

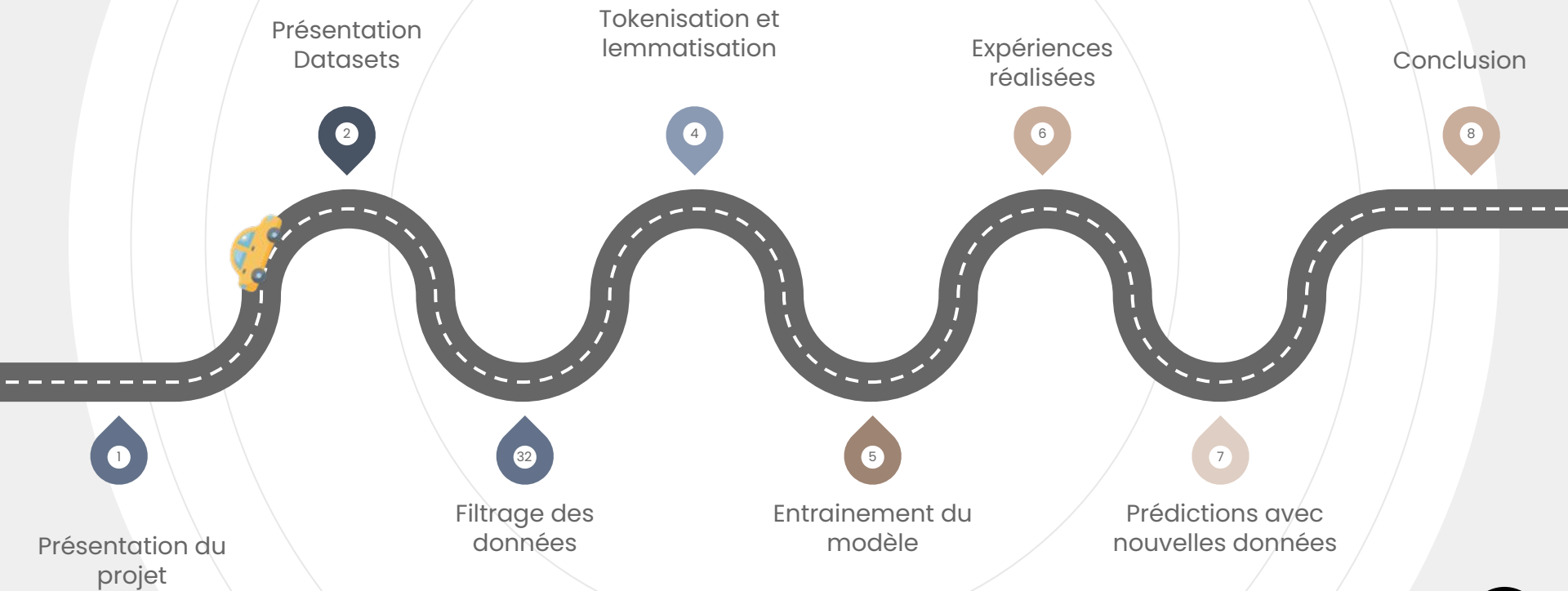


TATIA

**Détection automatique de genres
de film par IA**

Benjamin VALLEIX, Augustin GIRAUDIER

Plan



Présentation

Deviner le genre des films à partir de leur synopsis

Objectif :

- Entrée : fournir une liste de synopsis
- Sortie : liste de genres (au départ mono-label)



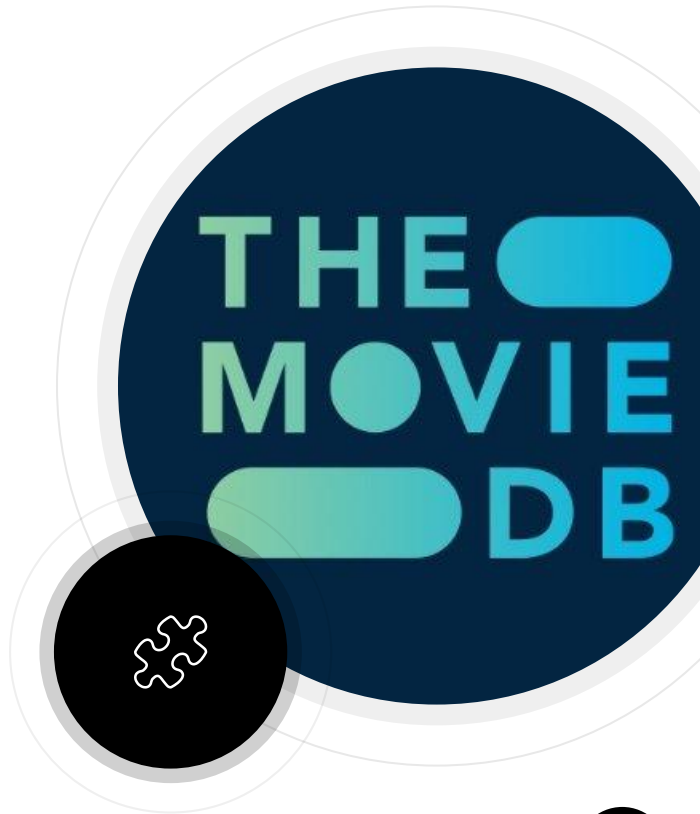
Présentation Dataset


Petit

- 5k données
- Champs : titre, synopsis, genres, poster, ...
- Nécessité de filtrer les données (plusieurs synopsis manquants)

Grand

- 50k données
- Champs : titre, synopsis, genres, poster, ...
- Nécessité de filtrer les données (7% données genres manquantes)



The background features a series of concentric circles in shades of gray. In the center, there is a light gray document icon with a folded corner. Overlaid on this icon is the title text in a bold, black, sans-serif font.

Filtrage des données

Filtrage des données

- Suppression des lignes sans synopsis ou genres

```
data.dropna(subset=['overview', 'genre'], inplace=True)
```

- Garder colonne intéressante

```
sorted_data = data[['original_title', 'overview', 'genre']]  
filtered_data = sorted_data[sorted_data['overview'].str.len() >  
10]
```

The background features a series of concentric circles in light gray. In the center, there is a stylized icon of a document with a folded corner, rendered in a light gray color. Overlaid on this is the main title text in a large, bold, black font.

Tokenisation & Lemmatisation

Tokenisation & lemmatisation

- **Tokenisation**: division d'un texte en unités plus petites appelées "tokens"
- **Lemmatisation**: La lemmatisation est le processus de réduction des mots à leur forme de base, appelée "lemme".





Tokenisation

```
tokens = word_tokenize(text)
```

AVANT	APRES
<pre>['As youngs and naives recruits in Vietnam']</pre>	<pre>['As', 'youngs', 'and', 'naives', 'recruits', 'in', 'Vietnam']</pre>

Lemmatisation

```
lemmatized_tokens = [lemmatizer.lemmatize(token.lower()) for token in tokens]
```

AVANT

```
['Twin', 'sisters', 'Emma',  
'and', 'Sam', 'come', 'up',  
'with', 'a', 'scheme', 'to',  
'switch', 'places']
```

APRES

```
['twin', 'sister', 'emma',  
'and', 'sam', 'come', 'up',  
'with', 'a', 'scheme', 'to',  
'switch', 'place']
```



Entraînement du modèle



Entraînement du modèle

1. Divisions des données en ensemble d'entraînement et test

```
X_train, X_test, y_train, y_test =  
train_test_split(filtered_data['overview'], filtered_data['genre'],  
test_size=0.2, random_state=42)
```



Entraînement du modèle

2. Entraînement du modèle Word2Vec

```
model = Word2Vec(sentences=train_tokens, vector_size=100, window=5,  
min_count=1, workers=4)
```

Train_tokens = ensemble de données qui est utilisée pour entraîner le modèle sur lequel on a appliqué le prétraitement (tokenisation et lemmatisation)

Entraînement du modèle

3. Transformation des données (entraînement + test) en embeddings

```
X_train_embeddings = np.array([np.mean([model.wv[word] for word in doc if word in model.wv] or [np.zeros(model.vector_size)], axis=0) for doc in train_tokens])
```

C'est ce qui va permettre ensuite aux modèles d'apprentissage de mieux comprendre la structure des données

C'est X_train_embeddings qui sera ensuite utilisé comme entrée pour le modèle

Entraînement du modèle

3. Transformation des données (entraînement + test) en embeddings

```
X_train_embeddings = np.array([np.mean([model.wv[word] for word in doc if word in model.wv] or [np.zeros(model.vector_size)], axis=0) for doc in train_tokens])
```

C'est ce qui va permettre ensuite aux modèles d'apprentissage de mieux comprendre la structure des données

C'est X_train_embeddings qui sera ensuite utilisé comme entrée pour le modèle

Entraînement du modèle

4. Entraîner le modèle de classification sur l'ensemble d'entraînement

```
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train_embeddings, y_train)
```

- Création d'une instance de RandomForestClassifier avec 100 arbres de décision et une graine aléatoire (42) -> Assurer la reproductibilité des résultats
- Entraînement du modèle (fit) avec les données d'entraînements sous formes d'embeddings

Entraînement du modèle

5. Prédictions sur ensemble de test

```
y_pred = clf.predict(X_test_embeddings)
```

Une fois le modèle entraîné (avec RandomForestClassifier), il tente de retrouver le genre des films du dataset Test :

- Y_test : données réelles
- Y_pred : données prédites

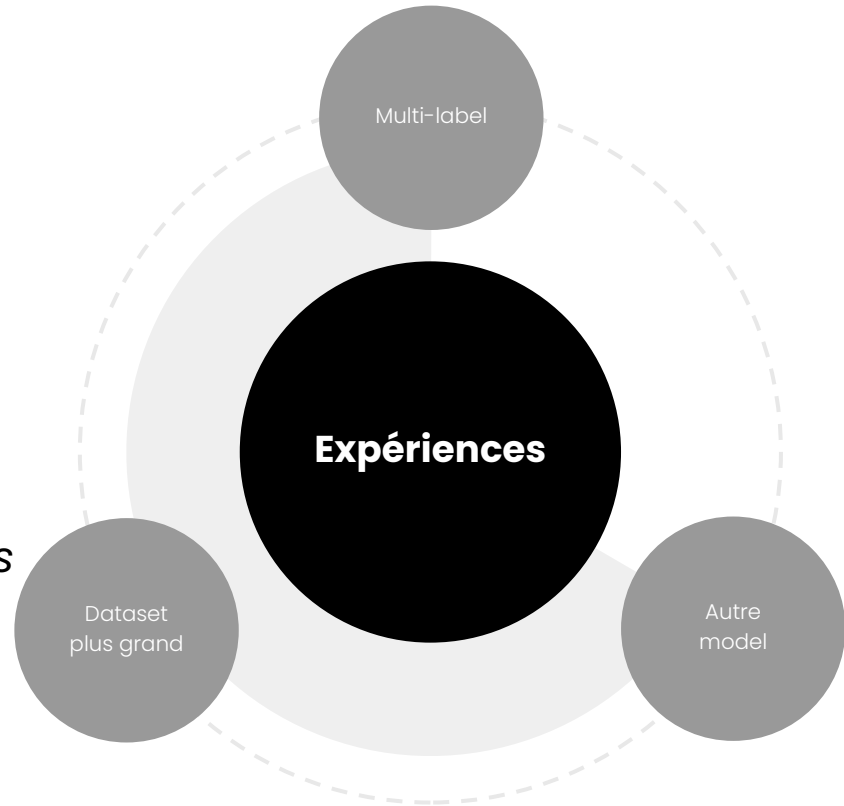
C'est grâce à ceci que l'on peut étudier la performance du modèle. (rapport et matrice de confusion)



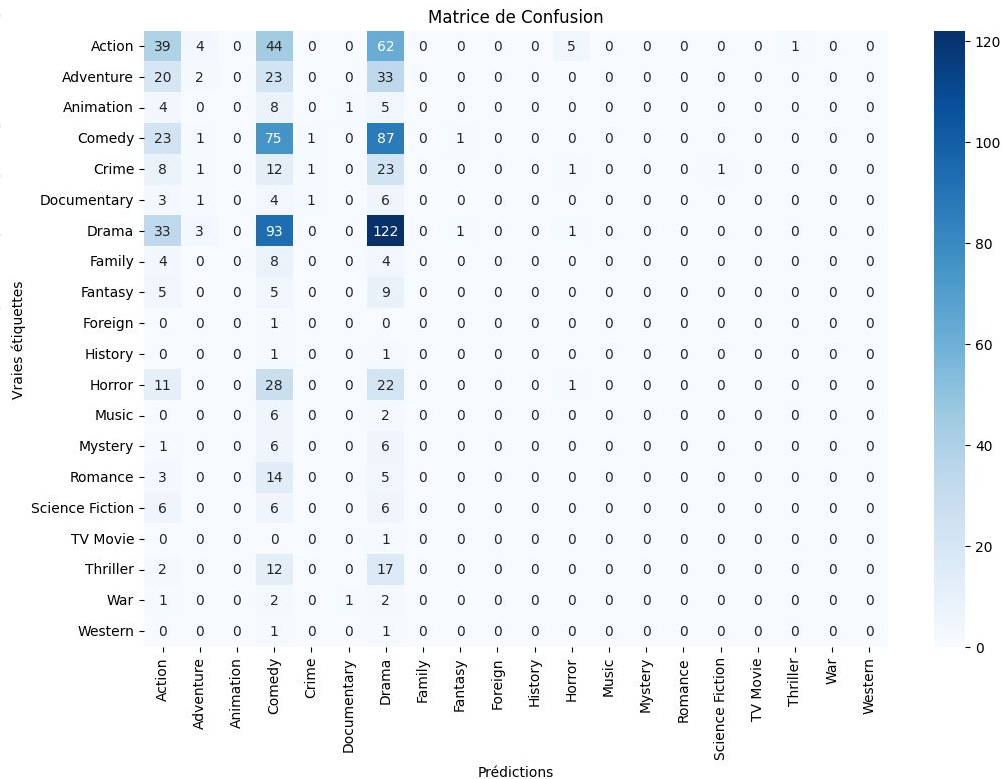
Expériences

Expériences réalisées

- *Utilisation de matrices de confusions pour modéliser les résultats*



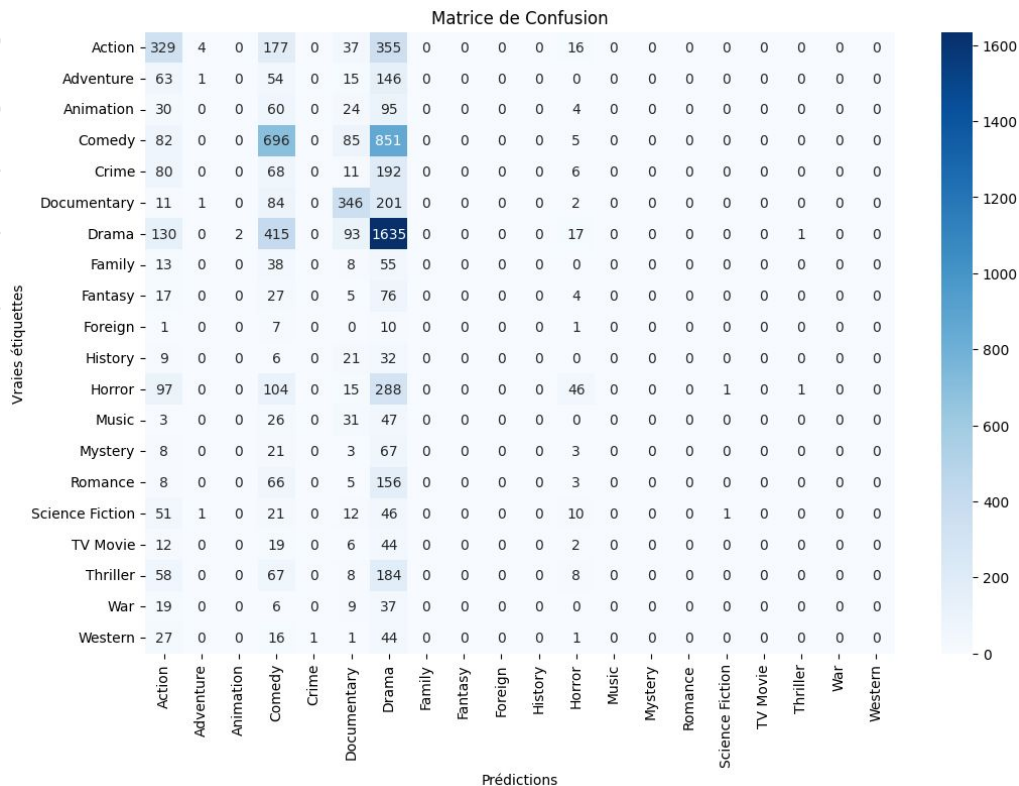
Expérience 1 :



Rapport de classification :

	precision	recall	f1-score	support
Action	0.24	0.25	0.25	155
Adventure	0.17	0.03	0.04	78
Animation	0.00	0.00	0.00	18
Comedy	0.21	0.40	0.28	188
Crime	0.33	0.02	0.04	47
Documentary	0.00	0.00	0.00	15
Drama	0.29	0.48	0.37	253
Family	0.00	0.00	0.00	16
Fantasy	0.00	0.00	0.00	19
Foreign	0.00	0.00	0.00	1
History	0.00	0.00	0.00	2
Horror	0.12	0.02	0.03	62
Music	0.00	0.00	0.00	8
Mystery	0.00	0.00	0.00	13
Romance	0.00	0.00	0.00	22
Science Fiction	0.00	0.00	0.00	18
TV Movie	0.00	0.00	0.00	1
Thriller	0.00	0.00	0.00	31
War	0.00	0.00	0.00	6
Western	0.00	0.00	0.00	2
accuracy			0.25	955
macro avg	0.07	0.06	0.05	955
weighted avg	0.20	0.25	0.20	955

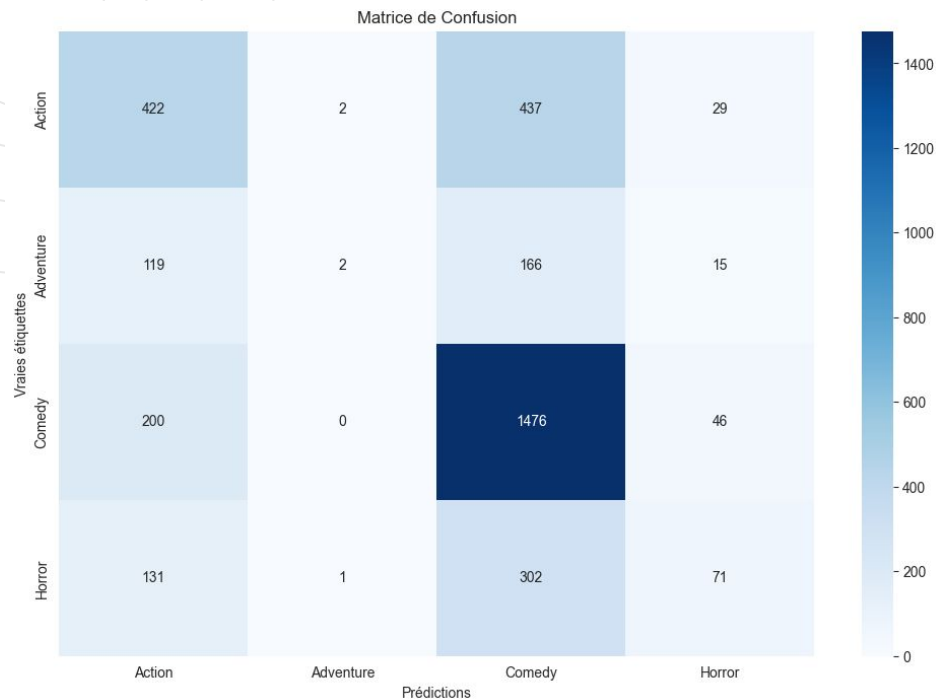
Expérience 2 : Dataset plus grand



Rapport de classification :

	precision	recall	f1-score	support
Action	0.31	0.36	0.33	918
Adventure	0.14	0.00	0.01	279
Animation	0.00	0.00	0.00	213
Comedy	0.35	0.40	0.38	1719
Crime	0.00	0.00	0.00	357
Documentary	0.47	0.54	0.50	645
Drama	0.36	0.71	0.48	2293
Family	0.00	0.00	0.00	114
Fantasy	0.00	0.00	0.00	129
Foreign	0.00	0.00	0.00	19
History	0.00	0.00	0.00	68
Horror	0.36	0.08	0.14	552
Music	0.00	0.00	0.00	107
Mystery	0.00	0.00	0.00	102
Romance	0.00	0.00	0.00	238
Science Fiction	0.50	0.01	0.01	142
TV Movie	0.00	0.00	0.00	83
Thriller	0.00	0.00	0.00	325
War	0.00	0.00	0.00	71
Western	0.00	0.00	0.00	90
accuracy			0.36	8464
macro avg	0.12	0.11	0.09	8464
weighted avg	0.28	0.36	0.29	8464

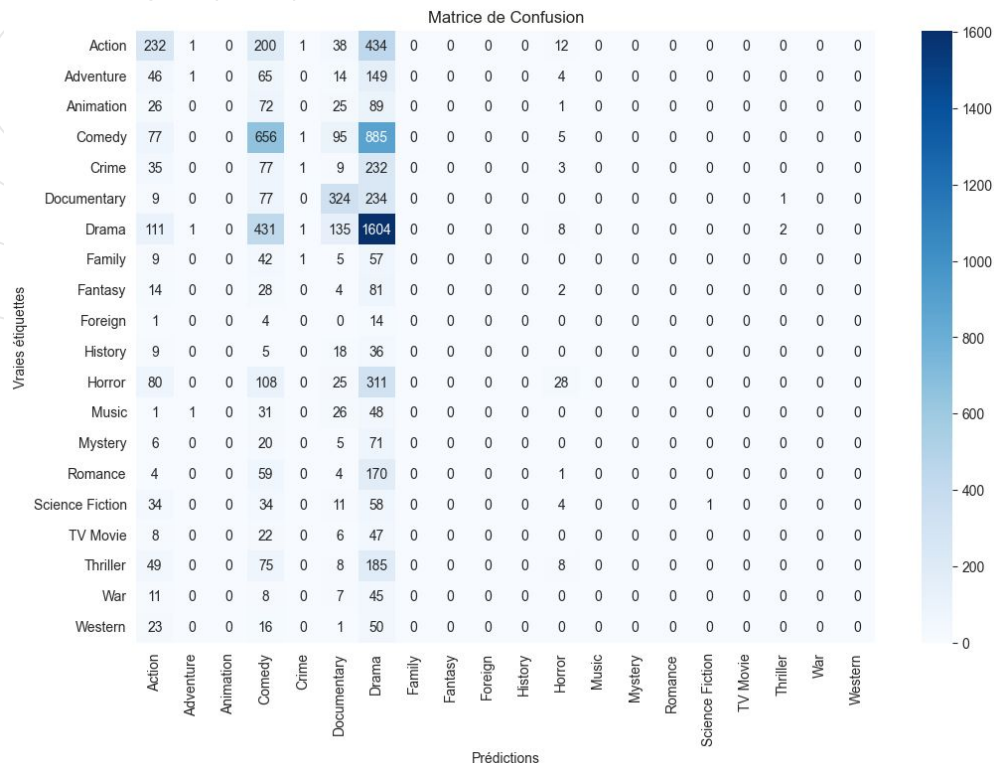
Expérience 3 : exclusion des classes faibles



Rapport de classification :

	precision	recall	f1-score	support
Action	0.48	0.47	0.48	890
Adventure	0.40	0.01	0.01	302
Comedy	0.62	0.86	0.72	1722
Horror	0.44	0.14	0.21	505
accuracy			0.58	3419
macro avg	0.49	0.37	0.36	3419
weighted avg	0.54	0.58	0.52	3419

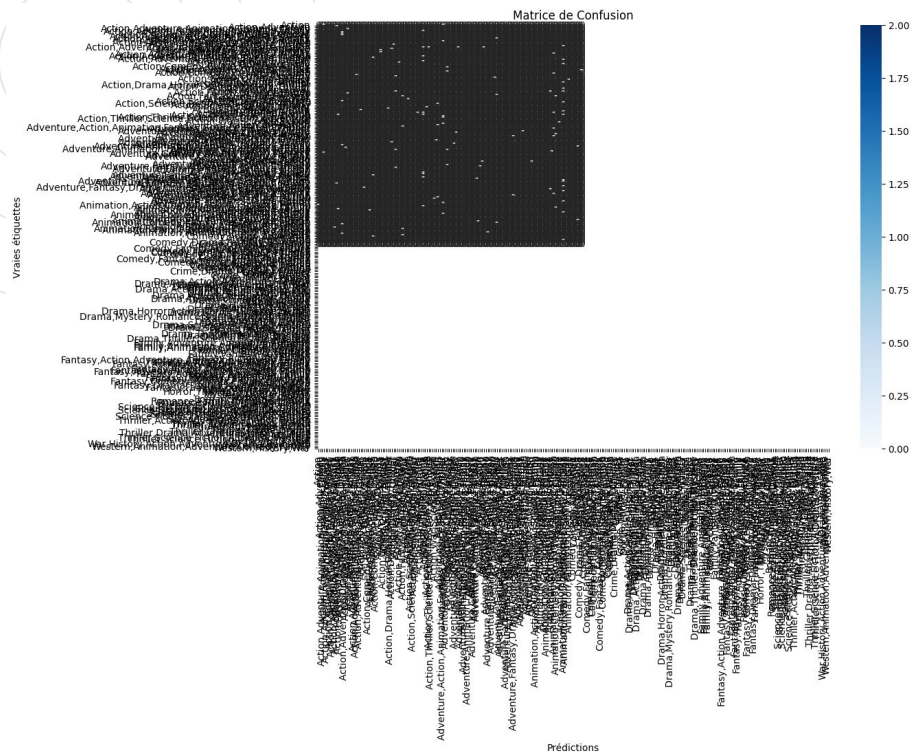
Expérience 4 : test du model FastText



Rapport de classification :

	precision	recall	f1-score	support
Action	0.30	0.25	0.27	918
Adventure	0.25	0.00	0.01	279
Animation	0.00	0.00	0.00	213
Comedy	0.32	0.38	0.35	1719
Crime	0.20	0.00	0.01	357
Documentary	0.43	0.50	0.46	645
Drama	0.33	0.70	0.45	2293
Family	0.00	0.00	0.00	114
Fantasy	0.00	0.00	0.00	129
Foreign	0.00	0.00	0.00	19
History	0.00	0.00	0.00	68
Horror	0.37	0.05	0.09	552
Music	0.00	0.00	0.00	107
Mystery	0.00	0.00	0.00	102
Romance	0.00	0.00	0.00	238
Science Fiction	1.00	0.01	0.01	142
TV Movie	0.00	0.00	0.00	83
Thriller	0.00	0.00	0.00	325
War	0.00	0.00	0.00	71
Western	0.00	0.00	0.00	90

Expérience 5 : tentative de multiclasse



Rapport de classification :

	precision	recall	f1-score	support
Action,Adventure	0.00	0.00	0.00	1
Action,Adventure,Animation,Family	0.00	0.00	0.00	0
Action,Adventure,Comedy,Drama,Mystery	0.00	0.00	0.00	1
Action,Adventure,Comedy,Science Fiction	0.00	0.00	0.00	1
Action,Adventure,Comedy,Science Fiction,Western	0.00	0.00	0.00	1
Action,Adventure,Crime	0.00	0.00	0.00	1
Action,Adventure,Crime,Fantasy,Science Fiction	0.00	0.00	0.00	1
Action,Adventure,Drama	0.00	0.00	0.00	1
Action,Adventure,Family,Fantasy	0.00	0.00	0.00	0
Action,Adventure,Fantasy	0.33	0.50	0.40	2
Action,Adventure,Fantasy,Science Fiction	0.00	0.00	0.00	2
Action,Adventure,Fantasy,Thriller	0.00	0.00	0.00	1
Action,Adventure,Science Fiction	0.00	0.00	0.00	0
Action,Adventure,Thriller	0.00	0.00	0.00	0
Action,Comedy,Drama,Thriller	0.00	0.00	0.00	1
Action,Comedy,Science Fiction	0.00	0.00	0.00	1
Action,Comedy,Thriller	0.00	0.00	0.00	1
Action,Crime,Drama,Thriller	0.00	0.00	0.00	0
Action,Crime,Fantasy	0.00	0.00	0.00	1
Action,Crime,Science Fiction,Thriller	0.00	0.00	0.00	1
Action,Drama,Adventure	0.00	0.00	0.00	0
Action,Drama,Mystery,Thriller	0.00	0.00	0.00	1

Prédiction avec données utilisateur

"A spaceship travels to distant galaxies."	"A romantic drama set in Paris."
<code>\Comedy'</code>	<code>\Drama'</code>
<code>['Adventure', 'Drama', 'Actions']</code>	<code>['Action', 'Adventure', 'Science Fiction']</code>

Conclusion



1 Multi-label essentiel

2 Sous classes

3 Dataset plus important