



RNCP35288 CDSD

# AT&T Spam Detector

## Deep Learning

**Rédacteur :** Jean-Yves Vuillequez





## Contexte et enjeu

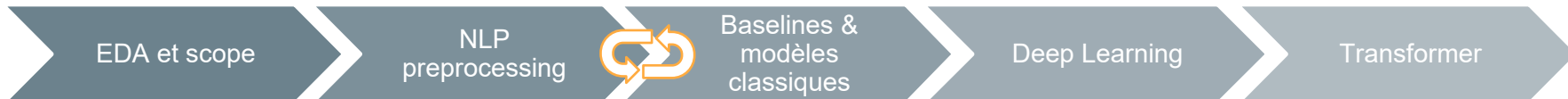


### Objectif :

- Mettre en place un classificateur de SMS capable de détecter automatiquement les messages spam afin de réduire le risque **sans bloquer les messages légitimes**.



# Méthodologie du projet



## Objectif :

- Comprendre la nature des et les enjeux de la détection de spam

## Actions :

- Analyse de la distribution **ham** / **spam**
- Analyse lexicale
- Analyse sémantique & sentiment spam)
- Identification des contraintes

## Objectif :

- Transformer le texte brut en représentations exploitables par les modèles

## Actions :

- Nettoyage et normalisation du texte + Tokenisation

## Deux approches :

- TF-IDF (baseline ML interprétable)
- Embedding (CNN / RNN / Transformers)

## Objectif :

- Établir des points de comparaison

## Actions :

- Modèle naïf (random stratifié)
- TF-IDF + Logistic Regression
- Évaluation via Precision / Recall / F1 / PR-AUC Analyse des erreurs (confusion matrix)

## Objectif :

- Évaluer l'apport des réseaux neuronaux sur données textuelles

## Actions :

- CNN : détection de motifs locaux (n-grams)
- GRU (RNN) : prise en compte de la séquence
- Suivi des métriques par epoch
- Comparaison avec les baselines ML

## Objectif :

- Maximiser la performance de détection du spam

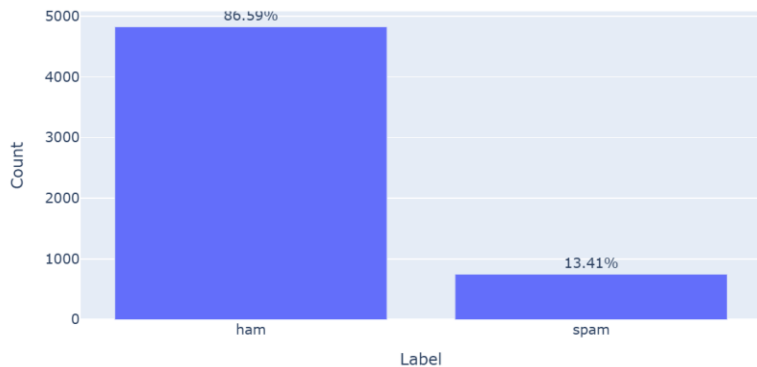
## Actions :

- Fine-tuning de DistilBERT
- Exploitation du contexte sémantique complet
- Évaluation finale et comparaison globale

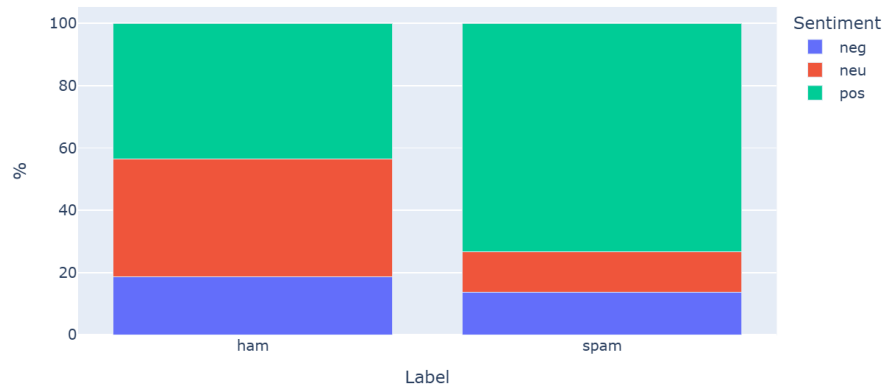


# Scope sémantique 1/2

Equilibre des classes (ham vs spam)



Répartition du sentiment (%) — spam vs ham



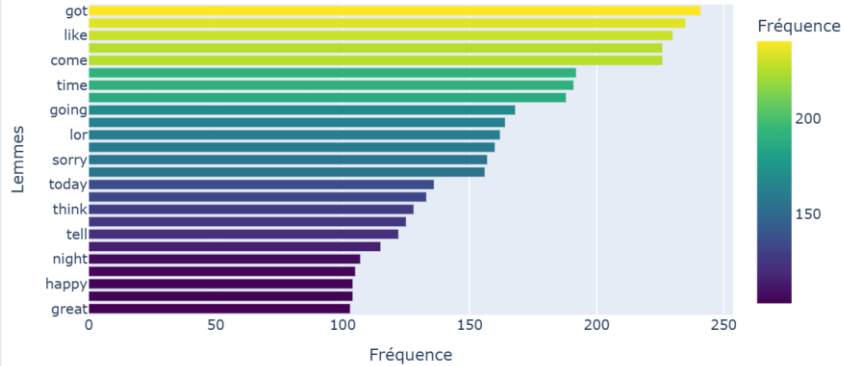
## Analyse :

- Dataset fortement déséquilibré : **86,6 % ham vs 13,4 % spam**
- Le spam présente un **biais émotionnel positif** marqué vs le ham est **plus neutre / négatif**
- Le vocabulaire des spams est **différent** de celui des messages normaux, ce qui permet au modèle d'apprendre à les distinguer

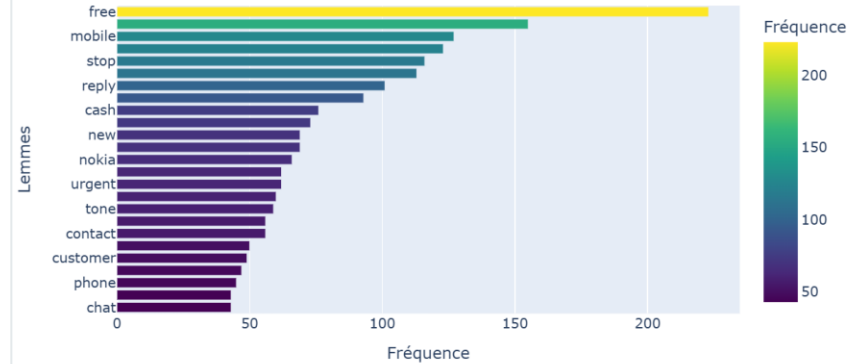


## Scope sémantique 2/2

Top 25 lemmes les plus fréquents — ham



Top 25 lemmes les plus fréquents — spam



### Analyse :

- Le vocabulaire ham est **conversationnel et contextuel** : temps, verbes d'intentions, relations
- Le vocabulaire spam est **transactionnel et incitatif** : gratuit, urgence, verbes d'actions, technique
- Il existe une classification bien marquée entre Ham et Spam avec des champs lexicaux très identifiables



# Résultats : Baseline vs Deep Learning vs Transformer

**F1-score spam** comme métrique principale (classes déséquilibrées), complété par **précision**, **rappel** et **PR-AUC** pour analyser les compromis faux positifs / faux négatifs et la qualité du ranking

Modèle	precision_spam	Recall_spam	f1_spam	pr_auc	Diagnostic
TF-IDF + LogReg	0.958	0.920	0.938	0.974	Baseline très solide, simple et robuste
SimpleCNN (PyTorch)	0.932	0.913	0.922	0.962	Capture de motifs locaux, perfs proches de la baseline
SimpleGRU (PyTorch)	0.929	0.873	0.900	0.958	Modélise la séquence, mais moins stable
DistilBERT (PyTorch)	0.979	0.953	0.966	0.989	Meilleure performance globale, mais coût élevé

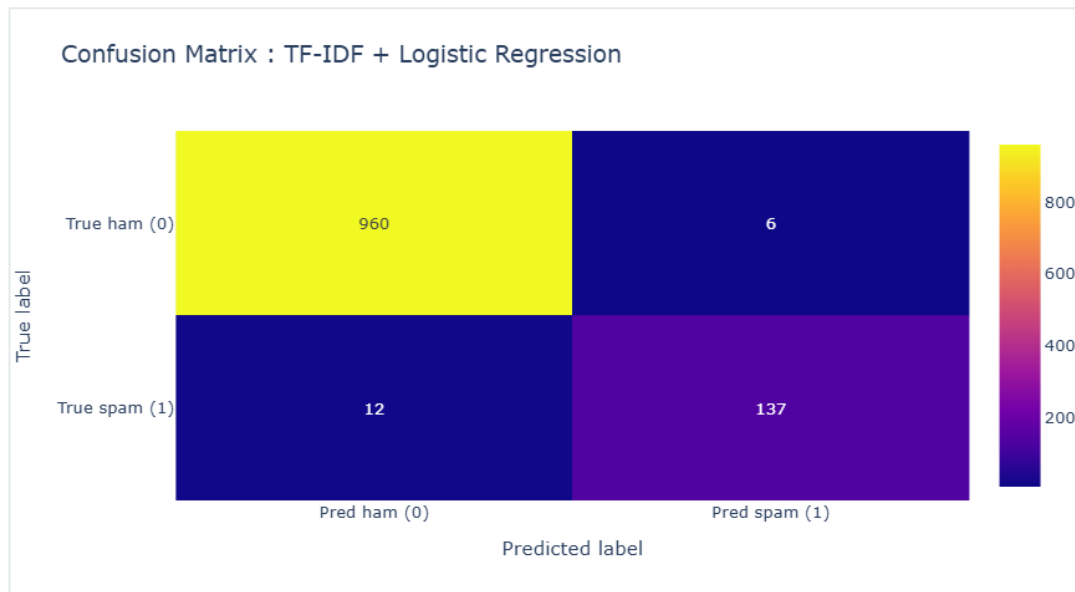


Résultats en phase avec « l'état de l'art » : [Paper 129-Optimized\\_SMS\\_Spam\\_Detection.pdf](#)



# Résultats Comparaisons

	Ham	Spam
True 0	TN	FP
True 1	FN	TP



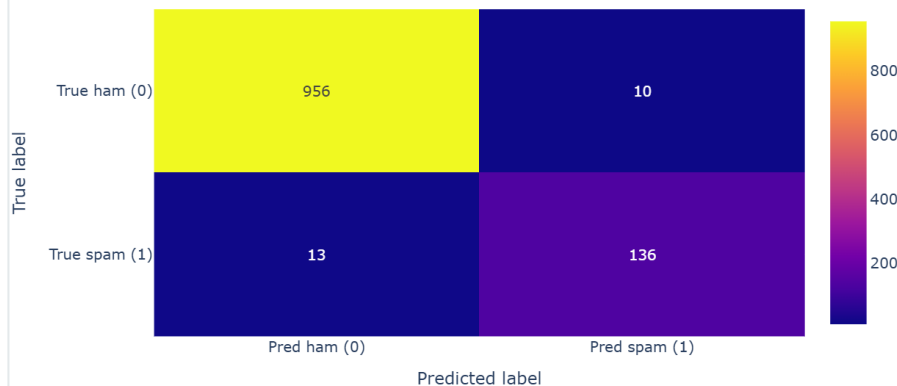
- Très bon compromis précision / rappel : faible taux de faux positifs (6) et de faux négatifs (12)
- Baseline robuste et déjà performante pour un modèle linéaire
- Sert de référence solide pour mesurer l'apport de modèles plus complexes



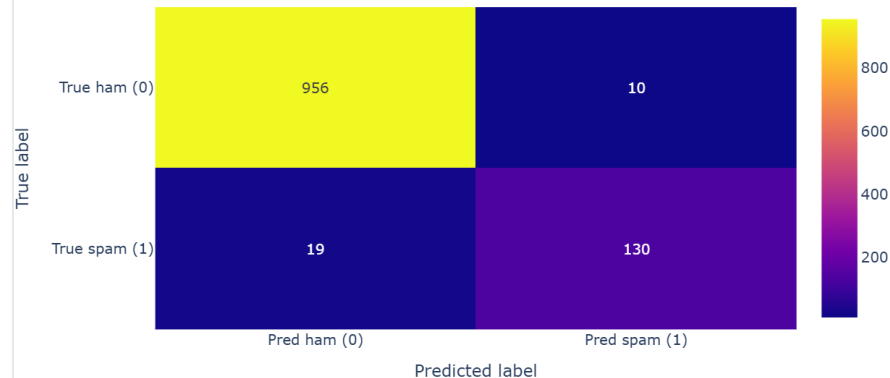
# Résultats Comparaisons

	Ham	Spam
True 0	TN	FP
True 1	FN	TP

Confusion Matrix : SimpleCNN (PyTorch)



Confusion Matrix : SimpleRNN (PyTorch)



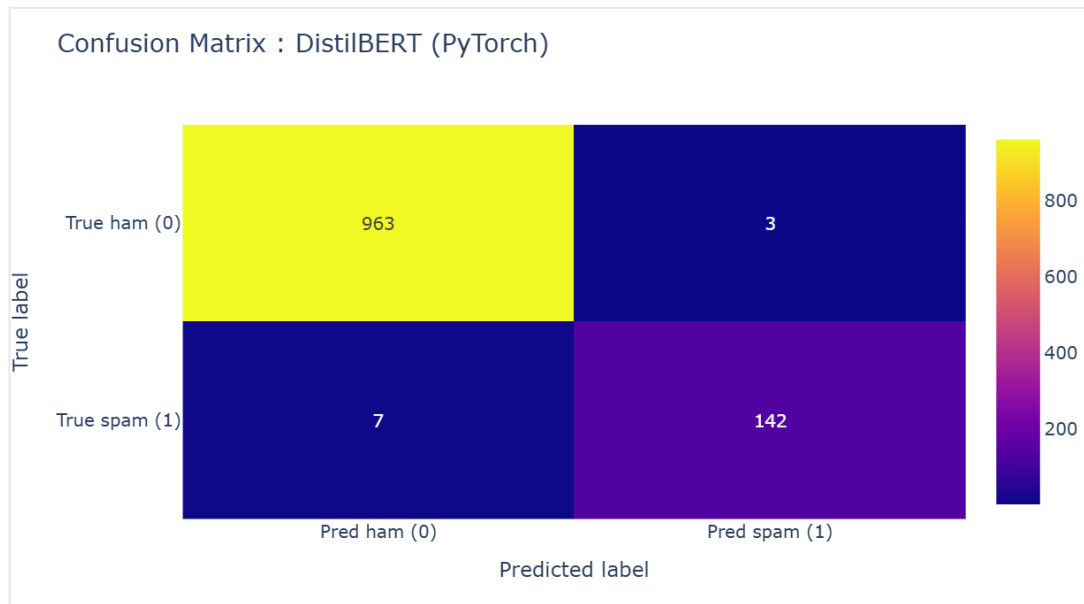
- 956 vrais messages ham correctement classés comme ham, **136 / 130 spams correctement détectés**
- **Forte réduction des faux négatifs** par rapport à la baseline





# Résultats Comparaisons

	Ham	Spam
True 0	TN	FP
True 1	FN	TP



- 963 messages ham correctement classés comme ham, **142 spams correctement détectés**
- Meilleure performance globale : moins de faux positifs et faux négatifs
- Modèle le plus robuste, **mais coût de calcul et complexité plus élevés**



Jedha

Merci pour votre attention  
Des questions ?

