ID and restarting predictions-app-projet7... done, v4
AWS_ACCESS_KEY_ID: AKIAST6S7IRG2LV3ZVOI

Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé/DOSSIER FORMATION
DATA SCIENTIST/PROJET 7 ML/API_Heroku (master)
$ heroku config:set AWS_SECRET_ACCESS_KEY=iE+f2evmObIJshB0Dkg065YtwxPeTvuLavaIx60
Setting AWS_SECRET_ACCESS_KEY and restarting predictions-app-projet7... done, v5
AWS_SECRET_ACCESS_KEY: iE+f2evmObIJshB0Dkg065YtwxPeTvuLavaIx60

Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé/DOSSIER FORMATION
DATA SCIENTIST/PROJET 7 ML/API_Heroku (master)
$ heroku restart --app predictions-app-projet7
Restarting dynos on predictions-app-projet7... done

Infogene@LAPTOP-2-Mohamed-Ali MINGW64 ~/Documents/Khoty_Privé/DOSSIER FORMATION
DATA SCIENTIST/PROJET 7 ML/API_Heroku (master)
$ heroku logs --tail --app predictions-app-projet7
2024-08-28T23:17:36.759956+00:00 app[web.1]: botocore.exceptions.PartialCredenti
alsError: Partial credentials found in env, missing: AWS_SECRET_ACCESS_KEY
2024-08-28T23:17:36.760056+00:00 app[web.1]: [2024-08-28 23:17:36 +0000] [9] [IN
FO] Worker exiting (pid: 9)
2024-08-28T23:17:36.842762+00:00 app[web.1]: [2024-08-28 23:17:36 +0000] [17] [E
```



```
C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\API_Heroku>cd C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\Notebook
C:\Users\Infogene\Documents\Khoty_Privé\DOSSIER FORMATION DATA SCIENTIST\PROJET 7 ML\Notebook>streamlit run dashboardStreamlit_Flask_distant_heroku.py

You can now view your Streamlit app in your browser.

Local URL: http://localhost:8501
Network URL: http://192.168.1.85:8501

<IPython.core.display.HTML object>
<IPython.core.display.HTML object>
<IPython.core.display.HTML object>
c:\python39\lib\site-packages\sklearn\base.py:376: InconsistentVersionWarning: Trying to unpickle estimator LabelEncoder from version 1.5.0 when using version 1.5.1. This might lead to breaking code or invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
    warnings.warn(
c:\python39\lib\site-packages\shap\plots\_waterfall.py:315: UserWarning:

FigureCanvasAgg is non-interactive, and thus cannot be shown

<IPython.core.display.HTML object>
c:\python39\lib\site-packages\sklearn\base.py:376: InconsistentVersionWarning:

Trying to unpickle estimator LabelEncoder from version 1.5.0 when using version 1.5.1. This might lead to breaking code or invalid results. Use at your own risk. For more info please refer to:
https://scikit-learn.org/stable/model_persistence.html#security-maintainability-limitations
```



□ Conception d'une application Streamlit qui interagit avec l'API Flask : Interface utilisateur de l'application

Sommaire :

Page d'accueil

À propos de l'application

Cette application a été développée pour permettre une transparence totale dans l'analyse financière, en aidant à comprendre les scores de crédit et les décisions financières de manière claire et accessible.

Conseils Financiers

- Épargnez régulièrement : Mettez de côté une partie de vos revenus chaque mois.
- Minimisez vos dettes : Évitez de contracter des dettes à haut intérêt.
- Investissez intelligemment : Diversifiez vos investissements pour réduire les risques.

Assistance et contact

Si vous avez des questions ou avez besoin d'assistance, veuillez contacter notre équipe de support.

- Courriel : support@finapp.com
- Téléphone : +33 1 23 45 67 89
- Chat en ligne : Disponible 24h/24 et 7j/7 sur notre site web.

Bonjour, Bienvenue sur votre portail de transparence financière :



Faciliter l'analyse et la compréhension des résultats pour tous les utilisateurs.

Objectif de l'application :

Aider clients et conseillers à naviguer facilement dans les données et à prendre des décisions éclairées.

Sommaire :

Informations Clients

Veuillez choisir le numéro de votre client :

Veuillez sélectionner un client pour afficher ses informations détaillées.

L'application vous permet de visualiser les scores de crédit, l'historique des clients et d'autres informations financières importantes.

Informations Clients :

Veuillez choisir le numéro de votre client :

N° Client : 100038
Prénom : Éléonore
Nom : Thierry

Résultats du prêt :

Réponse brute de l'API : [[0.3200379825296229, 0.6799620174703771]]





Niveau de remboursement : 32%

Importance des caractéristiques globales :

EXT_SOURCE_2: +0.31
EXT_SOURCE_3: -0.29
EXT_SOURCE_1: +0.18
CODE_GENDER: +0.11
DAYS_EMPLOYED: +0.09
AMT_ANNUITY: +0.09
PAYMENT_RATE: +0.09
INSTAL_DPD_MEAN: +0.08
OWN_CAR_AGE: +0.07
Sum of 610 other features: +2.04

mean(|SHAP value|)

Importance des caractéristiques locales :

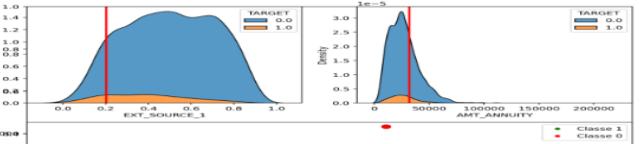
0.051 = PAYMENT_RATE: +0.31
-13040 = DAYS_BIRTH: -0.29
32067 = AMT_ANNUITY: +0.18
nan = ACTIVE_DAYS_CREDIT_MAX: +0.11
0.202 = EXT_SOURCE_1: +0.11
-4262 = DAYS_ID_PUBLISH: -0.09
3 = NAME_FAMILY_STATUS_Married: +0.09
625500 = AMT_CREDIT: +0.08
3 = NAME_CONTRACT_TYPE_Cashloans: +0.07
610 other features: +2.04

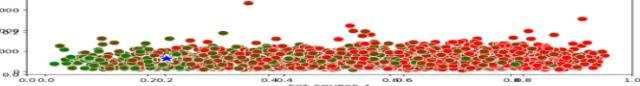
E(RX) = 0.032

Analyse des variables importantes :

Veuillez choisir la variable N°1 : EXT_SOURCE_1

Veuillez choisir la variable N°2 : AMT_ANNUITY







Contexte du Projet 6 : Etat de L'art

Etat de l'Art : BERT vs DeBERTa, Vers une Nouvelle Génération de Modèles NLP



Contexte : Data Scientist au sein de l'entreprise "Place de Marché", qui prévoit de lancer une marketplace e-commerce. Sur ce site, les vendeurs pourront proposer des articles aux acheteurs en publiant une photo et une description de chaque produit.

Mission : Actuellement, l'attribution des catégories des articles est effectuée manuellement par les vendeurs, ce qui entraîne une certaine imprécision et une fiabilité limitée. La mission sera de réaliser une étude de faisabilité d'un moteur de classification automatique d'articles, en utilisant leurs images et descriptions. Je serai également chargé d'automatiser et d'optimiser ce processus pour garantir une classification plus précise et cohérente des produits.

Objectif : Automatiser l'attribution des catégories des articles pour améliorer la fiabilité et simplifier l'expérience utilisateur. Appliquer une classification supervisée sur des textes.

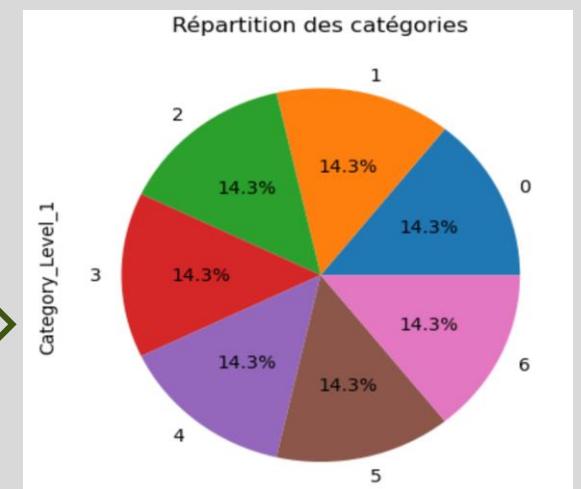
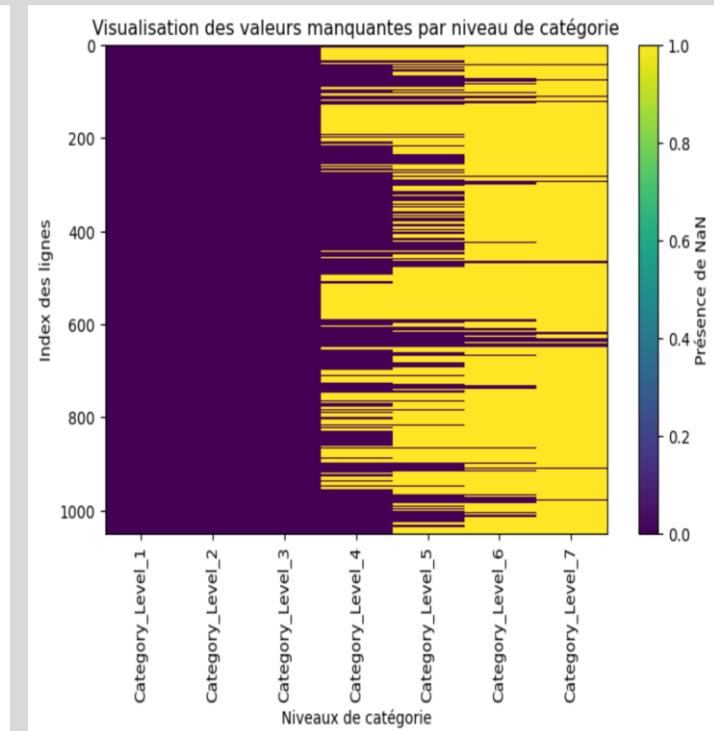
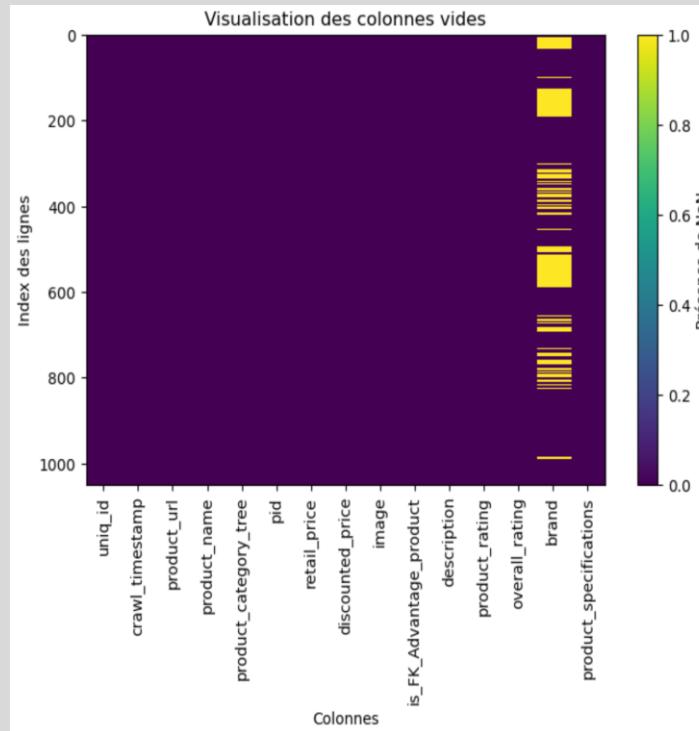


2. Présentation des données

Base de données comprenant 15 variables et 1050 enregistrements, avec 2% de valeurs manquantes.

Variables intéressantes :

- Nom du produit (entre 2 et 27 mots).
- Description du produit (entre 18 et 589 mots).
- Arbre des catégories (7 niveaux, entre 7 et 349 catégories).
- Images.



Un ensemble de données aux catégories équilibrées



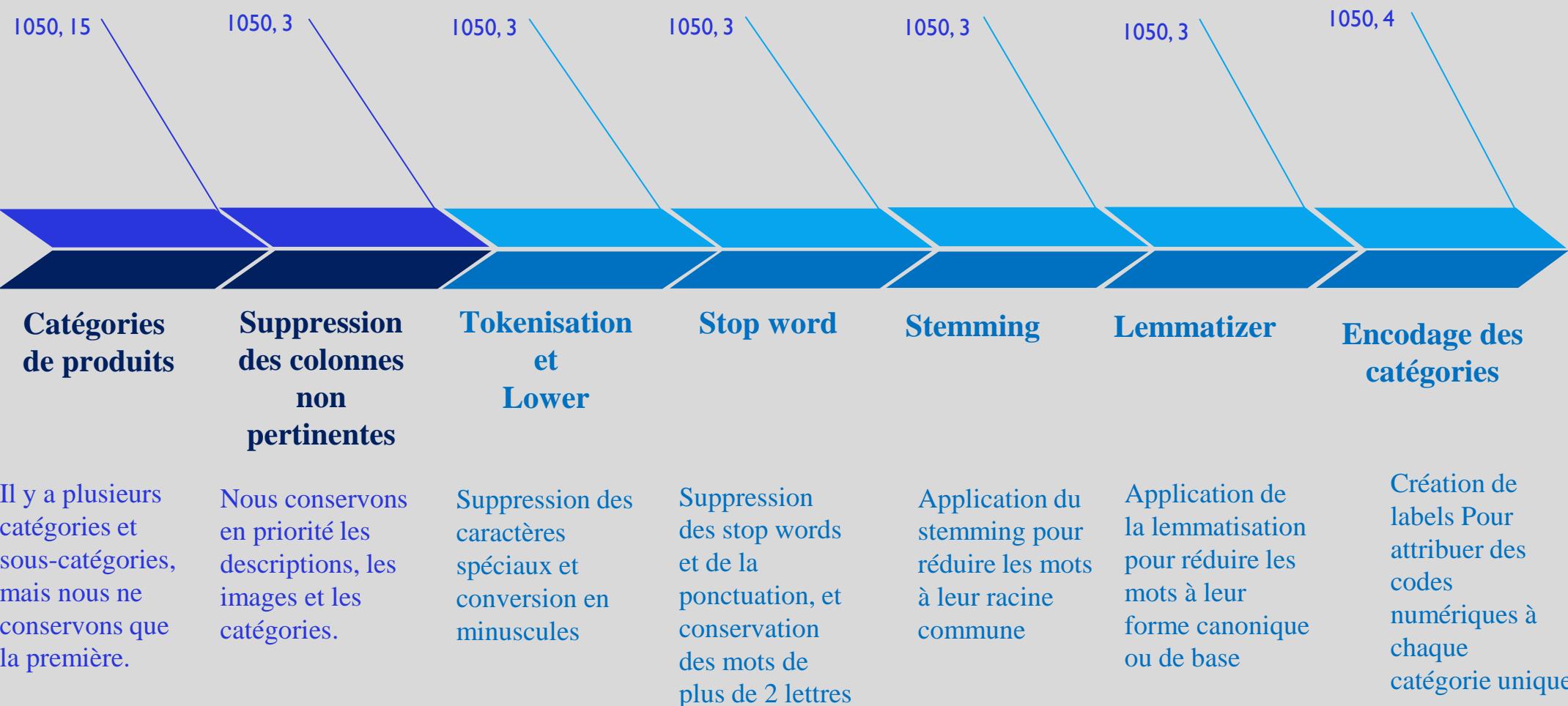
3. Détails des prétraitements, extractions de caractéristiques et résultats de l'étude de faisabilité

Procérons à la classification des données images et textes.

	Pré traitement	Extraction et description des caractéristiques pour la construction d'un vecteur numérique	Réduction de dimension	Clustering	Visualisation	Evaluation
Données textuelles	Extraction des tokens, nettoyage et création d'un vocabulaire	<ul style="list-style-type: none">❖ Bags of Words : Count-vectorizer, TF-IDF❖ Words Embedding : Word2Vec, BERT, USE	TruncatedSVD TSNE ACP	Algorithme de classification Kmeans	TSNE à l'aide de l'ACP	Calcul de l'Indice de Rand Ajusté (ARI)
Données textuelles (DeBERTa)	Extraction des tokens et nettoyage des données	<ul style="list-style-type: none">❖ Masquage désentrelacé (disentangled attention)❖ Masquage résiduel (enhanced masking)❖ Décodage optimisé				



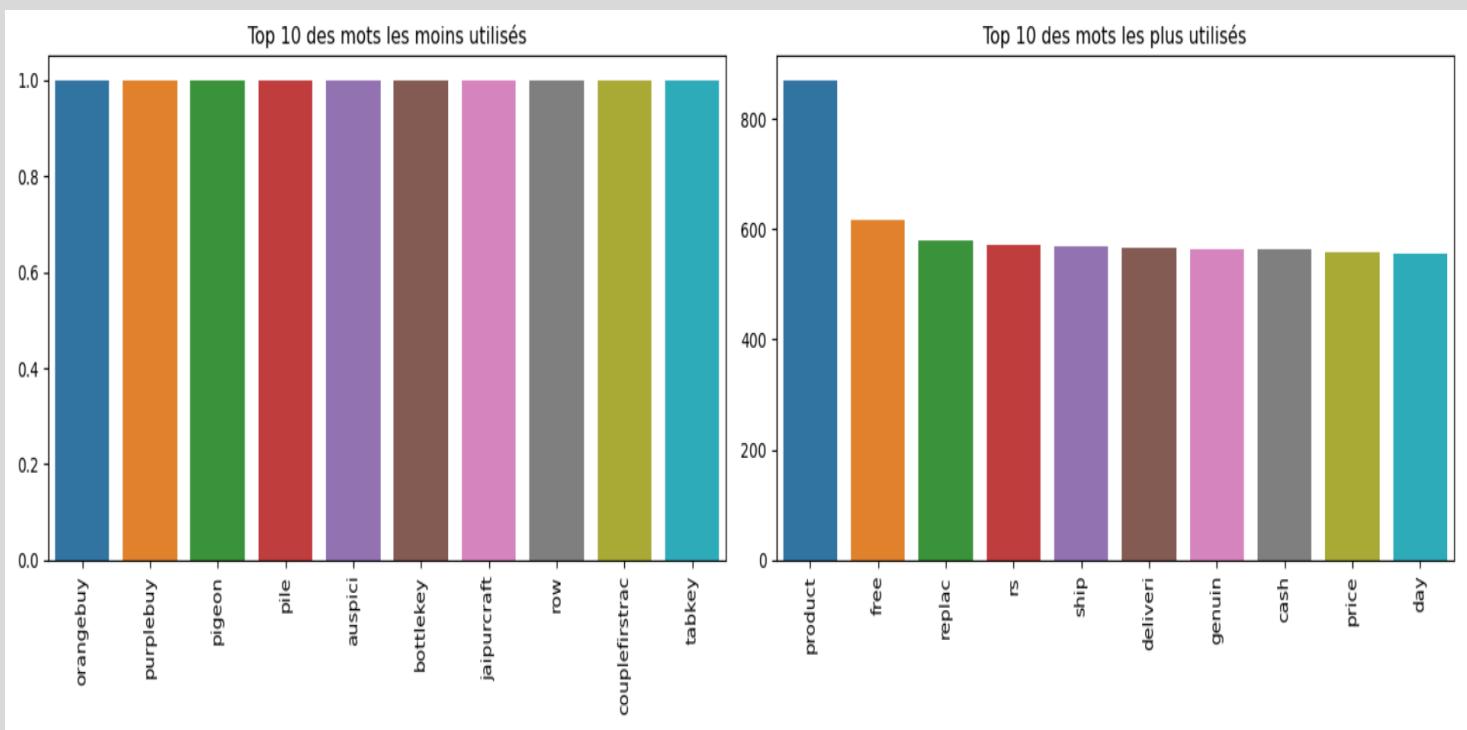
Les données textuelles ont été préparées de la manière suivante.



3.1 Traitement du Langage Naturel (NLP)

Extraction et simplification des données textuelles :

Fusion des textes du nom et de la description des produits en une seule variable (compris entre 21 et 593 mots).



Nous pouvons clairement voir les mots les plus utilisés.



□ Approche Bag-of-Words

Définition de CountVectorizer

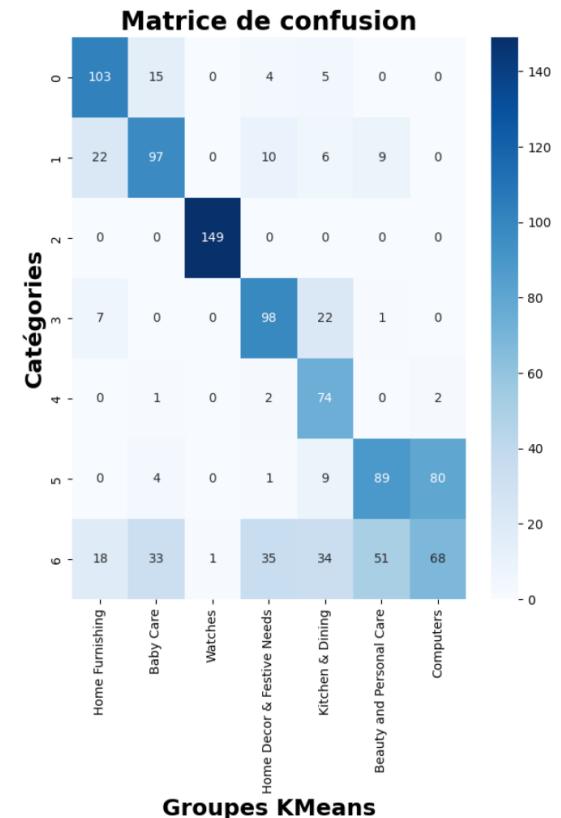
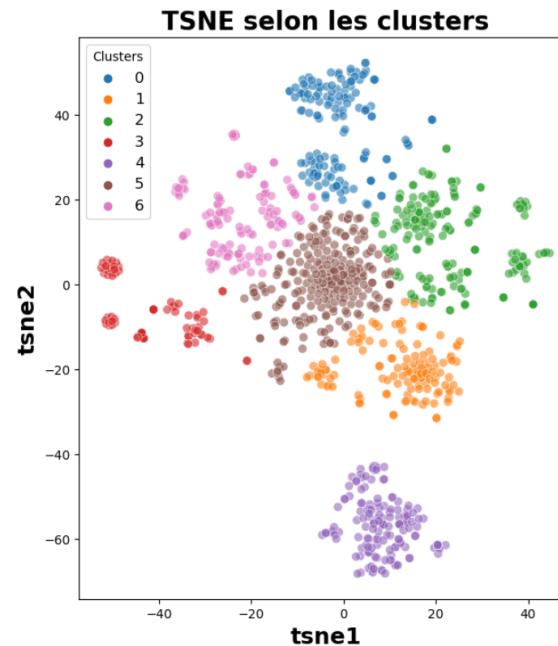
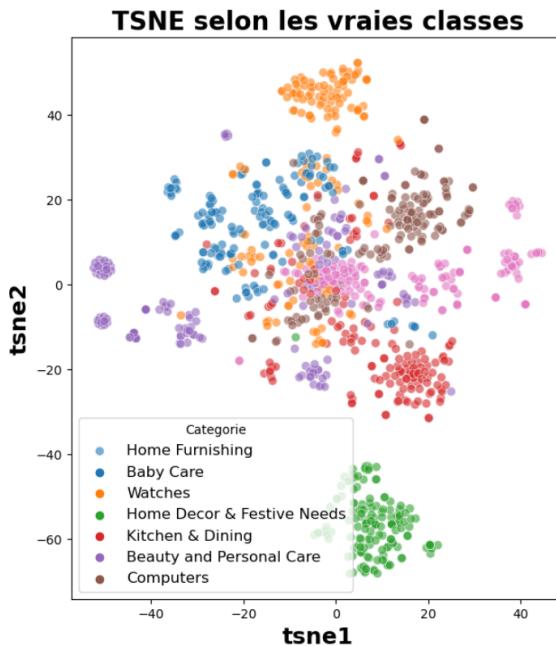
CountVectorizer est une classe de scikit-learn qui convertit une collection de documents textuels en une matrice de comptage de tokens. Chaque document est représenté par un vecteur indiquant la fréquence de chaque mot unique.

[Lien image: <https://towardsdatascience.com/basics-of-countvectorizer-e26677900f9c>]

	big	count	create	dataset	differnt	features	hello	james	name	notebook	of	python	this	try	trying	vectorizer	words
0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0	0	0
2	1	0	1	1	0	0	0	1	0	0	0	0	0	0	1	0	0
3	0	0	0	0	1	0	0	1	0	0	1	0	0	1	0	0	1
4	0	1	0	0	0	1	0	0	0	0	1	0	0	0	0	1	0



Résultats CountVectorizer



L'indice de Rand ajusté (ARI) pour ce modèle d'extraction de caractéristiques est de 0.421.

La classification comporte des erreurs et les catégories sont mal attribuées. En effet, la matrice de confusion montre des confusions significatives entre certaines catégories



Définition de TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) est une technique qui évalue l'importance d'un mot dans un document par rapport à un corpus. Contrairement au simple comptage des mots, TF-IDF pondère les mots en tenant compte de leur fréquence dans le document (TF) et de leur rareté dans le corpus (IDF), mettant ainsi en avant les mots significatifs tout en réduisant l'importance des mots courants.

[Lien image: <https://knowledge.dataiku.com/latest/ml-analytics/nlp/concept-natural-language-processing-challenges.html>]

REDUNDANT FEATURES



text

0	Eddard Stark is a king in the north.
1	A king but one king : kings are everywhere.
2	Hodor was different : he was not a king .
3	But the North could not change without him.

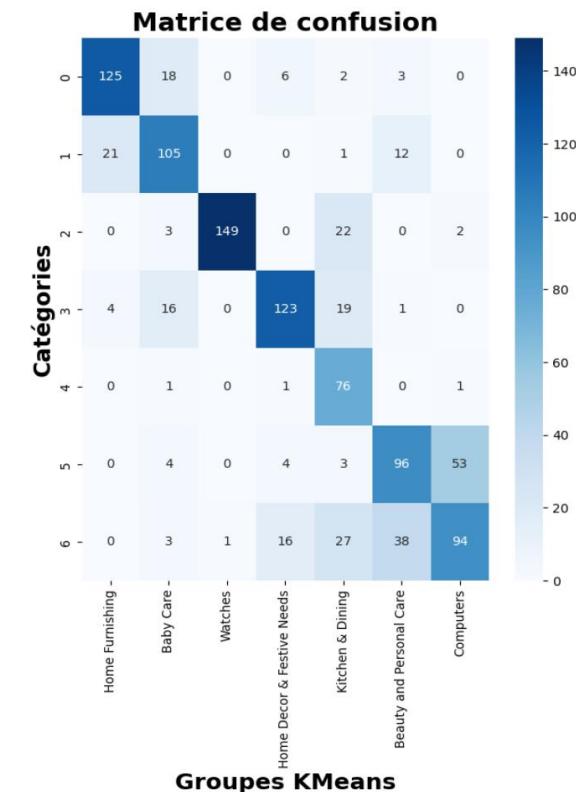
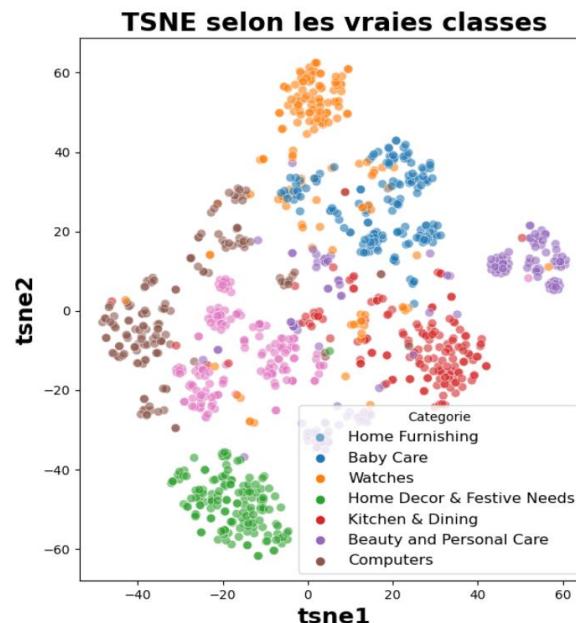
“North” vs. “north”? “north”
“king” vs. “kings”? “king”



	king	was	the	not	But	him	one	north	kings	is	in	he	Eddard	everywhere	different	could	change	but	are	Stark	North	Hodor	without
0	1	0	1	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0	0	1	0	0	0
1	2	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	1	1	0	0	0	0
2	1	2	0	1	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	1	0
3	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	1



Résultats TF-IDF



L'indice de Rand ajusté (ARI) pour ce modèle d'extraction de caractéristiques est de **0.519**.

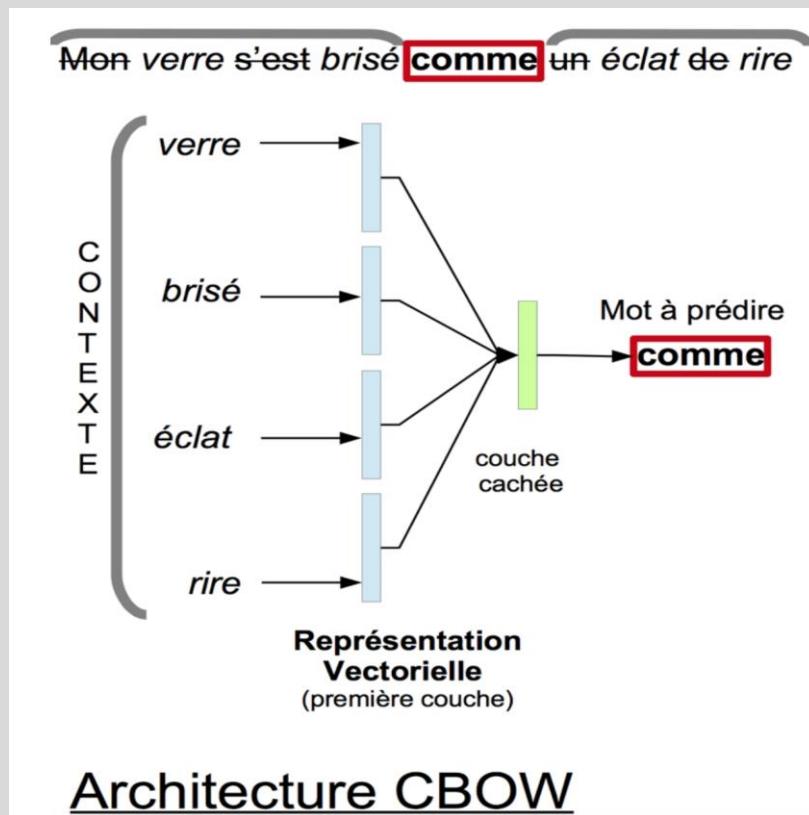
La classification est meilleure qu'avec le CountVectorizer. Les catégories sont relativement bien identifiées par l'algorithme. Cependant, il existe encore des confusions, comme indiqué par la matrice de confusion.



□ Approche Words Embedding

Définition de Word2Vec

Word2Vec est une méthode avancée qui représente les mots sous forme de vecteurs continus dans un espace vectoriel de dimension réduite. Contrairement à l'approche Bag-of-Words, Word2Vec capture les relations sémantiques entre les mots, de sorte que des mots similaires ont des vecteurs similaires. En utilisant les architectures CBOW et Skip-gram, Word2Vec fait des prédictions de mots basées sur leur contexte. [Lien image: <https://dataanalyticspost.com/Lexique/word2vec/>]



Développé par une équipe de **Google** en **2013** dirigée par **Tomas Mikolov**, il repose sur des **réseaux de neurones à deux couches**.



Résultats Word2Vec



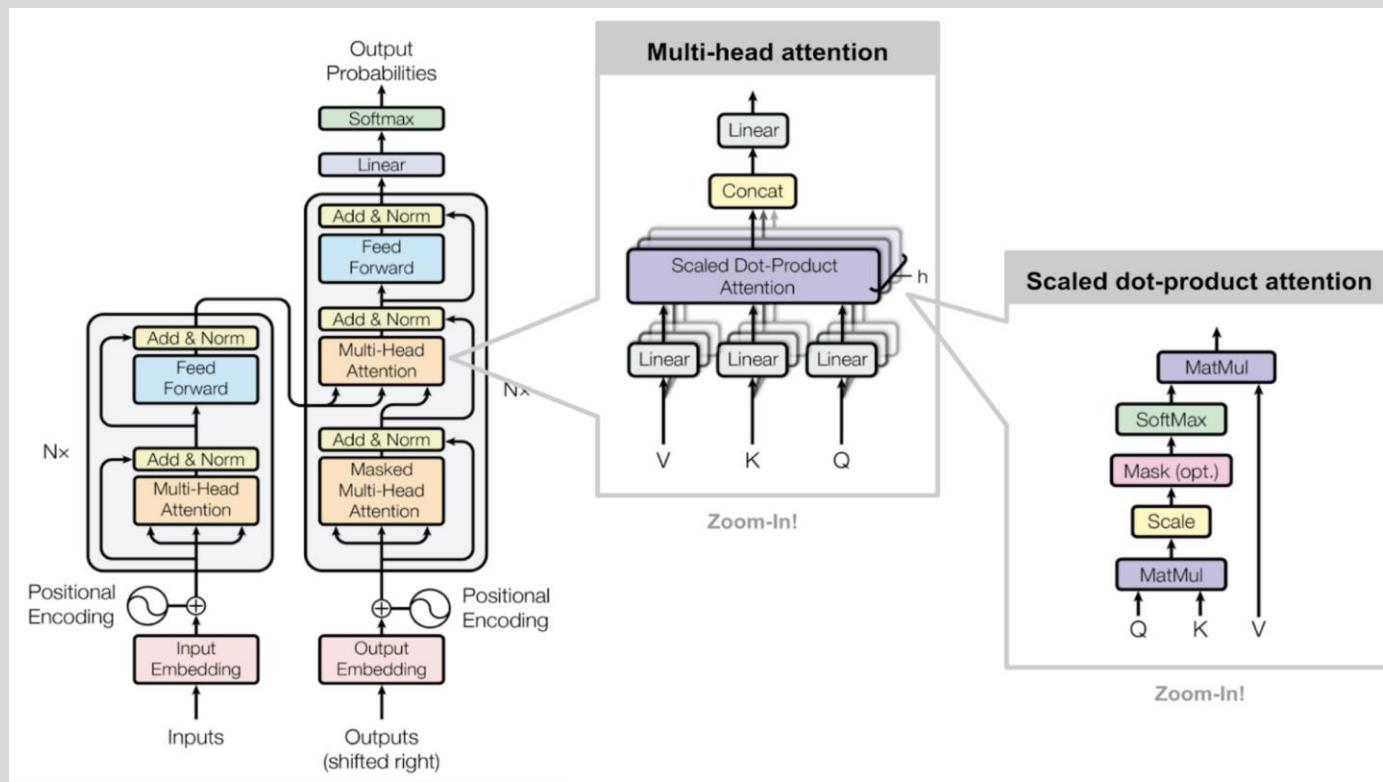
Nous observons une diminution des performances. Les catégories proches sont souvent mal attribuées, comme le montrent les clusters qui se chevauchent dans la visualisation TSNE et les confusions dans la matrice de confusion.



Définition de BERT

BERT (Bidirectional Encoder Representations from Transformers) est une méthode avancée pour obtenir des représentations contextuelles de mots et de phrases. Contrairement à Word2Vec, BERT prend en compte le contexte des mots dans les deux directions (avant et arrière), capturant ainsi des significations plus riches et précises.

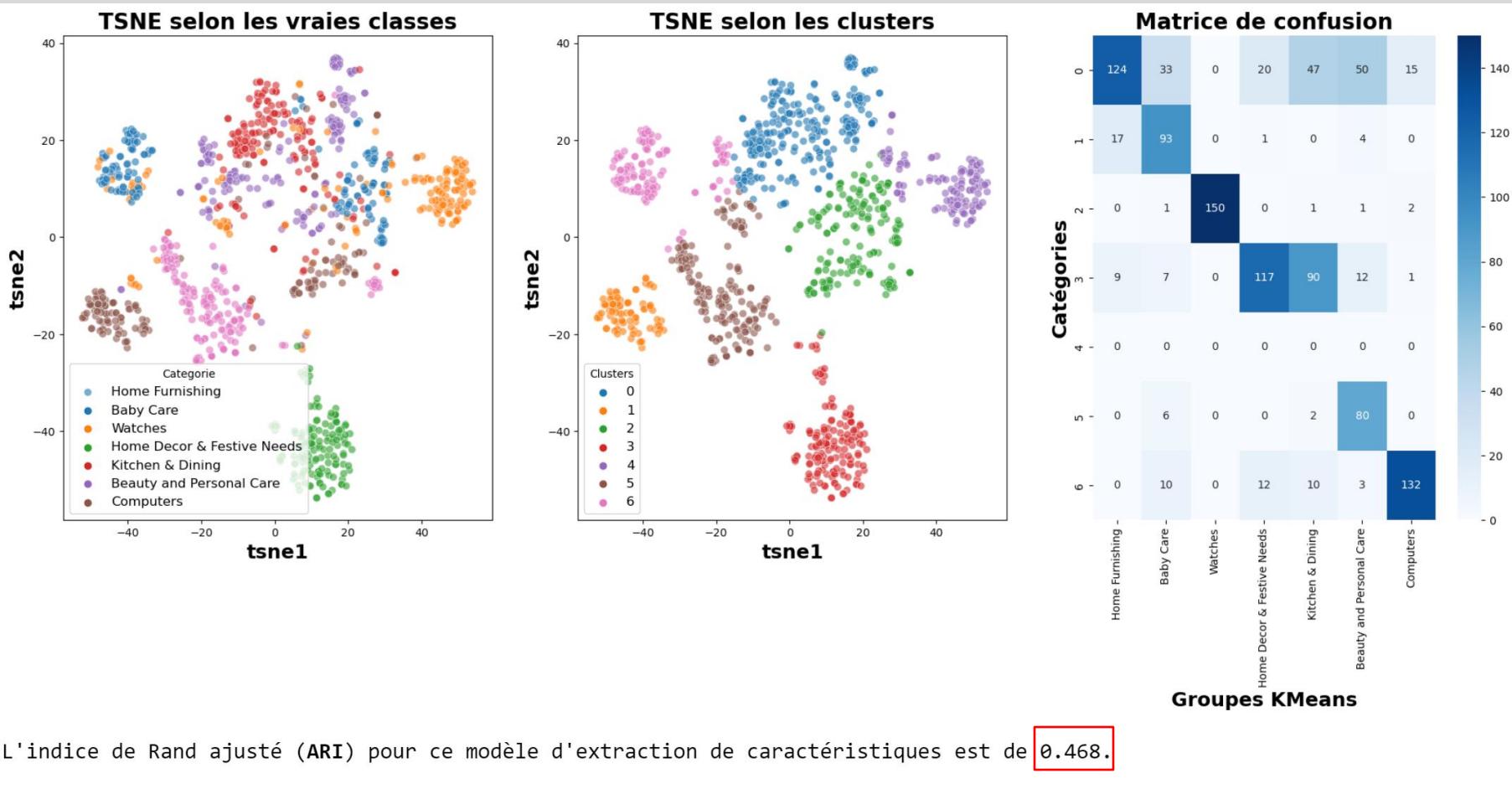
[Lie, image: <https://neptune.ai/blog/bert-and-the-transformer-architecture>]



BERT est un modèle de langage développé par **Google en 2018**. Il utilise l'architecture Transformer, qui repose sur des couches de neurones pour apprendre les relations contextuelles entre les mots ou sous-mots d'un texte. L'architecture comprend plusieurs blocs d'encodage, chacun composé de mécanismes d'attention multi-tête et de couches de feed-forward. Contrairement aux modèles directionnels qui lisent le texte séquentiellement, l'encodeur Transformer de BERT traite la séquence entière de mots simultanément, capturant ainsi le contexte complet des mots à la fois à gauche et à droite. Cela permet à BERT de générer des représentations bidirectionnelles des mots.



Résultats BERT



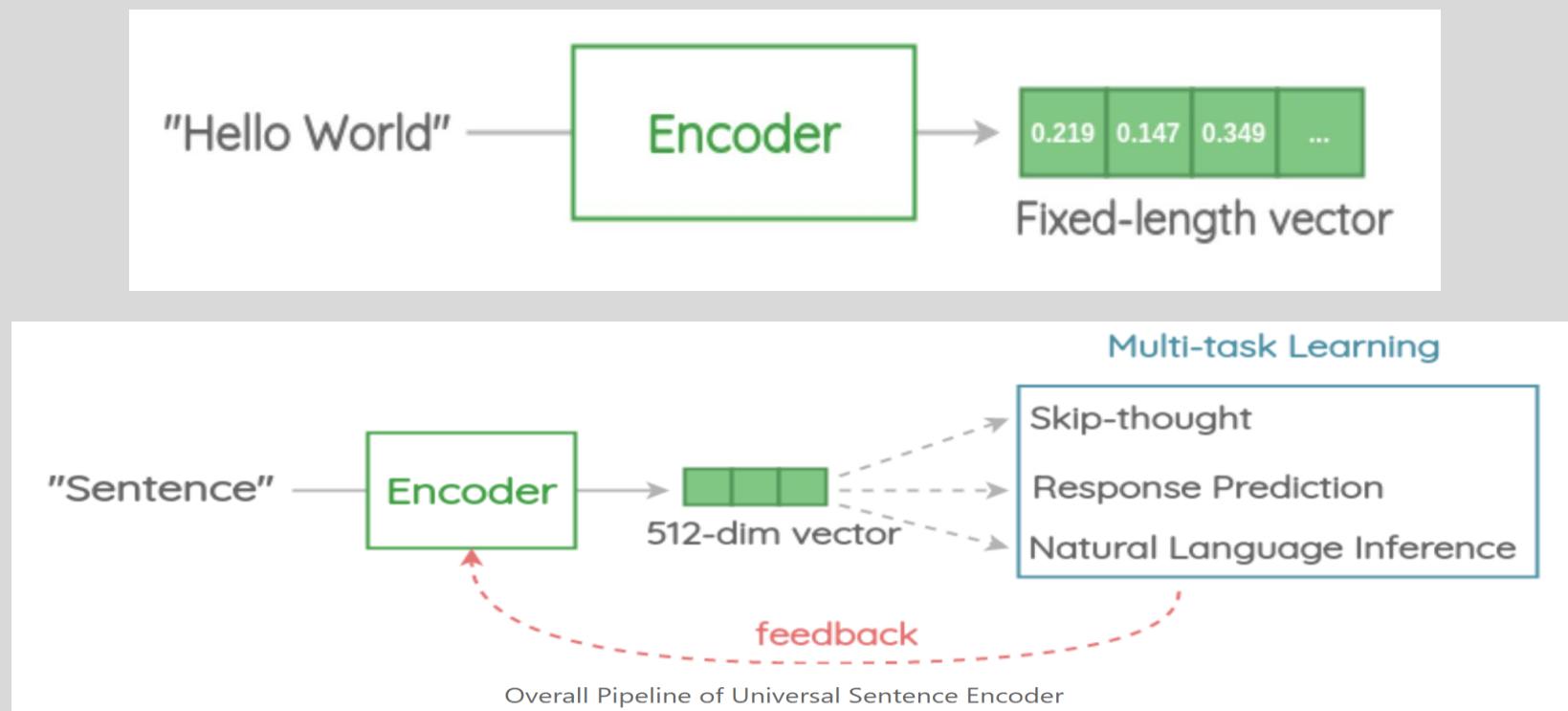
La classification est meilleure qu'avec Word2Vec. Les catégories proches sont relativement bien identifiées par l'algorithme.



Définition de USE

USE (Universal Sentence Encoder), développée par Google, produit des représentations vectorielles de haute qualité pour des phrases entières. Elle est particulièrement utile pour la classification, la similarité et le clustering de texte.

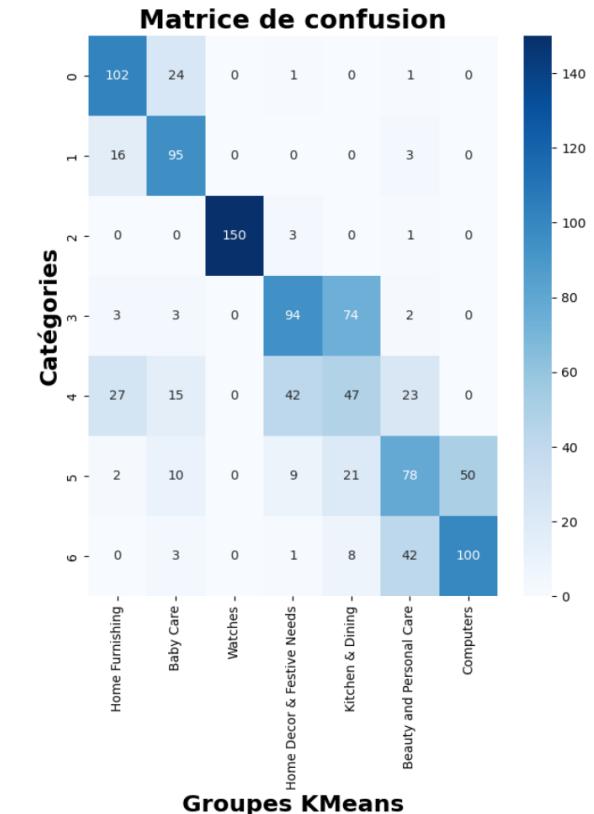
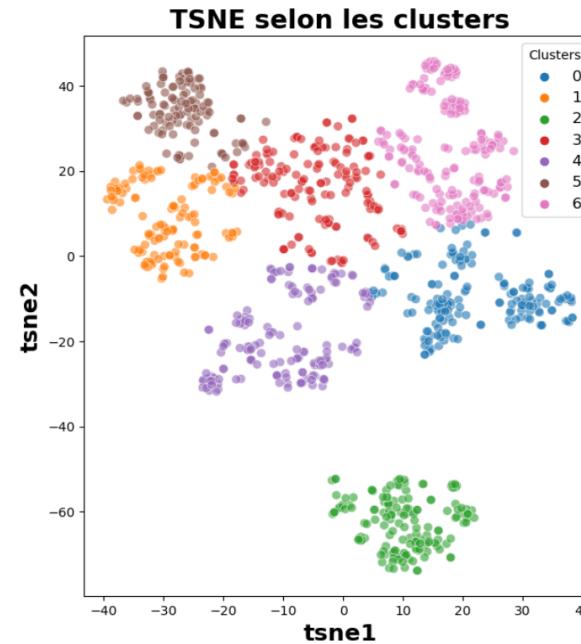
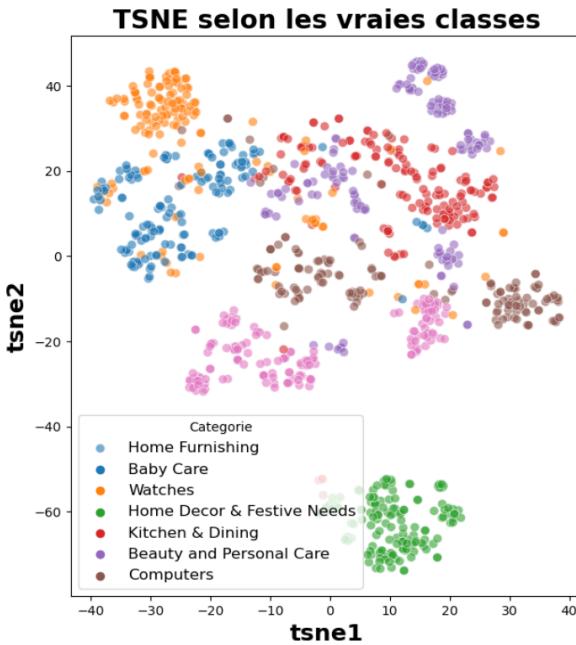
[Lien image: <https://amitness.com/posts/universal-sentence-encoder>]



L'Universal Sentence Encoder (USE) encode chaque phrase en un vecteur de 512 dimensions. En fonctionnant sur plusieurs tâches, il capture les caractéristiques sémantiques essentielles et élimine le bruit, produisant des représentations vectorielles robustes et polyvalentes. Ces vecteurs sont utiles pour diverses applications de traitement du langage naturel telles que la classification de texte, la similarité et l'inférence linguistique.



Résultats USE



L'indice de Rand ajusté (ARI) pour ce modèle d'extraction de caractéristiques est de 0.443.

Meilleure classification après le TF-IDF et BERT. Le principe d'embedding produit de meilleurs résultats et minimise les matrices vides.

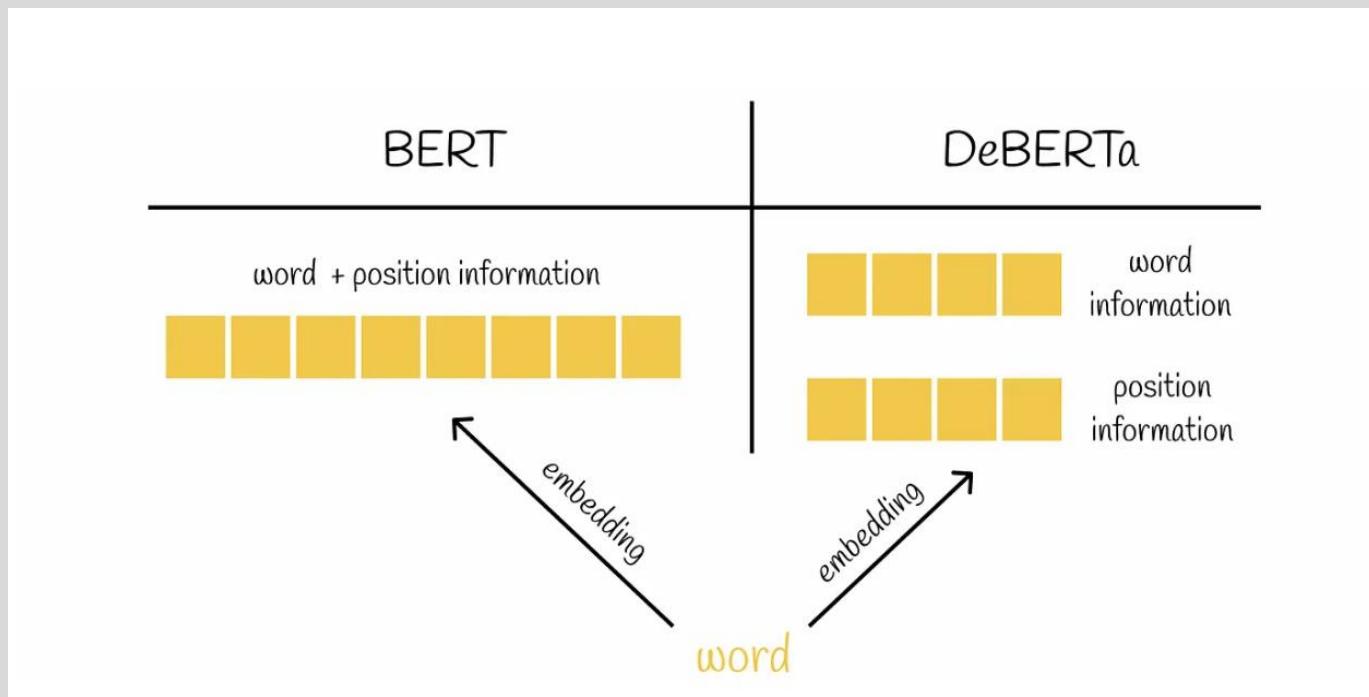


DeBERTa Attention désentrelacée

Définition de DeBERTa

DeBERTa (Decoding-enhanced BERT with Disentangled Attention) est un modèle de traitement du langage naturel (NLP) basé sur l'architecture **Transformer**. Il améliore BERT en séparant les informations de contenu et de position des mots dans des vecteurs distincts, ce qui permet de mieux capturer les relations contextuelles. De plus, DeBERTa utilise un décodeur de masque amélioré pour traiter efficacement les relations entre les mots. Ces innovations permettent d'éviter les pertes d'information et améliorent les performances sur des tâches telles que la classification de texte et la génération de réponses.

[Lien Image: <https://towardsdatascience.com/large-language-models-deberta-decoding-enhanced-bert-with-disentangled-attention-90016668db4b>]



- **BERT** : Il combine les informations de contenu du mot et de position dans un seul vecteur, ce qui peut entraîner une perte d'informations. Le modèle ne peut pas distinguer clairement si l'importance vient du mot lui-même ou de sa position dans la phrase.
- **DeBERTa** : Ce modèle sépare les informations de contenu et de position en deux vecteurs distincts, ce qui permet une meilleure compréhension des relations mot-position. Par exemple, si "chat" et "chien" apparaissent proches dans une phrase, DeBERTa reconnaît leur lien sémantique (des animaux). Cependant, s'ils sont éloignés, le modèle considérera cette relation comme moins pertinente. Cela montre comment DeBERTa capte mieux les relations contextuelles que BERT, qui combine tout en un seul vecteur.

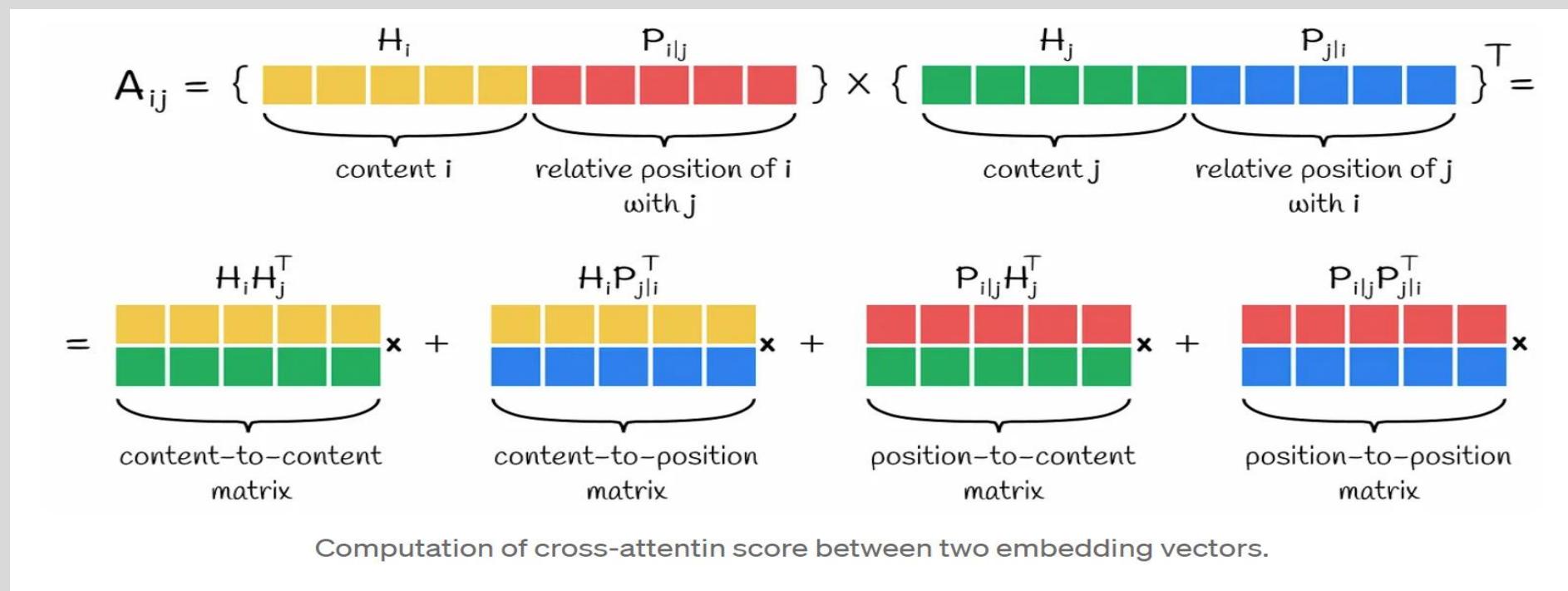


BeBERTa Attention désentrelacée Calcul du score

L'introduction de l'attention désentrelacée dans DeBERTa permet de calculer les scores d'attention en décomposant les embeddings en deux vecteurs : un pour le contenu et un pour la position. Le calcul des scores croisés entre deux embeddings se fait à partir de quatre types de matrices :

- **Content-to-content** : Comparaison du contenu des mots.
- **Content-to-position** : Relie le contenu du mot à la position relative de l'autre mot.
- **Position-to-content** : Relie la position relative à un mot spécifique.
- **Position-to-position** : Compare directement les positions relatives des mots.

Cette méthode permet au modèle de capturer de manière plus fine les relations contextuelles entre les mots et leurs positions, améliorant ainsi la précision du modèle NLP.





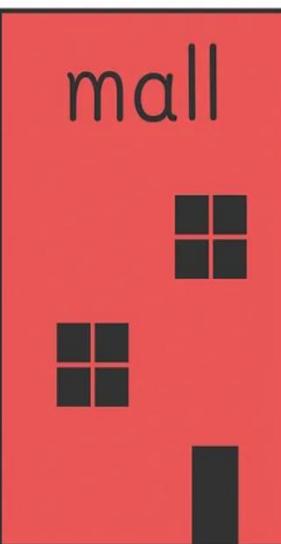
DeBERTa Décodeur de masque amélioré (DME)

Attention désentrelacée : contenu et position relative.

Ce n'est pas suffisant !

Le modèle DeBERTa utilise le **DME** (Décodeur de Masque Amélioré) pour intégrer les positions absolues dans la couche de décodage. Cela permet au modèle de mieux prédire les tokens masqués durant le pré-entraînement en combinant les informations de contenu et de position absolue pour une compréhension plus fine des relations entre les mots.

<<un nouveau ____ a ouvert à près du ____>>

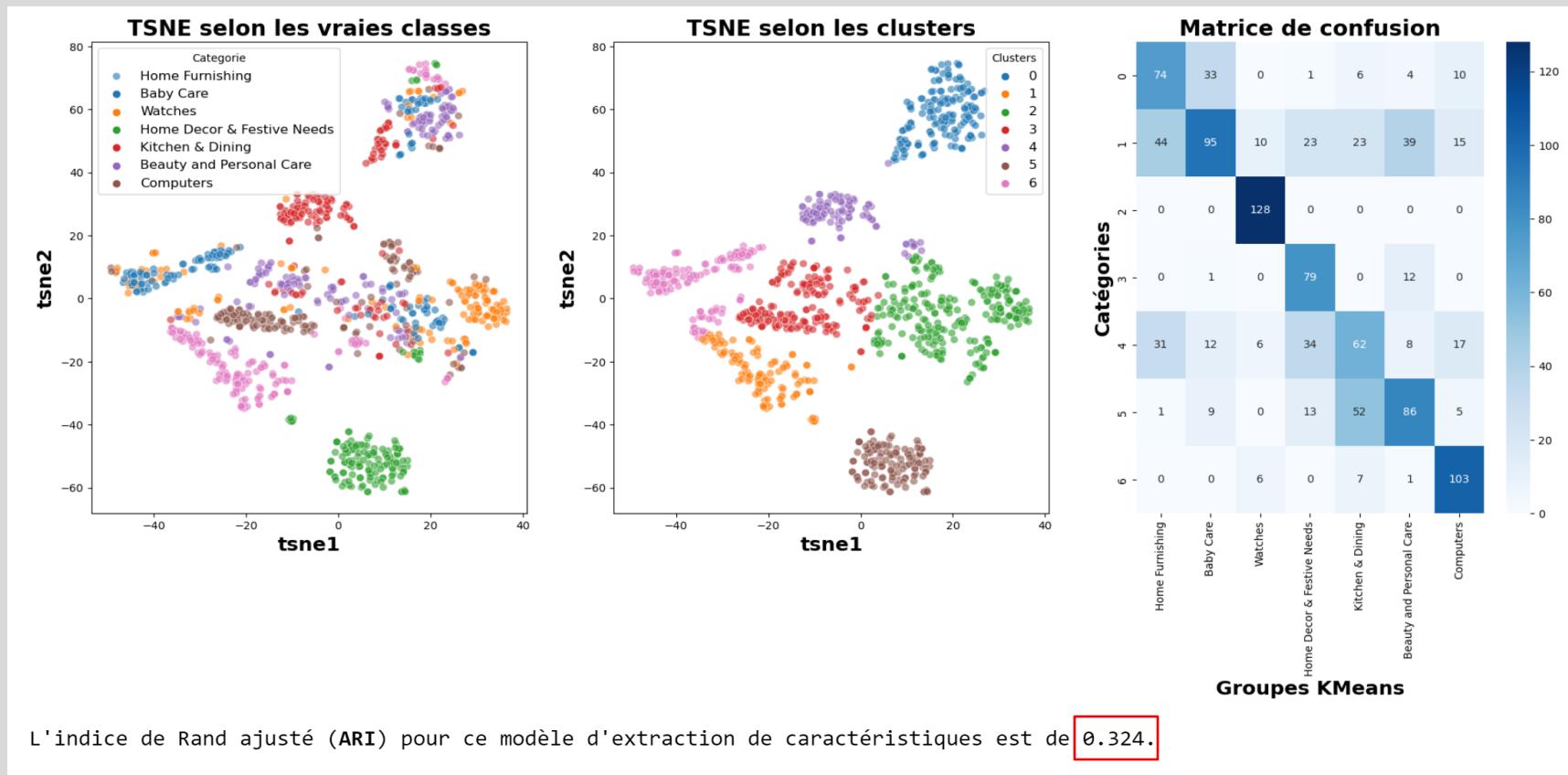


- Dans une phrase comme "**un nouveau magasin a ouvert près du centre commercial**", lorsque les mots "magasin" et "centre commercial" sont masqués, le modèle pourrait avoir du mal à déterminer correctement leur rôle sans tenir compte des positions absolues. Le modèle confondrait les deux mots en raison de leur proximité sémantique, mais leur rôle dans la phrase diffère selon leur position. Cette limitation souligne l'importance de connaître les positions absolues pour reconstruire correctement la phrase.



Résultats DeBERTa-base

DeBERTa-base : Un modèle DeBERTa plus petit avec environ 110 millions de paramètres. Il possède 12 couches de transformateurs, 12 têtes d'attention, et une taille d'embedding de 768. Ce modèle est moins gourmand en ressources, plus rapide à entraîner et à utiliser, mais offre des performances légèrement inférieures par rapport à DeBERTa-large, particulièrement adapté pour des tâches NLP standard et des environnements avec des ressources limitées.

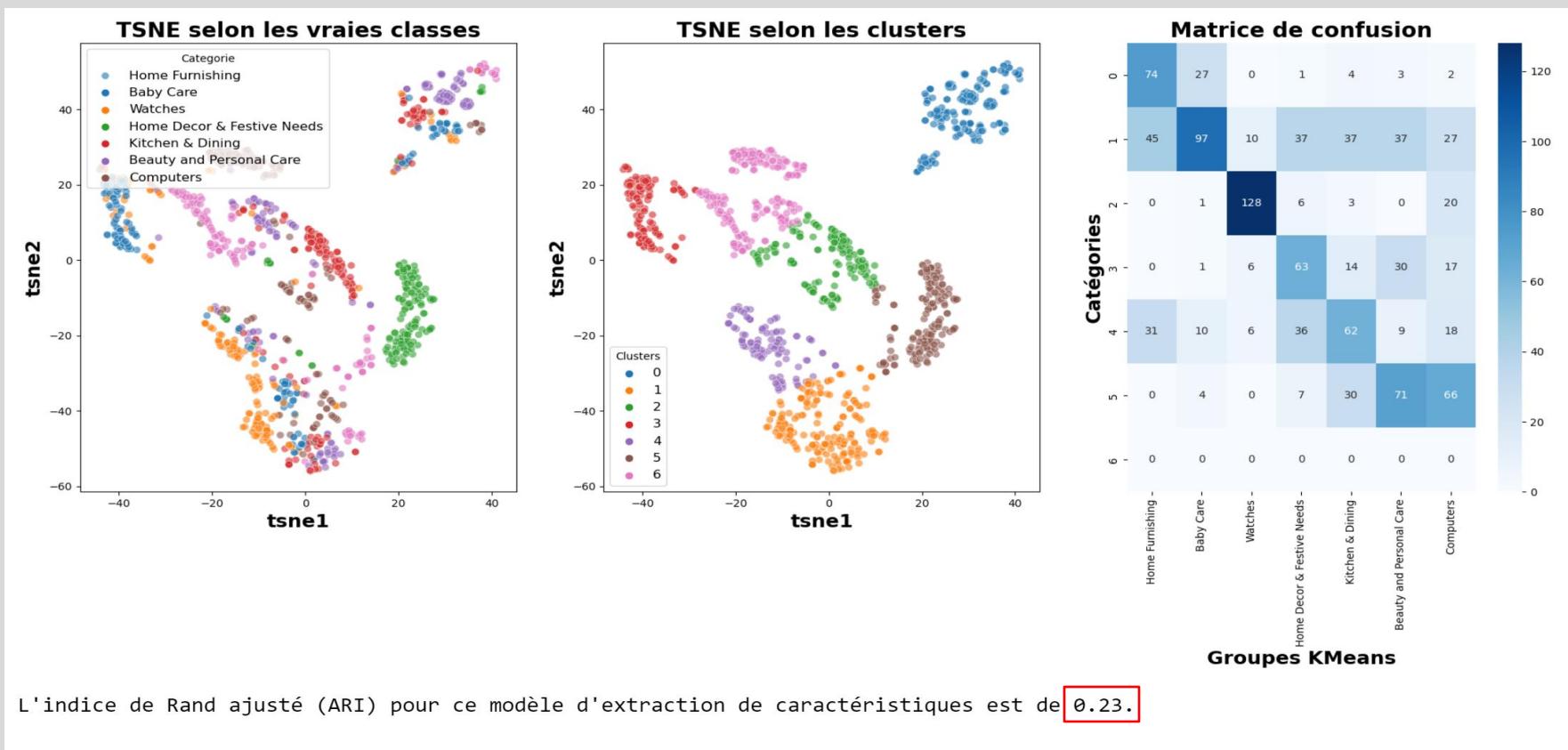


La classification comporte des erreurs et les catégories sont mal attribuées. En effet, la matrice de confusion montre des confusions significatives entre certaines catégories



Résultats DeBERTa-large

DeBERTa-large : Une version plus grande de DeBERTa avec environ 340 millions de paramètres. Il dispose de 24 couches de transformateurs, 16 têtes d'attention, et une taille d'embedding de 1024. Ce modèle est plus performant pour des tâches NLP complexes, offrant une meilleure capacité de modélisation des relations sémantiques, mais nécessite davantage de ressources computationnelles et de temps d'entraînement. Il est idéal pour des tâches où la précision est cruciale.



Performance moins meilleure que celle observer avec DeBERTa-bas. Les performances diminuent, avec des erreurs fréquentes dans les catégories proches. Cela se manifeste par des clusters qui se chevauchent dans la visualisation TSNE et des confusions récurrentes dans la matrice de confusion.



CONCLUSION

La comparaison entre **BERT** et **DeBERTa** montre que, sur ce dataset spécifique, BERT reste plus performant. Malgré ses avancées théoriques, **DeBERTa** n'a pas atteint les performances espérées, indiquant que des ajustements sont nécessaires pour exploiter son potentiel. L'amélioration des hyperparamètres, l'augmentation de la taille du dataset, et l'ajout de données supplémentaires pourraient aider. De plus, une réduction de la complexité ou l'utilisation d'un corpus plus spécialisé pourrait améliorer la généralisation et la vitesse des résultats de **DeBERTa**.