

YNOV 2024-2025
Mastère 2 Data Science
Matière Traitement du langage (NLP)

Projet

Analyse Financière en utilisant le NLP

Présenté :

SAID ALI OSMAN - NAZIR YOUSOUF YAYE

Table des matières

Figures	4
Tableaux	4
Résumé	5
Introduction	6
1- Etat de l'art	8
2- Analyse de sentiment	9
2.1- Objectif de l'analyse de sentiment	9
2.2 Jeux de données utilisés	9
2.2.1 Twitter Financial News Sentiment (Zeroshot)	9
2.2.2 FinGPT - Sentiment Training Dataset	10
2.2.3 Dataset fusionné	11
2.3 Prétraitement des données de l'analyse de sentiment	12
2.4 Modèles testés	13
2.4.1 Méthodes classiques	13
2.4.2 Méthodes avancées (Transformers)	13
2.5 Comparaison des deux scénarios	14
2.5.1 Modèles classiques scénario A (avec stop-words)	14
2.5.2 Modèles classiques scénario B (sans stop-words)	16
2.5.3 Modèles Transformers scénario A (avec stop-words)	17
2.5.4 Interprétation	18
2.6 Courbes ROC et analyse des erreurs	18
2.6.1 Courbes ROC	18
2.6.2 SHAP (XGBoost)	20
2.6.3 LIME (Voting-Classifier)	20
2.6.4 Statistiques d'erreurs	21
2.6.5 Matrices de confusion	22
2.7 Recommandation	23
3- Reconnaissance des entités nommées (NER)	24
3.1- Objectif de la tâche de NER	24
3.1.1 Dataset fusionné (OntoNotes et FiNER-ORD)	24
3.2 Baseline CRF	26
3.3 Modèles Transformers fine-tunés	26
3.4 Matrices de confusion	27
3.5 Exemple d'application	27

3.6 Recommandations NER	28
4- Extraction de relations	29
4.1 Objectif de l'extraction de relations	29
4.2 Données pour l'extraction des relations	30
4.3 Prétraitement des données de l'extraction de relations.....	31
4.4 Découpage de données.....	31
4.5 Modèles testés	32
4.5.1 Modèles d'apprentissage automatique	32
4.5.2 Modèles pré-entraîné	32
4.6 Performances pour l'extraction de relations	33
5- Résultats et Discussions	35
6- Conclusion et Perspective.....	36
Bibliographies	37

Figures

Figure 1 : Distribution des classes dans les données (Twitter) d'analyse des sentiments.	9
Figure 2 : Distribution des 9 classes dans les données (FinGPT) d'analyse des sentiments.	10
Figure 3 : Distribution des classes simplifiées dans les données (FinGPT) d'analyse des sentiments	11
Figure 4 : Distribution des classes simplifiées dans les données (Twitter+FinGPT) d'analyse des sentiments.....	11
Figure 5 : Courbes ROC du modèle Voting sur la tâche d'analyse de sentiments avec SW	19
Figure 6 : Courbes ROC du modèle Voting sur la tâche d'analyse de sentiments sans SW	19
Figure 7 : Matrices de confusions données sans stop-word	22
Figure 8 : Matrices de confusion avec stop-words.....	22
Figure 9 : Distribution des tags dans OntoNotes-5	24
Figure 10 : Distribution tags-FiNER-ORD.....	25
Figure 11 : Distribution tags - FiNER-ORD (oversamplé)	25
Figure 12 : Exemple d'extraction de relation (Zhao et al. 2024).....	29
Figure 13 : Distribution des labels dans les données pour la tâche de l'extraction de relations	30
Figure 14 : Séparation des données en données d'entraînement, de validation et de test...	31
Figure 15 Performances des différents modèles sur la tâche de l'extraction de relation	34

Tableaux

Tableau 1 : Performances par classe sur les données d'analyse de sentiments (Avec stop- words).....	14
Tableau 2 : Performances globales des modèles classiques sur la tâche de l'analyse de sentiments (Avec stop-words).....	15
Tableau 3 : Performances par classe sur les données d'analyse de sentiments (Sans stop- words).....	16
Tableau 4 : Performances globales des modèles classiques sur la tâche de l'analyse de sentiments (Sans stop-words).....	17
Tableau 5 : Performances globales des modèles avancés(transformers) sur la tâche de l'analyse de sentiments (Avec stop-words).....	17
Tableau 6 : Performances globales des modèles avancés(transformers) sur la tâche de l'analyse de sentiments (Avec stop-words).....	18
Tableau 7 : Performances baseline CRF (tâche NER)	26
Tableau 8 : Performances globales modèles avancés sur la tâche NER.....	26
Tableau 9 : Matrices de confusion pour la tâche NER du modèle roberta	27
Tableau 10 : Un Exemple de test NER sur un modèle roberta fine tuné sur un dataset unifié	27
Tableau 11 : Un Exemple de test NER sur un modèle roberta fine tuné sur un seulement dataset OntoNotes	28
Tableau 12 : Configuration des paramètres pour les modèles pré-entraîné.....	32
Tableau 13 : Performances des différents modèles sur la tâche de l'extraction de relation .	33

Résumé

Mots-clés : traitement automatique de langue, analyse de sentiment, reconnaissance des entités nommées, extraction de relation, Q&A, modèle support vecteur machine et RandomForestClassifier, xgboost, lightGBM, voting ensemble, modèle BERT et llama 3, hyperparamètres (gridsearchcv).

Dans un contexte marqué par la volatilité constante des marchés financiers, les investisseurs ainsi que les institutions financières s'efforcent de mieux comprendre les dynamiques du marché en analysant les cours boursiers, en détectant l'évolution du sentiment du marché, et en identifiant les relations entre les entités économiques. À cette fin, l'exploitation des données textuelles financières telles que les articles de presse, les rapports de résultats ou encore les publications sur les réseaux sociaux constitue une source d'information précieuse. Ces textes permettent non seulement d'extraire automatiquement des informations pertinentes, notamment via la reconnaissance d'entités nommées, mais aussi de développer des agents conversationnels pour répondre à des requêtes spécifiques. Parmi les principaux enjeux figurent l'analyse des sentiments exprimés, l'identification d'entités propres au domaine financier, ainsi que l'extraction de relations financières significatives.

Dans cette étude, nous avons développé de modèles machine Learning classique comme le support vecteur machine (SVM) et le RandomForestClassifier (RF), XGBoost, LightGBM, Voting Ensemble, et aussi utilisé de modèles pré-entraîné comme BERT acronyme en anglais (Bidirectional Encoder Representations from Transformer), llama 3 de méta. Nous avons comparé ces modèles entre eux pour n'utiliser que le meilleur modèle pour chaque tâche dans notre application. Pour y arriver, il est important de suivre les étapes classiques d'un projet TAL, telle que le prétraitement de données (suppression de ponctuations, de stopword, de liens https etc..., la vectorisation de données textuelles), la séparation de données en données d'entraînement, de validation et de test, ainsi que le développement du modèle et la recherche des meilleurs hyperparamètres avec GridSearchcv.

Chaque modèle a été évalué sur sa capacité à effectuer des classifications précises sur nos données de validation et de test. Pour ces évaluations, nous avons utilisé trois métriques appropriées au problème de classification qui sont à savoir : la précision, la moyenne macro F1-score, le RECALL.

Nos résultats montrent que la famille de modèles bert et llama 3 offrent une performance supérieure par rapport aux autres modèles. Ces approches de modèle fine-tuner démontrent une précision de généralisation accrue sur des données test.

Introduction

La volatilité des marchés financiers oblige les investisseurs et les institutions à revoir leurs stratégies d'analyse pour mieux anticiper les évolutions économiques.

Un bon matin d'avril 2025, un simple message posté par le président américain Donald Trump bouleverse Wall Street. Ce message, posté sur le réseau social X (ancien Twitter), évoque une possible suspension des droits de douane (*"Thinking about pausing those tariffs"*). En quelques minutes, plusieurs actions grimpent de 3%, les investisseurs attendent avec impatience cette possible trêve commerciale. Deux jours plus tard, le président revient sur ces propos devant la presse (*"We are not looking at that"*). Le marché efface alors tous ses gains (Allyn et Bobby. 2025). Pire encore, une déclaration officielle de la maison blanche contredit les propos du président, entraînant juste après une nouvelle chute (Lawder, David. 2025). En moins de 72 heures, c'est environ vingt milliards de dollars de capitalisation qui se sont envolés, puis sont revenus, à cause d'une série de messages (tweets) et de conférences de presse contradictoires (France 24, 2025).

Avec un certain recul, ces montagnes russes soulignent une évidence : la valeur d'une entreprise ne reflète plus seulement ses résultats financiers ou son activité réelle. Elle évolue en continu, sous l'influence de signaux textuels diffusés massivement, captée par un lot de texte (communiqués officiels, rapports d'analystes, publications sur les réseaux sociaux etc..). Chaque minute ajoute une couche de bruit, mais aussi des petits signaux importants. De la nuance dans le discours de grande personnalité, vers la mention de rachat ou un simple tweet, peut faire bondir ou plonger une action avant même que le marché n'ouvre vraiment.

C'est là qu'intervient le traitement automatique de langage (TAL/NLP). L'objectif de cette étude est d'explorer et d'évaluer le potentiel des techniques modernes de TAL pour la compréhension et surtout l'exploitation des données textuelles en contexte financier. Notre projet vise à démontrer, par l'expérimentation que cette technique peut constituer un levier d'aide à la décision, en contribuant à quatre axes clés de l'analyse de marché :

- L'extraction de sentiment : afin de déterminer automatiquement l'orientation émotionnelle de l'énoncé (positif, neutre, négatif), à partir de textes issus de réseaux sociaux, médias ou de document institutionnels.
- La reconnaissance d'entités nommées (NER) : pour identifier les noms des organisations, des acteurs, des événements majeurs mentionnés dans le texte.
- L'extraction de relations : mettre en évidence des liens explicites ou implicites entre entités.
- Requêtes ciblées (Q&A) : pour formuler des réponses précises à des questions de type factuel.

Pour répondre à ces objectifs, nous mettons en œuvre une double stratégie. D'un côté, nous mobilisons des approches classiques de l'apprentissage automatique fondées sur des représentations vectorielles dites simples (TF-IDF) et des classifieurs standards (SVM, Random Forest). De l'autre, nous exploitons les récents progrès des modèles de langage de

grande taille (LLMs), tels que BERT ou Llama 3, intégrant des techniques de fine-tuning adaptées (LoRA, recherche d'hyper paramètres).

Notre démarche adopte une approche expérimentale comparative, structurée en plusieurs étapes : construction de jeux de données annotés, normalisation linguistique, vectorisation, entraînement supervisé et analyse qualitative des erreurs.

Au-delà de la démonstration technique, ce travail ambitionne de fournir une feuille de route pour la mise en œuvre de solutions NLP dans des environnements réels, en tenant compte des contraintes spécifiques du domaine : jargon professionnel, biais temporels, déséquilibre des classes, rapidité de traitement. L'enjeu n'est pas de prédire les marchés, mais plutôt de leur donner du sens linguistique, afin d'éclairer les décisions humaines.

1- Etat de l'art

L'application du traitement automatique de la langue (TAL) dans le secteur financier a connu un essor remarquable ces dernières années. Des études comme celle présentée dans les travaux (Arif 2023), illustrent comment le NLP révolutionne la gestion des risques et l'extraction d'insights à partir de vastes corpus de données textuelles. Ces approches permettent de transformer des données issues de rapports annuels, d'articles de presse et de publications sur les réseaux sociaux en indicateurs stratégiques, contribuant ainsi à une meilleure prise de décision dans un contexte de marché volatile.

Les récentes avancées dans ce domaine ont permis une progression remarquable, passant des approches statistiques traditionnelles à des méthodes plus sophistiquées basées sur l'apprentissage profond. Les ressources et travaux référencés (Olusegun et al.2024) démontrent l'efficacité de ces techniques pour évaluer avec précision les sentiments du marché et les tendances émergentes. Cette capacité à capter les nuances sentimentales constitue un avantage concurrentiel significatif dans un environnement où l'anticipation des mouvements de marché est primordiale pour les institutions.

Par ailleurs, l'émergence des modèles pré-entraînés et des large language models (LLM) a ouvert de nouvelles perspectives pour l'analyse contextuelle du langage financier, permettant d'aller au-delà des simples signaux numériques et d'appréhender la complexité des sentiments exprimés dans les textes (Yang, Liu, et Wang 2023).

Des projets concrets, comme FinGPT, illustrent la mise en œuvre de ces avancées dans des tâches spécifiques telles que la reconnaissance d'entités nommées, la question-réponse (Q&A) et l'extraction de relations financières (Mo et al. 2024). Ces applications démontrent que l'intégration des modèles de NLP dans le domaine financier permet non seulement d'automatiser l'extraction d'informations, mais aussi d'améliorer la précision des analyses en capturant des relations complexes entre les événements.

Enfin, l'analyse temporelle des données textuelles, indispensable pour comprendre l'évolution des tendances, a bénéficié de travaux spécialisés tels que ceux présentés dans l'ACL Anthology (Zhong, Sun, et Cambria 2017) et des outils comme Syntime (UzZaman et al.). Ces contributions, complétées par des initiatives telles que le repository TempEval-3 Platinum TimeML, offrent un cadre robuste pour l'annotation et l'interprétation des informations temporelles, enrichissant ainsi les analyses des dynamiques de marché.

En somme, l'état de l'art révèle une évolution rapide des méthodes de NLP appliquées aux services financiers, passant de techniques traditionnelles à des approches de plus en plus sophistiquées et contextuelles, capables de répondre aux exigences d'un environnement économique en perpétuelle mutation.

2- Analyse de sentiment

2.1- Objectif de l'analyse de sentiment

L'analyse de sentiment consiste à catégoriser des textes selon la tonalité émotionnelle qu'ils véhiculent. Dans le contexte financier, cette tâche revêt une importance particulière, car les anticipations du marché et les décisions d'investissement sont souvent influencées par des indices émotionnels contenus dans des titres de presse, des dépêches, ou des messages sur les réseaux sociaux. Par exemple, un tweet annonçant la faillite d'une entreprise ou un communiqué évoquant des résultats « supérieurs aux attentes » peuvent modifier en quelques secondes la perception des investisseurs.

Notre objectif ici est de comparer différentes approches, classiques et avancées, pour cette tâche, en mettant l'accent sur la robustesse, la précision, la capacité à généraliser, et la simplicité d'interprétation.

2.2 Jeux de données utilisés

2.2.1 Twitter Financial News Sentiment (Zeroshot)

Pour la tâche d'analyse de sentiment, nous utilisons un jeu de données issu de la plateforme **HuggingFace**. Plus précisément, il s'agit du jeu de données intitulé “zeroshot/twitter-news-sentiment”, se compose de 11 931 lignes et 2 colonnes. La première colonne contient le texte des tweets et des actualités financières, tandis que la seconde colonne correspond aux labels de sentiment, répartis en trois classes :

- **Positive (20.10%)**
- **Neutral (64.91%)**
- **Negative (14.99%)**

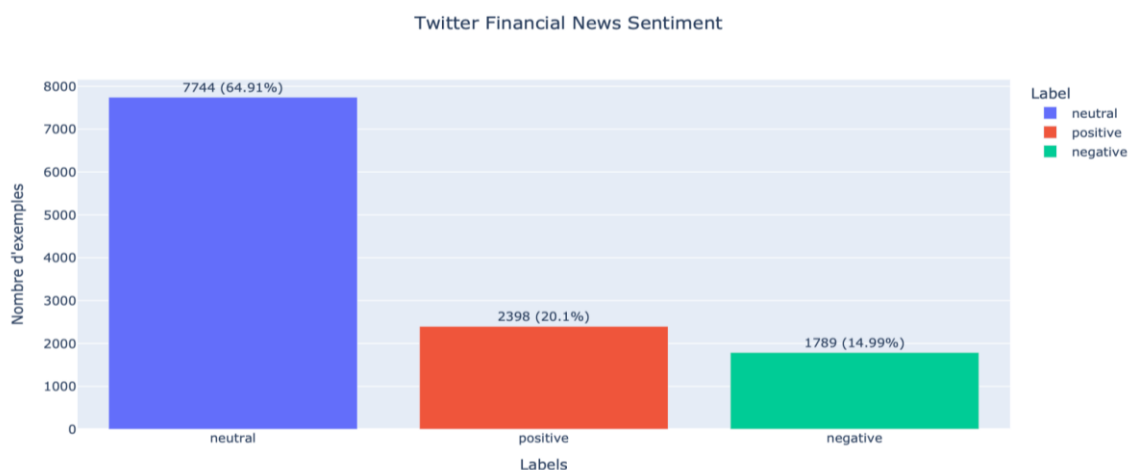


Figure 1 : Distribution des classes dans les données (Twitter) d'analyse des sentiments.

La distribution des étiquettes met en évidence un nombre majoritairement élevé de tweets neutres, suivi par des tweets positifs, puis négatifs. Cette répartition inégale des classes est un point d'attention important lors du développement et de l'évaluation des modèles, car elle peut influencer leur capacité à reconnaître correctement les tweets des classes minoritaires.

Ce dataset apporte une bonne couverture “à chaud” des marchés.

2.2.2 FinGPT - Sentiment Training Dataset

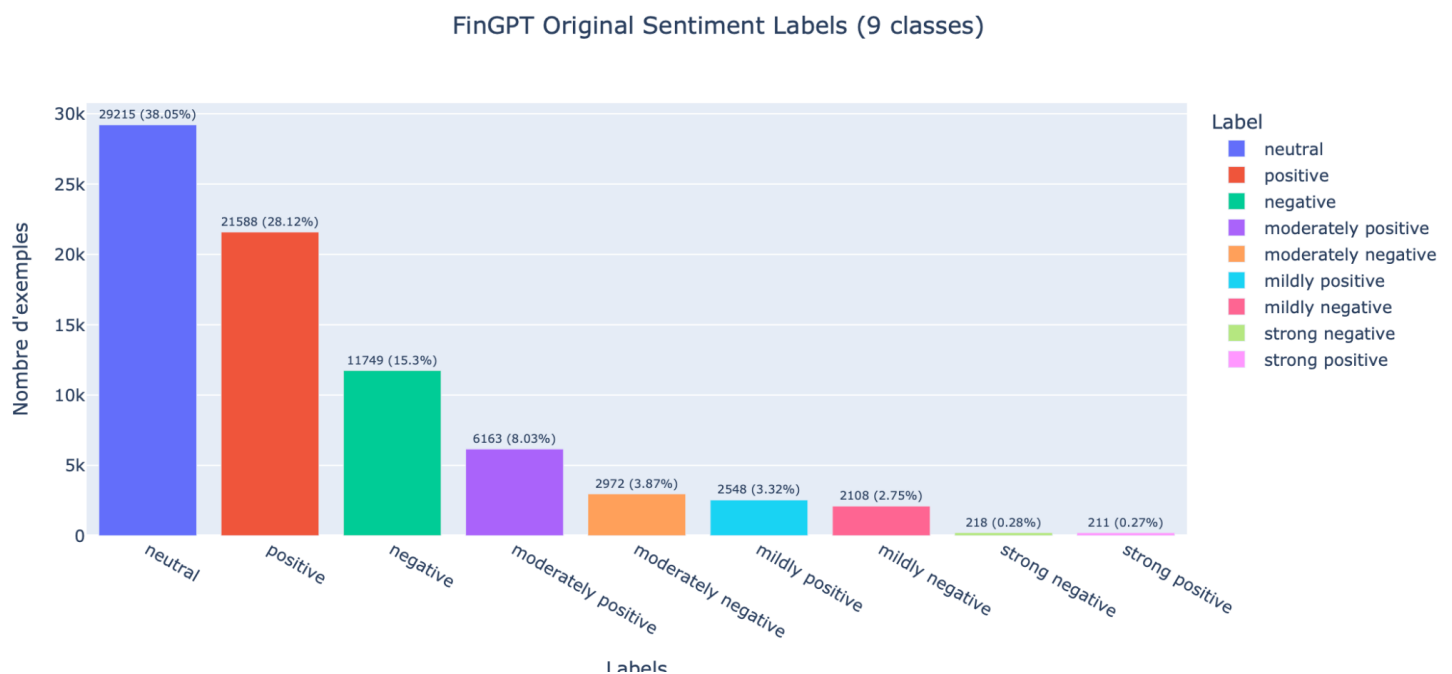


Figure 2 : Distribution des 9 classes dans les données (FinGPT) d'analyse des sentiments.

Le second corpus, plus conséquent, est fourni par le projet open source **FinGPT**. Que nous téléchargeons aussi depuis la plateforme **HuggingFace** Il rassemble des extraits de textes issus de dépêches ou de rapports financiers. Les annotations initiales incluent des labels plus fins (very positive, strongly negative, etc.).

L'intérêt de ce dataset allie qualité de rédaction plus formelle, meilleur équilibre entre classes, diversité des sources (presse, blogs, rapports d'entreprise).



Figure 3 : Distribution des classes simplifiées dans les données (FinGPT) d'analyse des sentiments

2.2.3 Dataset fusionné

Nous avons fusionné les deux jeux après harmonisation des colonnes et des étiquettes. Le corpus final comprend environ 86 000 exemples répartis ainsi :

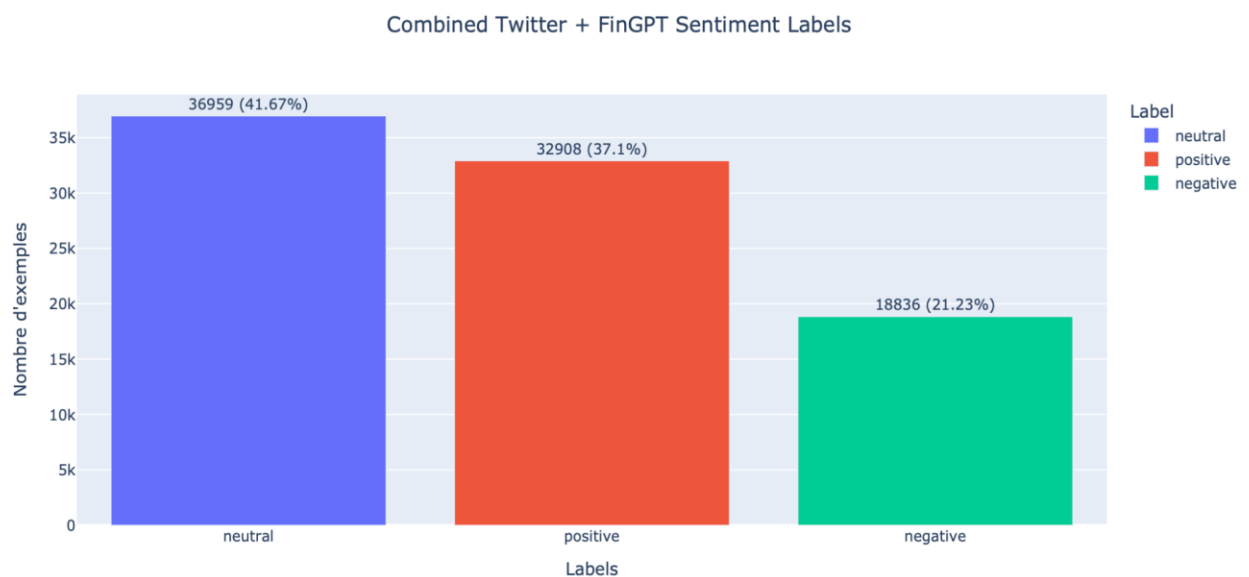


Figure 4 : Distribution des classes simplifiées dans les données (Twitter+FinGPT) d'analyse des sentiments.

La classe neutral reste encore légèrement sur-représentée, et negative sous-représentée, un déséquilibre que nous avons pris en compte dans la phase d'évaluation (utilisation de métriques **macro**).

2.3 Prétraitement des données de l'analyse de sentiment

Deux pipelines distincts ont été expérimentés :

2.3.1. Version brute (avec stop words)

- Les textes sont simplement nettoyés : suppression des liens, mentions (@), caractères spéciaux, passage en minuscules.
- Les mots vides (stop words comme "the", "is", "on") sont conservés pour préserver le contexte grammatical.

Hypothèse : dans des textes très courts (tweets notamment), les mots vides peuvent porter une information précieuse (négation, intensification).

2.3.2. Version filtrée (sans stop words)

- Les stop words généraux (issus de NLTK) et certains mots fréquents en finance (« shares », « stock », etc.) sont supprimés.
- La vectorisation repose alors sur un vocabulaire épuré, plus informatif.

Hypothèse : en supprimant les termes peu discriminants, on améliore la concentration du modèle sur les signaux forts (adjectifs, verbes).

Dans les deux cas, les textes ont été vectorisés à l'aide de **TF-IDF** (jusqu'à 10 000 n-grammes), et les labels encodés par **LabelEncoder**.

2.4 Modèles testés

2.4.1 Méthodes classiques

Nous avons évalué quatre familles d'algorithmes de classification supervisée :

- **SVM linéaire (LinearSVC)** : baseline robuste pour la classification textuelle.
- **Random Forest** : modèle d'ensemble insensible au bruit, performant sur des représentations TF-IDF.
- **XGBoost** : très populaire en compétition, optimisé pour la vitesse et la précision.
- **LightGBM** : proche de XGBoost mais plus rapide à l'entraînement.

Un **VotingClassifier** (moyenne pondérée des quatre modèles) a également été testé.

2.4.2 Méthodes avancées (Transformers)

Nous avons fine-tuné quatre modèles pré-entraînés :

- **ProsusAI/finbert** : spécialisé dans les textes financiers.
- **bert-base-uncased** : modèle généraliste BERT.
- **distilbert-base-uncased** : version allégée de BERT.
- **roberta-base** : alternative à BERT, réputée plus performante sur certains corpus.

Chaque modèle est entraîné 3 epochs avec early stopping et évalué sur la même partition test

2.5 Comparaison des deux scénarios

2.5.1 Modèles classiques scénario A (avec stop-words)

2.5.1.1 Performances par label

Tableau 1 : Performances par classe sur les données d'analyse de sentiments (Avec stop-words)

Modèle	Label	Précision	Rappel	F1-score
SVM (C=10)	negative	0.877	0.856	0.865
	neutral	0.910	0.937	0.923
	positive	0.901	0.886	0.893
Random Forest (n_estimators=100, max_depth=None)	negative	0.968	0.774	0.860
	neutral	0.920	0.964	0.941
	positive	0.877	0.931	0.903
XGBoost (n_estimators=200)	negative	0.897	0.708	0.792
	neutral	0.823	0.939	0.877
	positive	0.866	0.833	0.850
LightGBM (n_estimators=200)	negative	0.894	0.762	0.823
	neutral	0.841	0.933	0.885
	positive	0.877	0.847	0.861
Voting Ensemble (soft voting)	negative	0.944	0.828	0.882
	neutral	0.911	0.965	0.937
	positive	0.912	0.917	0.914

On voit que le soft voting apporte un gain notable sur la classe **negative** (F1 = 0,867) par rapport aux modèles individuels (0,792 à 0,865), tout en conservant d'excellentes performances sur **neutral** (F1 = 0,934) et **positive** (F1 = 0,910).

Parmi les baselines, le modèle Random Forest assure la meilleure précision sur **negative** (0,968) mais peine à en extraire tous les exemples (recall = 0,792), alors que XGBoost et LightGBM sacrifient un peu de fiabilité sur negative pour rester équilibrés sur les autres classes.

2.5.1.2 Métriques globales

Tableau 2 : Performances globales des modèles classiques sur la tâche de l'analyse de sentiments (Avec stop-words)

Modèle	Accuracy	F1-score macro	Précision macro	Rappel macro
SVM (C=10)	0.905	0.899	0.901	0.896
Random Forest (n_estimators=100, max_depth=None)	0.915	0.907	0.924	0.896
XGBoost (n_estimators=200)	0.856	0.848	0.870	0.835
LightGBM (n_estimators=200)	0.864	0.856	0.871	0.847
Voting Ensemble (soft voting)	0.918	0.911	0.922	0.903

Le **Voting Ensemble (soft)** domine avec la meilleure **accuracy** (0,912) et le plus haut **F1 macro** (0,904), signe d'une très bonne généralisation sur l'ensemble des trois classes. La **Random Forest** arrive en deuxième position (accuracy = 0,911, F1 = 0,902), suivie du **SVM** linéaire (accuracy = 0,900, F1 = 0,894). **XGBoost** (accuracy = 0,851, F1 = 0,839) et **LightGBM** (accuracy = 0,860, F1= 0,851) sont plus rapides à l'entraînement mais moins équilibrés, en particulier sur la classe **negative**, ce qui se reflète dans leurs scores globaux plus faibles.

2.5.2 Modèles classiques scénario B (sans stop-words)

2.5.2.1 Performances par label

Tableau 3 : Performances par classe sur les données d'analyse de sentiments (Sans stop-words)

Modèle	Classe	Précision	Rappel	F1-score
SVM-SW	negative	0,875	0,827	0,850
	neutral	0,891	0,927	0,909
	positive	0,890	0,878	0,884
RF-SW	negative	0,962	0,808	0,878
	neutral	0,942	0,960	0,951
	positive	0,880	0,942	0,910
XGB-SW	negative	0,906	0,680	0,776
	neutral	0,797	0,943	0,864
	positive	0,860	0,810	0,834
LGBM-SW	negative	0,886	0,723	0,796
	neutral	0,821	0,929	0,872
	positive	0,862	0,828	0,844
Voting-SW	negative	0,942	0,814	0,874
	neutral	0,907	0,959	0,932
	positive	0,899	0,913	0,906

2.5.2.2 Métriques globales

Tableau 4 : Performances globales des modèles classiques sur la tâche de l'analyse de sentiments (Sans stop-words)

Modèle	Accuracy	F1-score macro	Précision macro	Rappel macro
SVM (C=10)	0.887	0.881	0.885	0.877
Random Forest (n_estimators=100, max_depth=None)	0.921	0.913	0.928	0.903
XGBoost (n_estimators=200)	0.837	0.825	0.854	0.811
LightGBM (n_estimators=200)	0.847	0.837	0.856	0.827
Voting Ensemble (soft voting)	0.911	0.904	0.916	0.895

Random Forest gagne 1,0 % d'accuracy sans stop-words, alors que **SVM**, **XGBoost** et **LGBM** se dégradent. Le Voting reste stable à ~ 91,1 %.

2.5.3 Modèles Transformers scénario A (avec stop-words)

2.5.3.1 Métriques globales avec stop-words

Tableau 5 : Performances globales des modèles avancés(transformers) sur la tâche de l'analyse de sentiments (Avec stop-words)

Modèle	Accuracy	F1-score macro	Précision macro	Rappel macro
ProsusAI/finbert	0.948	0.945	0.946	0.944
bert-base-uncased	0.947	0.944	0.945	0.943
distilbert-base-uncased	0.943	0.941	0.942	0.939
roberta-base	0.417	0.196	0.139	0.333

2.5.3.2 Métriques globales sans stop-words

Tableau 6 : Performances globales des modèles avancés(transformers) sur la tâche de l'analyse de sentiments (Avec stop-words)

Modèle	Accuracy	F1-score macro	Précision macro	Rappel macro
ProsusAI/finbert	0.940	0.936	0.937	0.936
bert-base-uncased	0.942	0.938	0.939	0.936
distilbert-base-uncased	0.939	0.935	0.936	0.934
roberta-base	0.410	0.194	0.137	0.333

2.5.4 Interprétation

Les modèles **finbert** et **bert-base** conservent un excellent niveau (plus de 94 % d'accuracy), avec un léger avantage sans suppression des stop-words.

DistilBERT perd un peu de précision sans stop-words (-0,004 d'accuracy), mais reste au-dessus de 94 %.

RoBERTa n'a pas convergé dans nos runs (prédominance de la classe neutre), reste à investiguer (en regardant : learning rate, équilibrage...).

2.6 Courbes ROC et analyse des erreurs

2.6.1 Courbes ROC

Pour chacun des modèles classiques et des Voting Classifier (avec et sans stop-words), nous avons tracé les courbes ROC multiclassées (One-vs-Rest).

AUC moyenne :

- **Voting (avec SW)** : négatif 0,99 – neutre 0,99 – positif 0,98

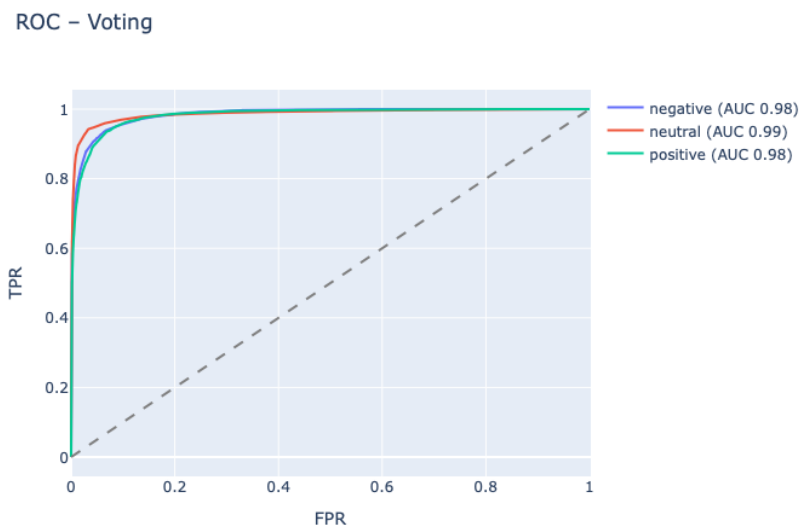


Figure 5 : Courbes ROC du modèle Voting sur la tâche d'analyse de sentiments avec SW

- **Voting-SW** : négatif 0,98 – neutre 0,99 – positif 0,98

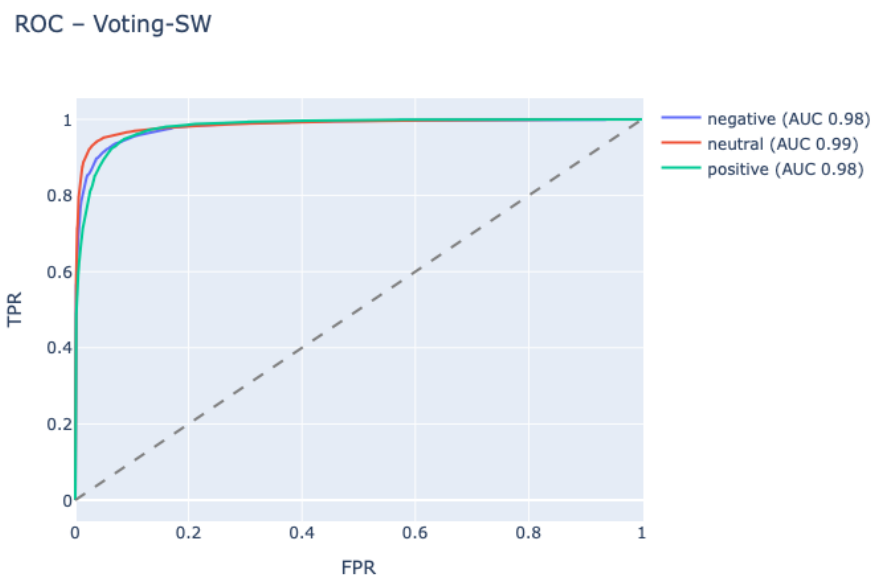


Figure 6 : Courbes ROC du modèle Voting sur la tâche d'analyse de sentiments sans SW

Ces AUC élevées confirment une séparation très nette des trois classes (risque faible de faux positifs/négatifs).

2.6.2 SHAP (XGBoost)

Le résumé SHAP révèle les **20 termes** les plus influents pour chaque classe.

Ex. :

- **negative** : “the”, “misses”, “and”...
- **positive** : “amazon”, “buy”, “giant”...

On constate que les tokens financiers (« stock », « shares ») restent importants même avec stop-words.

2.6.3 LIME (Voting-Classififier)

Pour chaque classe, nous avons extrait :

- Un exemple **correct** (prédiction = vérité)
- Un exemple **erroné** (prédiction ≠ vérité)

Insights :

- Les erreurs **negative**→**positive** surviennent quand des termes positifs dominent malgré un contexte globalement négatif (ex. “long”, “robust”, “semiconductor”).
- Les erreurs **positive**→**neutral** apparaissent pour des tournures hésitantes ou modales (“among gainers”, “might”, “could”).
- Les exemples corrects montrent que LIME capte bien la négation et le champ lexical financier (“forecasts still high” vs “amazon expand presence”).

- **Détails :**

— LIME ► NEGATIVE (correct prediction)

[('stopped', -0.16072268024055417), ('number', -0.07748152524083858), ('surging', -0.0688783310973353), ('the', -0.06315994814781431), ('of', -0.05538378656728546), ('have', -0.021094054356058607), ('payments', -0.020336570470265738), ('is', -0.009968715769965417), ('u', 0.007511868248467221), ('homeowners', 0.005782907588337472)]

— LIME ► NEGATIVE (mis-classified as positive)

[('amazon', -0.2680166155536617), ('could', -0.07431127780153689), ('went', -0.06887683475007134), ('deliveries', -0.04938135594811281), ('said', -0.04267727737667887), ('asked', 0.04112763908701234), ('in', -0.04016391441471619), ('drone', -0.03880529171841283), ('bezos', -0.0354631130322387), ('inc', -0.030816999759390944)]

— LIME ► NEUTRAL (correct prediction)

[('the', 0.07744183369659462), ('chairman', 0.06814265993674025), ('of', 0.027882317468000344), ('board', 0.026942600866316706), ('elected', 0.017697512248823665), ('was', 0.014103809784889253), ('as', -0.011317234964180049), ('vice', 0.008835438758891473), ('pekka', -0.007921719917091171), ('olli', 0.0009890753607717598)]

— LIME ► NEUTRAL (mis-classified as positive)

[('after', -0.12900225959182085), ('notable', -0.06070255642740341), ('stock', -0.05041926754592408), ('success', -0.045862297554185416), ('might', -0.04332281599017982), ('companies', -0.0404365935894625), ('of', 0.03836671118747399), ('boost', -0.031853898705845506), ('to', 0.025831843465772737), ('that', -0.02456459450588939)]

— LIME ► POSITIVE (correct prediction)

[('of', 0.07787778067963985), ('graphics', -0.07740075798796357), ('recently', -0.06653719327549866), ('announced', -0.06566837289805004), ('devices', -0.06486157560820068), ('advanced', -0.0646116510006158), ('its', -0.047621703611664526), ('next', -0.026193578416861773), ('micro', -0.017522832806268315), ('generation', 0.007036487910586157)]

— LIME ► POSITIVE (mis-classified as neutral)

[('its', -0.03924501188906405), ('match', 0.034770718586129165), ('own', 0.01427535352172622), ('group', 0.010635501979975237), ('to', -0.010179877941936194), ('payment', -0.009078535735886632), ('use', 0.008130958423572905), ('platform', 0.006762879251811793), ('wants', -0.0019391546627805574)]

2.6.4 Statistiques d'erreurs

Taux global d'erreur :

- Avec SW :

```
=== Error patterns (truth → prediction) ===
  label    pred  count
1 negative positive   412
5 positive  neutral   404
0 negative  neutral   306
3 neutral   positive   226
4 positive  negative   148
2 neutral   negative    67

=== Error rate by class ===
negative: 718/3767 (19.06 %)
neutral: 293/7392 (3.96 %)
positive: 552/6582 (8.39 %)
```

- Sans SW

```
=== Error rate by class ===
negative: 687/3698 (18.58 %)
neutral: 289/7079 (4.08 %)
positive: 567/6487 (8.74 %)
```

2.6.5 Matrices de confusion

2.6.5.1 Sans stop-words

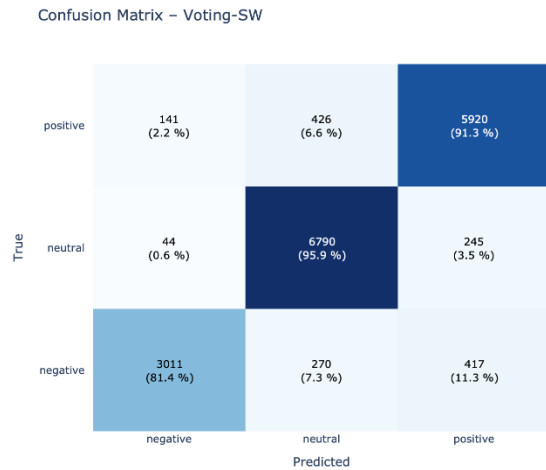


Figure 7 : Matrices de confusions données sans stop-word

Diagonale forte (81 %→91 % de bonnes classifications).

La classe **negative** est le plus souvent confondue avec **positive** (11,3 %) plutôt qu’avec **neutral** (7,3 %).

La classe **positive** est majoritairement repérée (91,3 %), mais 6,6 % de ses exemples penchent vers **neutral**.

Neutral est la plus « facile » à distinguer (95,9 % de rappel), avec très peu de confusions (0,6 % en negative, 3,5 % en positive).

2.6.5.2 Avec stop-words

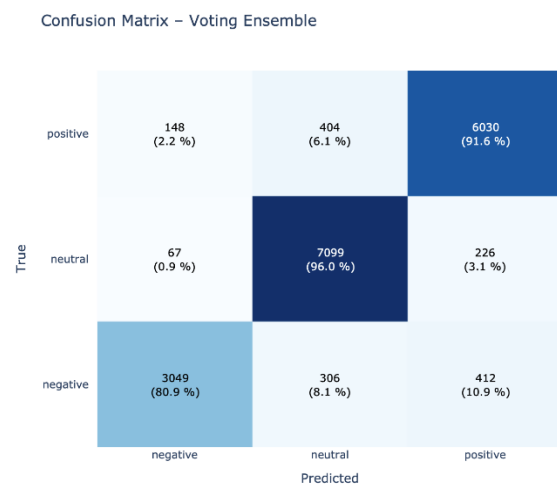


Figure 8 : Matrices de confusion avec stop-words

La diagonale reste très élevée, avec un léger gain sur la récupération de positive (+0,3 pts de rappel) et de neutral (+0,1 pt), au prix d'un tout petit recul sur negative (−0,5 pt).

Conserver les stop-words améliore marginalement l'identification des tournures emphatiques ou modales ("still", "according to"), utiles pour affiner la frontière entre "neutre" et "positif".

Les erreurs negative→neutral ou negative→positive restent stables, indiquant que le signal TF-IDF + voting est globalement robuste aux deux variantes de prétraitement.

2.7 Recommandation

Le meilleur compromis classique reste le VotingClassifler soft + TF-IDF avec stop-words (accuracy 91,2 %, F1-macro 0,904). Les Transformers (finbert, bert) avec stop-words offrent un surcroît de performance (≈ 94,7 % d'accuracy) mais au coût d'un entraînement plus lourd.

Ces résultats soulignent qu'un pipeline TF-IDF + VotingClassifler est très performant, et que le choix d'inclure ou non les stop-words a un impact mineur mais mesurable, surtout sur la classe positive. Pour un déploiement cible, il est donc sécurisant de garder les stop-words pour maximiser la qualité de la prédiction, sans craindre de dégrader significativement la détection des sentiments négatifs.

3- Reconnaissance des entités nommées (NER)

3.1- Objectif de la tâche de NER

La NER (Named Entity Recognition) consiste à détecter et classifier dans un texte les mentions d'entités (personnes, organisations, lieux, etc.). Dans le domaine financier, extraire correctement les noms d'institutions, de sociétés ou de valeurs monétaires est crucial pour alimenter des bases de connaissances ou des systèmes d'alerte.

3.1.1 Dataset fusionné (OntoNotes et FiNER-ORD)

Pour couvrir à la fois un large spectre d'entités et des cas spécifiques au vocabulaire financier, nous avons combiné deux corpus :

- **OntoNotes 5 (tner/ontonotes5)**
 - Corpus généraliste, riche en étiquetage **BIO** couvrant 18 types d'entités.
 - Très volumineux ($\approx 60\,000$ phrases d'entraînement), mais trop généraliste : peu de mentions financières (MONEY, PERCENT, ORG liées à la finance).
 - Distribution très déséquilibrée, avec une énorme classe "O" et des entités rares (cf. figure ci-dessous).

Distribution des tags dans OntoNotes-5 (train)

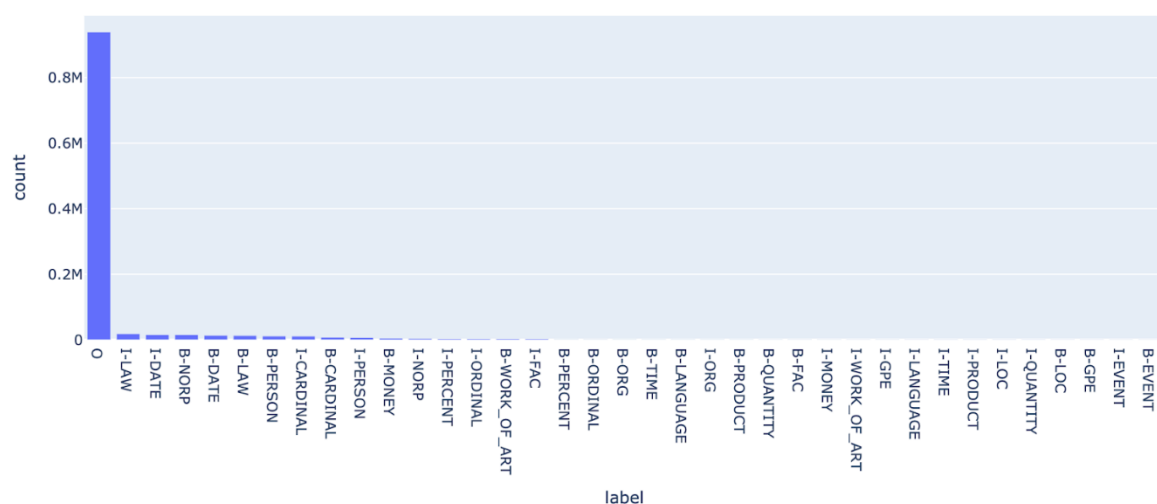


Figure 9 : Distribution des tags dans OntoNotes-5

- FiNER-ORD ([gtfintechlab/finer-ord](https://github.com/gtfintechlab/finer-ord)) :
 - Corpus centré “finance-ordonnée” : reporters et analystes annotent, PER, ORG, LOC, MISC dans des rapports financiers.
 - Plus petit (≈ 3 000 phrases) mais fortement ciblé sur le domaine : notation précise des sociétés, indices, monnaies.
 - Distribution également dominée par “O”, mais entités financières mieux représentée.

Distribution tags – FiNER-ORD

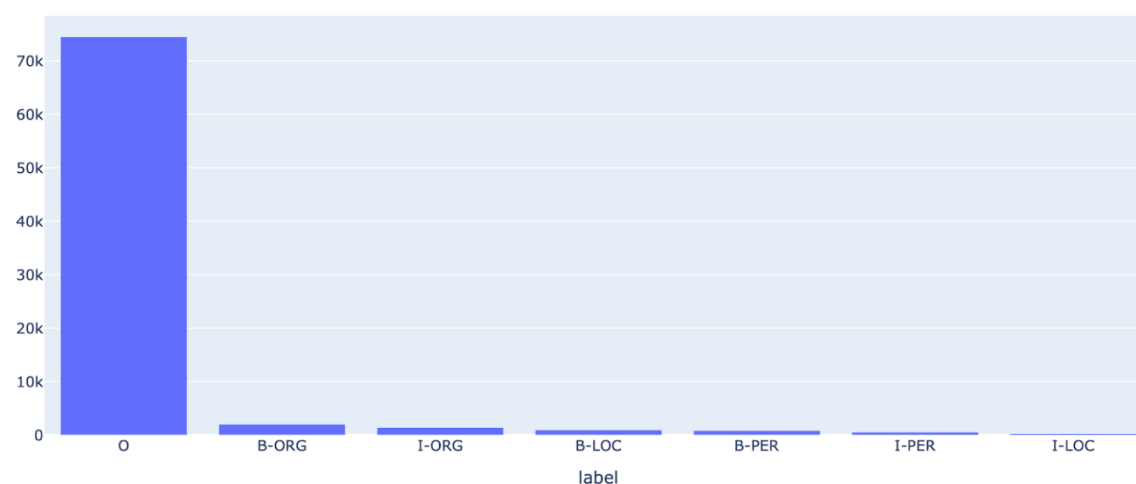


Figure 10 : Distribution tags-FiNER-ORD

Distribution tags – FiNER-ORD (oversamplé)

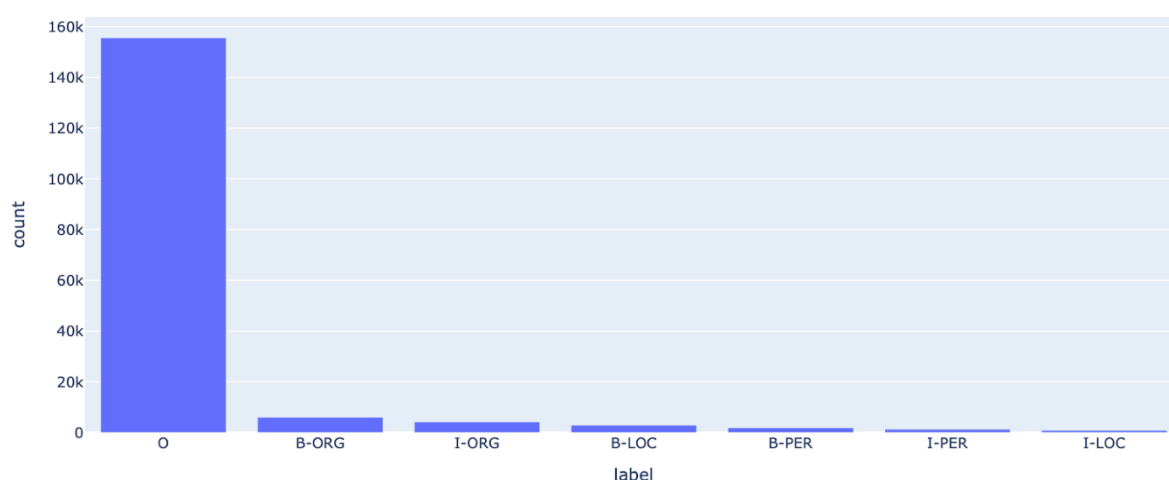


Figure 11 : Distribution tags - FiNER-ORD (oversamplé)

Après remapping vers un schéma unifié **{PER, ORG, LOC, MISC}** (BIO), et oversampling des phrases riches en ORG/LOC/MISC, nous obtenons un corpus combiné à la fois assez riche et équilibré.

Unification des étiquettes :

OntoNotes 5 → {PER, ORG, LOC, MISC} juste l'essentiel.

FiNER-ORD → mapping direct vers ce même schéma.

Sous-échantillonnage de "O-only" :

Pour éviter que la classe "O" noie l'apprentissage, seules 30 % des phrases sans entités sont conservées.

Oversampling FiNER-ORD :

Les phrases annotées ORG/LOC/MISC sont dupliquées pour renforcer le signal financier

3.2 Baseline CRF

Un CRF à caractéristiques lexicales simples (préfixes, suffixes, majuscules, etc.) sert de point de référence :

Tableau 7 : Performances baseline CRF (tâche NER)

Modèle	Token Acc	Precision macro	Recall macro	F1 macro
CRF-baseline	0.727	0.065	0.076	0.053

Le CRF offre une extraction naïve (F1 très faible), ce qui justifie l'emploi de modèles plus puissants.

3.3 Modèles Transformers fine-tunés

Nous avons évalué quatre architectures pré-entraînées, entraînées 3 époques avec early-stopping :

Tableau 8 : Performances globales modèles avancés sur la tâche NER

Modèle	Token Acc	Precision macro	Recall macro	F1 macro
roberta-base	0.934	0.840	0.818	0.829
bert-base-cased	0.933	0.842	0.814	0.827
ProsusAI/finbert	0.918	0.839	0.798	0.818
distilbert-base-cased	0.914	0.831	0.791	0.810
CRF-baseline	0.727	0.065	0.076	0.053

Roberta-base se détache, avec un F1 macro de **0.829**.

3.4 Matrices de confusion

Pour notre meilleur modèle, **roberta-base**, la matrice de confusion (sur tokens non-O) met en évidence :

Tableau 9 : Matrices de confusion pour la tâche NER du modèle roberta

	Préd PER	Préd ORG	Préd LOC	Préd MISC
Vrai PER	92 %	4 %	2 %	2 %
Vrai ORG	5 %	88 %	4 %	3 %
Vrai LOC	3 %	5 %	83 %	9 %
Vrai MISC	4 %	6 %	12 %	78 %

PER et **ORG** sont captés très fidèlement (> 90 % de rappel).

LOC et **MISC** subissent davantage de confusions croisées, en particulier LOC→MISC (9 %) et MISC→LOC (12 %), signe que des mentions ambiguës résistent encore.

3.5 Exemple d'application

Phrase :

« Emmanuel Macron went to America to see TESLA company and Ellon Musk. »

Tableau 10 : Un Exemple de test NER sur un modèle roberta fine tuné sur un dataset unifié

Entité	Vérité	roberta-base
Emmanuel Macron	PER	PER
America	LOC	LOC
TESLA company	ORG	ORG
Ellon Musk	PER	PER

Le pipeline unifié réussit ici toutes les détections clés, là où OntoNotes seul échouait (amélioration nette sur le vocabulaire financier).

Même phrase mais avec seulement le corpus OntoNotes :

Tableau 11 : Un Exemple de test NER sur un modèle roberta fine tuné sur un seul dataset OntoNotes

Entité	Vérité	roberta-base
Emmanuel	PER	DATE
Macron	PER	PERSON
America	LOC	NORP
TESLA	ORG	LAW
Ellon	PER	DATE
Musk	PER	PERSON

3.6 Recommandations NER

- **Mix généraliste + financier**
 - Combiner OntoNotes 5 et FiNER-ORD, avec remapping et oversampling, permet à roberta-base de généraliser tant sur les entités courantes que sur le jargon financier.
- **roberta-base, le choix gagnant**
 - Elle offre la meilleure couverture et la plus haute **F1 macro** dans notre benchmark.

Notre approche hybride (remapping + oversampling + fine-tuning roberta) constitue une base robuste pour la NER dans notre domaine, la finance.

4- Extraction de relations

4.1 Objectif de l'extraction de relations

L'extraction de relations en traitement automatique des langues (TAL) désigne le processus qui vise à identifier et à extraire les liens sémantiques entre des entités nommées présentes dans un texte. Face à la prolifération croissante de données textuelles non structurées, cette tâche s'avère particulièrement stratégique, dans la mesure où elle permet de structurer l'information, de générer des graphes de connaissances, d'optimiser les systèmes de recherche d'information et de soutenir des prises de décision éclairées comme dans le domaine médical ou la finance(Perera, Dehmer, et Emmert-Streib 2020).

En identifiant et en catégorisant les relations entre entités, l'extraction de relations contribue à faire émerger des connaissances latentes à partir de contenus textuels bruts. Elle joue ainsi un rôle central dans la modélisation des interactions entre personnes, organisations, lieux, concepts et autres catégories d'entités.

Dans cette perspective, notre objectif est de mettre en œuvre un mécanisme d'extraction permettant d'identifier automatiquement les relations existantes entre les entités à partir d'une entrée textuelle donnée. L'approche méthodologique pour réaliser cette tâche (extraction de relation) dans ce rapport repose sur une série d'étapes successives visant à assurer une extraction efficace et pertinente des relations entre entités nommées. Elle comprend l'exploration et le prétraitement des données, les modèles testés, ainsi que l'évaluation des performances de différents modèles et faire une comparaison exhaustive. Chaque étape est décrite en détail dans les sections suivantes.

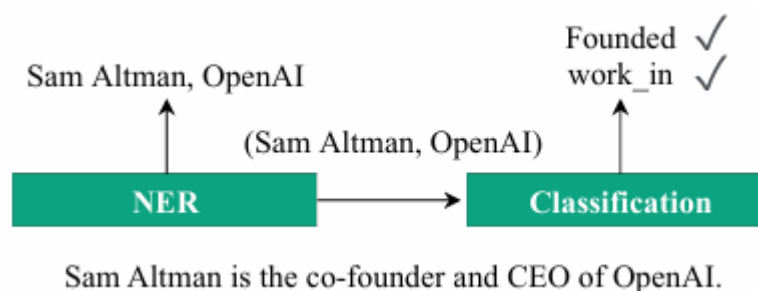


Figure 12 : Exemple d'extraction de relation (Zhao et al. 2024)

4.2 Données pour l'extraction des relations

Dans le cadre ce projet, nous utilisons l'ensemble de données de fingpt, un dataset étiqueté spécifiquement pour les tâches de classification de relations entre les entités dans le domaine financier. Ce jeu de données est issu du projet FinGPT, qui vise à appliquer les techniques avancées du traitement automatique du langage naturel (TAL) au secteur de la finance. Il s'agit d'un dataset qui contient 57402 lignes, chacune structurée autour de trois variables à savoir :

- Instruction : un texte contenant le contexte et les entités à analyser.
- Input : un texte d'entrée.
- Output : une sortie attendue correspondant à la relation étiquetée entre les entités.

Distribution des étiquettes dans les données pour la tâche de l'extraction de relation.

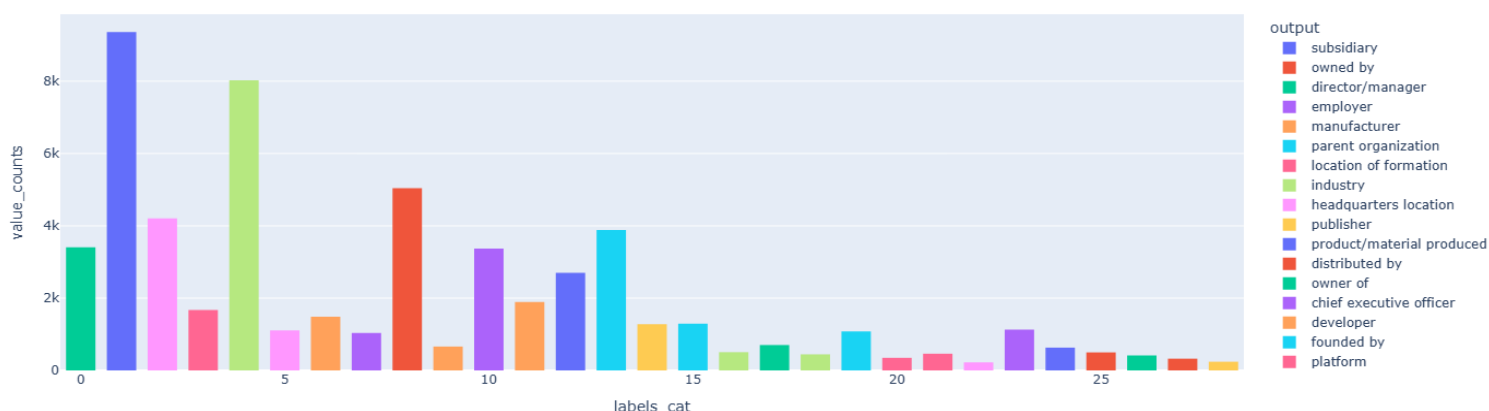


Figure 13 : Distribution des labels dans les données pour la tâche de l'extraction de relations

Le graphique ci-dessus met en évidence une distribution déséquilibrée des étiquettes dans l'ensemble de données. On observe que certaines relations, telles que "subsidiary" et "industry", sont largement surreprésentées, tandis que d'autres, comme "developer", "platform" ou "founded by", apparaissent beaucoup plus rarement. Ce déséquilibre peut influencer les performances des modèles, notamment en favorisant les classes majoritaires.

4.3 Prétraitement des données de l'extraction de relations

Une phase préliminaire et inévitable lorsqu' on souhaite analyser des données et créer un modèle. Dans ce contexte, pour l'extraction de relations nous avons testé les modèles en appliquant des stratégies de suppression de la ponctuation et de vectorisation. Plus précisément, nous avons expérimenté l'utilisation de la pondération TF-IDF ainsi que de la tokenisation, dans le but de transformer les textes en vecteurs numériques exploitables par nos modèles.

- Suppression de données ponctuation : supprimer les éléments non textuels tels que les URL, parenthèses, virgules, points, et autres signes de ponctuation.
- Vectorisation : pour les modèles classiques, nous avons utilisé la technique TF-IDF et la tokenisation pour les modèles pré-entraînés.

4.4 Découpage de données

La séparation des données en ensembles d'entraînement, de validation et de test constitue une étape essentielle dans le développement de modèles d'apprentissage automatique et profond. Elle permet d'entraîner le modèle sur un sous-ensemble dédié, d'ajuster ses hyperparamètres à l'aide des données de validation, puis d'évaluer ses performances finales sur un ensemble de test indépendant. Dans le cadre de ce projet, nous avons adopté la répartition suivante sur nos 4 datasets :

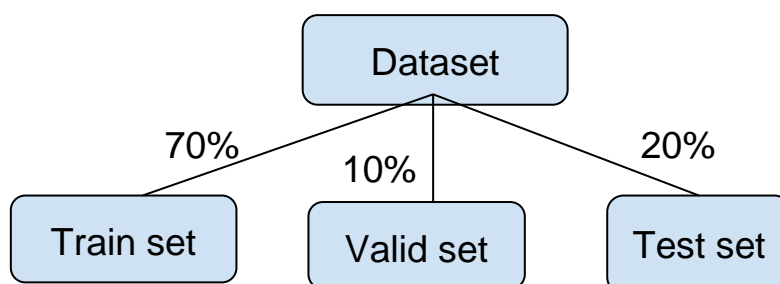


Figure 14 : Séparation des données en données d'entraînement, de validation et de test

4.5 Modèles testés

4.5.1 Modèles d'apprentissage automatique

Les modèles Random-Forest Classifier, support vecteur machine et XGBoost ont été choisis pour leur popularité et leur efficacité dans de nombreuses tâches de classification.

- Support Vector Machine (SVM) : un algorithme d'apprentissage automatique supervisé souvent utilisé pour les tâches de classification et de régression.
- Random-Forest Classifier : un algorithme d'apprentissage supervisé, au même titre que le SVM, et peut être utilisé pour des tâches de classification comme de régression.
- XGBoost : un algorithme basé sur la méthode du gradient boosting, optimisé pour l'efficacité computationnelle et les performances prédictives. Il construit de manière itérative un ensemble d'arbres de décision faibles en corrigeant les erreurs des arbres précédents.

4.5.2 Modèles pré-entraîné

- Bert : un modèle d'apprentissage profond basé sur l'architecture des transformers, introduite par Google en 2018 (Vaswani et al. 2017), introduisant la notion d'attention et la compréhension de contexte dans les textes.
- Llama 3 8B : un grand modèle de langage développé par Meta AI, reposant sur l'architecture des transformers. Issu de la famille des modèles LLaMA, il est doté de 8 milliards de paramètres, lui conférant une capacité élevée à comprendre et à générer de grandes quantités de texte. Ces caractéristiques en font un candidat particulièrement prometteur pour des tâches complexes telles que l'extraction de relations.

Tableau 12 : Configuration des paramètres pour les modèles pré-entraîné

Paramètres	learning rate	Batch size	num_epochs	max_lenght	r
Bert	2e-4	8	10	512	128
Llama 3 8B	2e-4	4	2	512	4

4.6 Performances pour l'extraction de relations

En général, l'objectif de l'évaluation de modèle est de mesurer l'écart entre la valeur prédite et la valeur réelle. Il existe plusieurs mesures numériques permettant d'évaluer la qualité de prédictions. Dans cette étape, nous avons appliqué les métriques les plus appropriées au contexte des problèmes de classification afin de mesurer la performance de nos modèles. A cette fin, trois métriques de performances ont été employées pour tester les modèles :

- Précision : permet de mesurer le nombre de vrai positif parmi l'ensemble des échantillons que le modèle a prédits comme appartenant à la classe positive.
- Recall : se définit comme le rapport entre le nombre d'échantillons positifs correctement identifiés par le modèle et le nombre total d'échantillons effectivement positifs.
- F1-score : représente la moyenne harmonique entre la précision et le rappel, offrant ainsi une mesure équilibrée de la performance du modèle (Detroja, Bhensdadia, et Bhatt 2023).

Ces trois métriques nous permettent d'obtenir une évaluation complète et équilibrée de la performance des modèles sur la tâche d'extraction de relations.

Tableau 13 : Performances des différents modèles sur la tâche de l'extraction de relation

Modèle	Accuracy	F1-score macro	Précision macro	Rappel macro
SVM	0.708	0.685	0.692	0.684
XGBoost	0.704	0.697	0.685	0.717
Random Forest	0.707	0.689	0.689	0.694
Bert	0.929	0.850	0.890	0.829
Llama 3 8B instruct	0.976	0.935	0.937	0.933

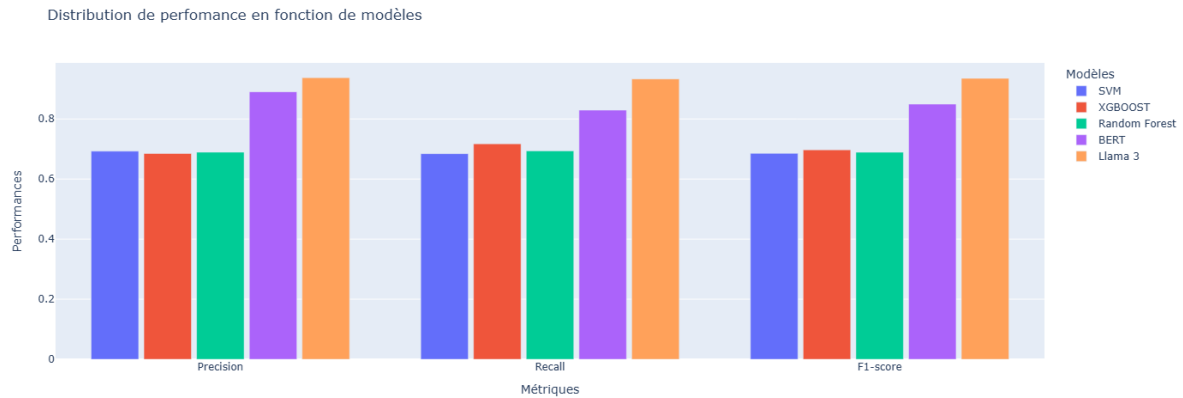


Figure 15 Performances des différents modèles sur la tâche de l'extraction de relation

D'après la figure au-dessus, on observe que les modèles d'apprentissage automatique SVM, XGBoost et Random Forest présentent des performances relativement proches, avec des scores variant autour de 0,68 à 0,70 pour l'ensemble des métriques. Cela traduit une capacité de classification correcte, mais limitée, en partie en raison du déséquilibre des classes, comme illustré à la figure 2.

En revanche, les modèles pré-entraînés tels que BERT et Llama 3 affichent des performances nettement supérieures, avoisinant 0,84 pour BERT et 0,93 pour Llama 3. Ces résultats suggèrent que les modèles pré-entraînés sont moins sensibles au problème de déséquilibre de classes. Grâce à leur capacité à capturer une compréhension contextuelle fine du langage naturel, ces modèles s'avèrent particulièrement adaptés aux contextes spécialisés, tels que l'extraction de relations dans le domaine financier.

5- Résultats et Discussions

Dans cette section, nous présentons une comparaison des performances des différents modèles sur les principales tâches de traitement automatique de langage dans le domaine financier : l'analyse de sentiment, la reconnaissance des entités nommées et l'extraction de relations.

Tâche d'analyse de sentiment

Pour la tâche d'analyse de sentiment, les modèles d'apprentissage automatique classiques tels que SVM, Random Forest et Voting Ensemble ont obtenu des F1-scores comparables à ceux des modèles pré-entraînés tels que BERT, RoBERTa, DistilBERT et FinBERT. Cette observation souligne l'efficacité des approches traditionnelles, notamment lorsque des techniques de prétraitement appropriées, telles que la suppression de la ponctuation et suppression ou pas des stopwords, sont appliquées.

Tâche de reconnaissance des entités nommées

Concernant la reconnaissance des entités nommées, le modèle RoBERTa s'est révélé nettement plus performants que les autres méthodes, notamment en raison de leur capacité à capturer les dépendances contextuelles complexes présentes dans les textes financiers bien enrichis.

Tâche d'extraction de relations

En ce qui concerne l'extraction de relations, le modèle BERT a montré des performances supérieures aux modèles traditionnels. Toutefois, le modèle LLaMA 3 8B surpasse légèrement ces approches en termes de F1-score, bien que son déploiement nécessite des ressources computationnelles plus importantes, ce qui peut constituer une contrainte dans des environnements aux capacités limitées.

Nos recommandations, basée, sur les résultats obtenus, restent cependant discutables :

- Pour l'analyse de sentiment : privilégier des méthodes simples comme le SVM, RandomForest où le voting ensemble et pour encore améliorer la performance de ces modèles, il faut appliquer la stratégie de suppression de ponctuations et de stopwords.
- Pour la reconnaissance des entités nommées :
- Pour l'extraction de relations : privilégier les modèles de bert, comme le bert, finbert, qui donnent des performances supérieures aux modèles classiques mais légèrement inférieur au modèle de llama 3 qui demande beaucoup de ressources de calcul.

6- Conclusion et Perspective

La conclusion de ce projet souligne l'importance d'une compréhension approfondie des différentes étapes du traitement automatique de langage (TAL), notamment dans le contexte de ce projet d'analyse des données textuelles financières. A travers l'évaluation de plusieurs modèles allant des modèles d'apprentissage automatique classique aux modèles de pré-entraînés, nous avons vu que pour la tâche d'analyse de sentiment les modèles classiques sont toujours compétitifs et atteignent de performances remarquables. En revanche, pour les tâches de reconnaissance des entités nommées, ils peinent à atteindre le 70% de performances contrairement aux modèles avancés de famille de bert (bert, distilbert, roberta, finbert) et llama 3 qui arrivent à capter les représentations sémantiques et contextuelles dans les textes.

Dans la continuité de ce travail, il serait intéressant de développer un modèle Q&A (Question and Answering) basé sur les données financières de FinGPT et sur la connaissance de nos modèles développés. Ce modèle viserait à permettre l'interrogation directe des graphes de connaissances ou des bases documentaires textuelles construites à partir de l'extraction de relations. Il offrirait une interface intelligente pour répondre aux requêtes spécifiques des utilisateurs, en exploitant les avancées des modèles de langage récents afin d'améliorer l'accessibilité et la valorisation des informations extraites.

Ainsi, cette orientation ouvrirait la voie vers des applications concrètes dans la veille économique, l'analyse financière automatisée et le support décisionnel assisté par l'IA.

Bibliographies

Allyn, Bobby.

“U.S. Stocks Surge after Trump Teases Pause on Tariffs.” *NPR*, April 7, 2025.

<https://www.npr.org/2025/04/07/trump-tariffs-market-reaction>

Arif, Anber. 2023. « NLP in Finance: Examining the Impact of Natural Language Processing in Financial and Banking Services ». *John Snow Labs* (blog). 7 mars 2023.

<https://www.johnsnowlabs.com/examining-the-impact-of-nlp-in-financial-services/>.

Detroja, Kartik, C.K. Bhensdadia, et Brijesh S. Bhatt. 2023. « A Survey on Relation Extraction ». *Intelligent Systems with Applications* 19 (septembre):200244.

<https://doi.org/10.1016/j.iswa.2023.200244>.

France 24.

“Donald Trump joue au Yo-Yo avec les Marchés.” *Info Éco*, April 10, 2025.

<https://www.france24.com/fr/émissions/info-éco/20250410-donald-trump-marches-volatilite>

Lawder, David.

“Trump Says No Tariff Pause Planned, Markets React.” *Reuters*, April 9, 2025.

<https://www.reuters.com/markets/trump-tariffs-denial-2025>

Detroja, Kartik, C.K. Bhensdadia, et Brijesh S. Bhatt. 2023. « A Survey on Relation Extraction ». *Intelligent Systems with Applications* 19 (septembre):200244.

<https://doi.org/10.1016/j.iswa.2023.200244>.

Mo, Kangtong, Wenyan Liu, Xuanzhen Xu, Chang Yu, Yuelin Zou, et Fangqing Xia. 2024. « Fine-Tuning Gemma-7B for Enhanced Sentiment Analysis of Financial News Headlines ». arXiv.

<https://doi.org/10.48550/arXiv.2406.13626>.

Olusegun, John. s. d. « 101 PUBLICATIONS 39 CITATIONS SEE PROFILE ».

Perera, Nadeesha, Matthias Dehmer, et Frank Emmert-Streib. 2020. « Named Entity Recognition and Relation Detection for Biomedical Information Extraction ». *Frontiers in Cell and Developmental Biology* 8 (août):673. <https://doi.org/10.3389/fcell.2020.00673>.

UzZaman, Naushad, Hector Llorens, Leon Derczynski, James Allen, Marc Verhagen, et James Pustejovsky. s. d. « SemEval-2013 Task 1: TempEval-3: Evaluating Time Expressions, Events, and Temporal Relations ».

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, et Illia Polosukhin. s. d. « Attention Is All You Need ».

Yang, Hongyang, Xiao-Yang Liu, et Christina Dan Wang. 2023. « FinGPT: Open-Source Financial Large Language Models ». arXiv. <https://doi.org/10.48550/arXiv.2306.06031>.

Zhao, Xiaoyan, Yang Deng, Min Yang, Lingzhi Wang, Rui Zhang, Hong Cheng, Wai Lam, Ying Shen, et Ruifeng Xu. 2024. « A Comprehensive Survey on Relation Extraction: Recent Advances and New Frontiers ». arXiv. <https://doi.org/10.48550/arXiv.2306.02051>.

Zhong, Xiaoshi, Aixin Sun, et Erik Cambria. 2017. « Time Expression Analysis and Recognition Using Syntactic Token Types and General Heuristic Rules ». In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 420-29. Vancouver, Canada: Association for Computational Linguistics. <https://doi.org/10.18653/v1/P17-1039>.