Digitalisation du processus de veille presse de la Lutte contre le blanchiment de capitaux et le financement du terrorisme en utilisant le NLP

Enge NOUADJE FOTSO
École Supérieure d'Ingénieur Léonard de Vinci
Paris, France
enge.nouadje_fotso@edu.devinci.fr

Abstract

Dans le cadre de la lutte contre le blanchiment de capitaux et le financement du terrorisme (LCBFT) au sein de la Banque de France, cette publication présente un pipeline de « natural language processing » (NLP) visant à automatiser la veille réglementaire par la détection d'articles de presse en ligne pertinents. Réalisé lors d'un stage de 20 semaines à la Banque de France, le projet intègre des techniques de scraping, de stockage de données et de classification d'articles à l'aide de méthodes basées sur le NLP.

Dans cette publication, différentes approches ont été explorées, mobilisant des modèles venants d'API privées comme OpenAI ou Mistral AI et open source comme CamemBERT (Martin, Muller, Javier Suárez, Dupont, Romary, Clergerie, Seddah, Sagot, 2020) ou jina-embeddings-v3 (Sturua , Moh , Akram, Günthe, Wang , Krimmel, Wang, Mastrapas, Koukounas, Wang and Xiao, 2024), ainsi que plusieurs méthodes de classification : similarité cosinus, fine-tuning de modèles transformers, et « Retrieval Augmented Generation » (RAG).

Le système a pour but d'être intégré aux processus métiers via une API Django ou des scripts Python interfacés avec Excel. Ces travaux soulignent l'apport des architectures de NLP pour renforcer les dispositifs de veille réglementaire dans le secteur financier.

Introduction

Les méthodes traditionnelles de veille médiatique, fondées sur des filtres par mots-clés ou une sélection manuelle, s'avèrent souvent insuffisantes face au volume, à la variabilité linguistique et à la rapidité de diffusion de l'information en ligne. Ces limites sont particulièrement problématiques dans le contexte LCB-FT, où l'identification rapide de signaux faibles et d'événements émergents est cruciale. Les experts faisant une sélection manuelle qui peut parfois être laborieuse et chronophage sont confrontés à ces limitations.

Pour remédier à ces limitations, on cherche un moyen de digitaliser le processus de veille de la LCB-FT à la Banque de France à travers l'avènement de la Data Science et de l'Intelligence artificielle. D'où le problème scientifique de cette publication « **Est-ce que le traitement automatique du langage (NLP) permet de digitaliser la veille presse dans le cadre de la Lutte contre le blanchiment de capitaux et le financement du terrorisme (LCBFT)?** ».

Ainsi, durant cette publication notre approche combine des techniques de scrapping, de vectorisation sémantique et de classification d'articles à l'aide du NLP pour répondre à ces enjeux.

I. État de l'art

Dans le cadre de la digitalisation du processus de veille presse en matière de Lutte contre le Blanchiment de Capitaux et le Financement du Terrorisme (LCB-FT), l'une des tâches centrales consiste à déterminer automatiquement si un article est pertinent ou non pour les analystes. Cette tâche, qui s'apparente à une classification binaire ("pertinent LCB-FT" / "non pertinent"), a fait l'objet de nombreuses approches en traitement automatique du langage naturel (NLP).

Historiquement, les méthodes classiques de classification reposaient sur des représentations statistiques du texte, comme les modèles Bag-of-Words (BoW) ou TF-IDF, combinées à des algorithmes d'apprentissage supervisé tels que la régression logistique, les forêts aléatoires ou les machines à vecteurs de support (SVM).

Bag-of-Words:

Le modèle Bag-of-Words (Abubaka, Umar, Bakale, 2022) consiste à représenter un document comme un simple ensemble de mots, sans tenir compte de l'ordre ou de la structure grammaticale. Chaque document est converti en un vecteur où chaque dimension correspond à un mot du vocabulaire global, et la valeur associée indique la fréquence de ce mot dans le document. Cette méthode est simple à implémenter, mais elle ne capture pas le contexte ni la signification des mots.

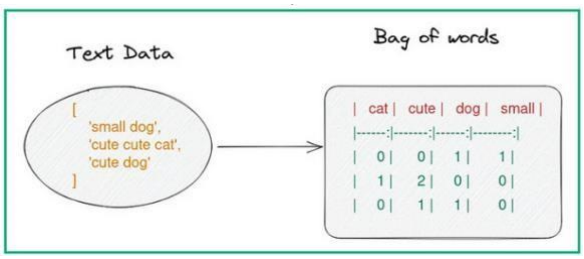

Figure 1 : Illustration Bag-of-words

TF-IDF:

Le modèle TF-IDF (Term Frequency–Inverse Document Frequency) améliore le modèle Bag-of-Words en pondérant les mots en fonction de leur importance (Abubaka, Umar, Bakale, 2022). La fréquence d'un mot est pondérée par l'inverse de sa fréquence dans l'ensemble du corpus, ce qui permet de diminuer l'influence des mots très courants (comme "le", "de", "et") et de mettre en valeur les mots plus informatifs. Cela permet d'obtenir des représentations textuelles plus discriminantes pour la classification.

$$w_{x,y} = tf_{x,y} \times \log\left(\frac{N}{df_x}\right)$$

TF-IDF
Term x within document y

$tf_{x,y}$ = frequency of x in y
df_x = number of documents containing x
N = total number of documents

Figure 2 : Formule fréquence TF-IDF

Bien que simples et rapides à mettre en œuvre, ces approches peinent souvent à capturer le sens sémantique des textes, ce qui limite leur performance dès lors que les formulations deviennent subtiles.

L'introduction des word embeddings, tels que Word2Vec a permis de mieux représenter les similarités lexicales et sémantiques entre mots.

Word2Vec:

Word2Vec (Abubaka, Umar, Bakale, 2022) est un modèle d'apprentissage non supervisé qui apprend des représentations vectorielles des mots à partir de leur contexte dans de grandes quantités de texte. Chaque mot est ainsi représenté par un vecteur dense de dimension fixe, de telle sorte que les mots apparaissant dans des contextes similaires ont des vecteurs proches dans l'espace vectoriel. Cela permet de capturer des relations sémantiques entre les mots. Par exemple, les termes « banque » et « finance » auront des vecteurs proches en raison de leur usage fréquent dans des contextes communs.

Dans le cadre de ce projet, c'est plus particulièrement l'architecture CBOW (Continuous Bag of Words) qui présente un intérêt. Cette variante du modèle Word2Vec consiste à prédire un mot cible à partir de son contexte (les mots voisins), ce qui est bien adapté à des corpus de taille limitée ou à des contextes thématiques restreints, comme celui des articles de presse liés à la LCB-FT. CBOW permet ainsi

de générer des représentations efficaces des mots clés du domaine, ce qui peut ensuite être exploité pour la classification de contenu.

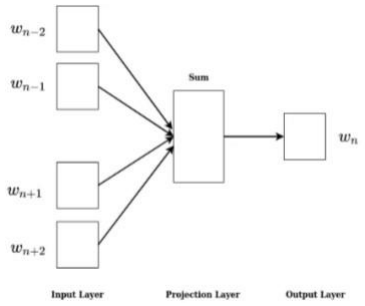

Figure 3 : Illustration Word2Vec (CBOW)

En agrégeant ces vecteurs pour représenter des documents entiers, les performances en classification se sont nettement améliorées, notamment sur des corpus spécialisés (comme ceux liés à la criminalité financière). Toutefois, ces modèles restent limités dans leur capacité à prendre en compte la structure syntaxique et contextuelle complète des phrases (Xu, Chen, Q. Weinberger, Sha, 2013).

L'avènement des modèles de type Transformer, en particulier BERT (Bidirectional Encoder Representations from Transformers), a marqué un tournant dans le domaine de la représentation du texte.

Transformers:

Les Transformers sont une architecture de réseau de neurones introduite en 2017 par Vaswani et al. dans l'article "Attention is All You Need" (Vaswani et al.,2017). Les Transformers traitent l'ensemble des mots d'une séquence en parallèle et utilisent un mécanisme d'attention pour modéliser les relations entre tous les mots, quelle que soit leur position. Cela permet au modèle de capturer efficacement les dépendances à longue distance dans les textes et de mieux comprendre le contexte global d'un mot dans une phrase. L'efficacité et la flexibilité de cette architecture ont rapidement conduit à son adoption massive dans de nombreuses tâches de NLP.

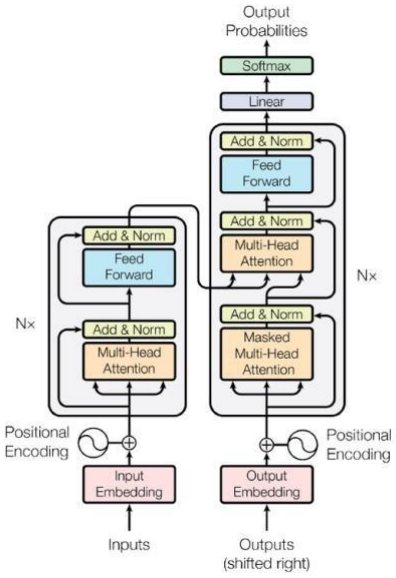

Figure 4 : Structure d'un Transformers

BERT:

BERT (Bidirectional Encoder Representations from Transformers) est un modèle fondé sur l'architecture Transformer qui introduit une approche bidirectionnelle du langage (Devlin, Chang, Lee, Toutanova, 2019). Contrairement à d'autres modèles qui lisent le texte de gauche à droite ou de droite à gauche, BERT lit les deux directions en même temps pour mieux comprendre le contexte de chaque mot. Il est pré-entraîné sur deux tâches : le masquage de mots, qui apprend à prédire des mots cachés dans une phrase, et la prédiction de la relation entre deux. Après ce pré-entraînement, BERT peut être entrainé (fine-tuned) sur des tâches spécifiques comme la classification de documents. Dans le cas de la veille LCB-FT, BERT permet de détecter des indices de pertinence parfois implicites, en s'appuyant sur une compréhension fine du langage, même lorsque les signaux sont peu évidents ou répartis sur plusieurs phrases.

Grâce à leur capacité à encoder les relations contextuelles entre tous les mots d'un texte, ces modèles ont établi de nouveaux standards pour la classification de textes courts ou longs. Dans un contexte LCB-FT, un modèle préentraîné comme BERT peut être entrainé (finetuned) sur un corpus annoté d'articles pertinents ou non pertinents, afin d'apprendre à repérer des signaux souvent diffus dans les textes, comme des liens indirects à des réseaux criminels, des structures financières opaques ou des faits de corruption.

Méthode/ Caractéristiques	Type de répresentation	Prise en compte du contexte	Capacité sémantique	Complexité de calcul
Bag-of-Words	Vecteur basé sur la fréquence des mots	Non	Faible	Faible
TF-IDF	Vecteur pondéré par fréquence inverse	Non	Faible	Faible
Word2Vec	Vecteur dense appris à partir du contexte local	Partiellement	Moyenne	Moyenne
Transformers	Représentation contextuelle par attention	Oui	Élevée	Élevée
BERT	Représentation bidirectionnelle contextuelle	Oui	Élevée	Élevée
RAG (Retrieval-Augmented Generation)	Représentation contextuelle avec recherche documentaire	Oui	Élevée	Très Élevée

Tableau 1 : Caractéristiques des méthodes

L'efficacité des modèles de classification repose cependant fortement sur la qualité du jeu de données d'entraînement. Dans le cas d'un domaine sensible comme celui de la LCB-FT, l'enjeu principal réside souvent dans la constitution d'un corpus suffisamment représentatif, équilibré et annoté de manière rigoureuse par des experts du domaine. Une autre piste de recherche consiste à intégrer des approches semi-supervisées ou des techniques d'apprentissage actif, permettant au modèle de s'améliorer progressivement à partir de retours d'experts humains.

Enfin, certaines approches plus récentes comme le RAG (Retrieval Augmented Generation) exploitent des architectures multimodales ou des techniques d'explicabilité afin de rendre les décisions du modèle plus transparentes. Un aspect crucial pour des applications réglementaires.

RAG:

Le RAG (Gaoa , Xiongb , Gaob ,Kangxiang Jiab , Panb , Bic , Daia , Suna , Wangc , Wang, 2024) est une technique d'IA combinant les capacités de génération de texte des modèles de langage avec des mécanismes de recherche dans une base de connaissances externe. Contrairement aux modèles purement génératifs qui s'appuient uniquement sur leur mémoire interne (venant entraînement), la méthode de RAG intègre une étape de "retrieval" (recherche) qui permet d'interroger une base documentaire (texte, articles ou bases réglementaires) pour extraire les informations les plus pertinentes. Ces documents récupérés sont ensuite utilisés comme contexte pour guider la génération de réponses plus factuelles, précises et vérifiables.

L'état de l'art évolue rapidement, mais l'objectif reste constant : fournir un outil fiable, interprétable et capable d'assister efficacement les analystes dans leur mission de veille sur des sujets critiques.

II. Méthodes et Supports

Pour mener cette recherche, une approche combinant revue de littérature, entretiens avec des experts, et structuration au fur et à mesure de la problématique a été adopté.

Une revue de la littérature sur la classification de texte a été réalisée à l'aide Google Scholar et arXiv. Cette recherche s'est concentrée sur les méthodes de classification fondées sur le « Natural Langage Processing » (NLP), avec un intérêt particulier pour les cas d'usage à visée réglementaire, notamment dans le domaine de la lutte contre le blanchiment de capitaux et le financement du terrorisme (LCBFT). Des ressources complémentaires comme Medium ou des blogs de particuliers ont

également été consultées pour enrichir la compréhension des pratiques actuelles.

Les objectifs de la recherche ont été progressivement affinés à mesure de l'avancée des travaux exploratoires. Cette phase de structuration a permis d'aligner les capacités techniques des approches NLP avec les exigences et les besoins métiers.

III. Résultats

Entrainement de CamemBERT :

Une première approche explorée pour la détection automatique d'articles de presse pertinents dans le contexte de la lutte contre le blanchiment de capitaux et le financement du terrorisme (LCB-FT) consiste à entraîner un modèle de type CamemBERT sur un jeu de données supervisé. Ce jeu de données est composé d'environ 1 100 titres d'articles, manuellement annotés comme pertinents ou non pour la LCBFT.

CamemBERT a été fine-tuné sur cette tâche de classification binaire. L'entraînement a été réalisé sur 3 époques, avec une taille batch size de 16 pour l'entraînement.

Cette phase de fine-tuning constitue une première base pour l'évaluation de la capacité des modèles de langage à détecter automatiquement les contenus à caractère sensible ou à risque, dans le cadre d'un système de veille automatisée de la LCB-FT.

Embeddings Jina + Similarité Cosinus :

Une seconde approche développée repose sur la comparaison sémantique entre les titres d'articles à classifier et un ensemble d'expressions clés représentatives des thématiques LCB-FT. Cette méthode utilise les représentations vectorielles générées par le modèle d'embedding « **jina-embeddings-v3** », couplées à une mesure de **similarité cosinus**.

Le principe de l'algorithme est le suivant : chaque titre d'article est préalablement segmenté en **n-grams** allants de n= 3 à n=5. Chaque **n-gram** ainsi que chaque expression clé LCB-FT est ensuite encodé à l'aide du modèle Jina. La similarité cosinus est calculée entre les vecteurs obtenus, et le score final de similarité pour un titre correspond au maximum des similarités entre l'un des **n-grams** du titre et l'ensemble des expressions LCB-FT.

Un seuil de décision a été fixé à 0,5 : si le score de similarité maximal dépasse ce seuil, l'article est considéré comme pertinent. Dans ce cas, l'expression clé LCB-FT associée au n-gram le plus proche est également retenue comme étiquette explicative. Cette méthode offre un cadre interprétable pour la détection en s'appuyant sur une approche sémantique non supervisée.

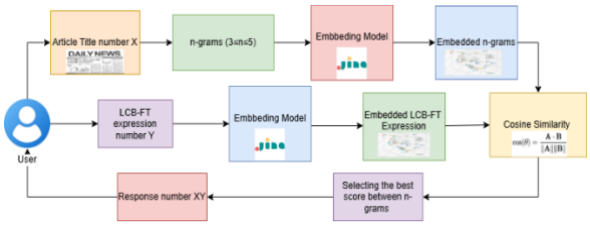

Figure 5: Fonctionnement méthode Similarité Cosinus

Retrieval-Augmented Generation:

Une troisième méthode explorée s'appuie sur le Retrieval-Augmented Generation (RAG), combinant des capacités de recherche sémantique avec une analyse assistée par un modèle de langage (ici GPT-nano). L'objectif est d'identifier, au sein du contenu complet des articles, les textes proches (de manière sémantique) d'expressions clés propres à la LCBFT, afin d'enrichir l'analyse et l'explicabilité des résultats.

L'algorithme mis en place repose sur une série de requêtes automatiques (queries) générées à partir des expressions clés LCB-FT. Pour chaque expression, une recherche est effectuée dans la base d'articles afin d'identifier les contenus pertinents. Contrairement aux approches précédentes limitées aux titres, cette méthode considère l'ensemble du corps de l'article, segmenté en fenêtres glissantes **(chunks)**.

Méthode/ Métriques	*F1-Score*	*precision*	*rappel*	*Accuracy*	*Nombre d'articles test*
Entrainement CamemBERT	0.62	0.64	0.60	0.61	509
Embeddings Jina + Similarité Cosinus	0.79	0.74	0.83	0.77	1132
RetrievalAugmented Generation	0.8	0.87	0.73	0.81	211

Tableau 2: Métriques des méthodes

Chaque chunk est ensuite analysé de **manière séquentielle** afin d'éviter le dépassement des limites de tokens imposées par l'API d'OpenAI (En termes de tokens par minute).

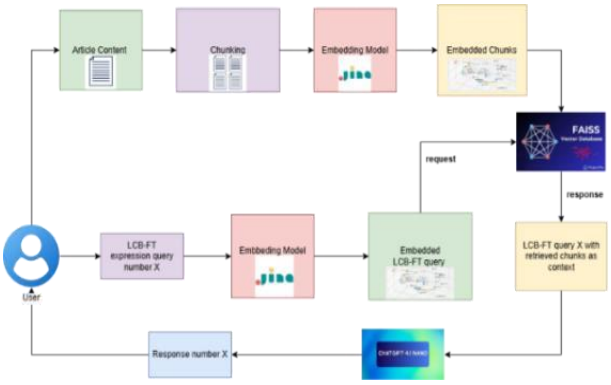

Figure 6: Fonctionnement méthode RAG

IV. Discussion

Le modèle CamemBERT fine-tuné obtient les moins bonnes performances globales, avec un F1Score de 0,62, une précision de 0,64 et un rappel de 0,60. Cela indique une faible capacité à identifier correctement les articles pertinents.

L'approche par embeddings Jina et similarité cosinus présente un meilleur rappel (0,83) et une meilleur précision (0,74), ce qui traduit une tendance à moins surdétecter les articles pertinents et à être plus sûr lorsque un article est détecté pertinent.

Enfin, la méthode par RetrievalAugmented Generation (RAG) montre un équilibre, avec une précision élevée (0,87) mais un rappel plus faible (0,73), suggérant qu'elle privilégie la fiabilité des détections au détriment de leur couverture.

Ces résultats montrent que la méthode RAG avec GPT-nano permet d'être plus efficaces lorsque les données possèdent un contexte étendu. Ils offrent un bon compromis entre exhaustivité (rappel) et exactitude (précision), ce qui est crucial pour des cas d'usage sensibles comme la veille réglementaire.

De plus, la méthode RAG, bien qu'ayant un rappel plus faible que la méthode Jina, présente l'avantage d'être plus facilement interprétable et modulable selon les contraintes métiers ou techniques (par exemple, l'exploration de textes longs ou l'adaptation à d'autres sujets).

Ces résultats ouvrent plusieurs pistes de recherche :

- Comment combiner les forces des approches supervisées (efficacité) et non supervisées (souplesse, explicabilité) ?
- Peut-on améliorer le rappel des approches RAG sans sacrifier la précision, par exemple via un meilleur découpage des chunks ou une meilleure requête ?
- Quel est l'impact de la qualité et de la diversité des expressions clés utilisées pour les méthodes non supervisées?

Méthode/ Benchmark	*Temps d'inférence pour classifier cent articles*	*Consommation d'énergie pour cent articles*
CamemBERT	~152 s	1.5 Wh (27 Wh d'entrainement)
Embeddings Jina + Similarité Cosinus	~30 s	~0.3 Wh
Retrieval-Augmented Generation	~2500 s	~1000 Wh

Tableau 3: Benchmark pour les différentes méthodes

V. Conclusion

Dans ce travail, nous avons comparé trois approches distinctes – un modèle supervisé (CamemBERT fine-tuné), une méthode non supervisée (embeddings Jina + similarité cosinus) et une stratégie hybride (RAG) – pour la classification d'articles en contexte de lutte contre

Les résultats montrent la stratégie hybride RAG demeure la méthode la plus performante, offrant un équilibre remarquable entre rappel et précision. L'approche RAG, en dépit de son rappel plus modeste que la technique avec la similarité cosinus, offre une forte précision et une traçabilité sémantique grâce à l'analyse détaillée (grâce aux chunks) des articles.

En conclusion, ce travail met en lumière les forces complémentaires des différentes stratégies d'analyse de texte et est encourageant pour le déploiement d'une application permettant aux experts de lancer la classification sur des articles sélectionnés et d'observer les résultats des classifications tout cela grâce à une interface API Django.

Bibliographie

Abubakar, H., Umar, M., & Bakale, M. (2022). Sentiment Classification: Review of Text Vectorization Methods: Bag of Words, TfIdf, Word2vec and Doc2vec. *Sule Lamido University Journal of Science & Technology*, 27-33.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv*, 1-16.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., . . . Wang, H. (2024). Retrieval-Augmented Generation for Large Language Models: A Survey. *arXiv*, 1-21.

Martin, L., Muller, B., Ortiz Suárez, P., Romary, L., Villemonte de la Clergerie, É., Seddah, D., & Sagot, B. (2020). CamemBERT: a Tasty French Language Model. *arXiv*, 1-17.

Sturua, S., Mohr, I., Akram, M., Günther, M., Wang, B., Krimmel, M., . . . Xiao, H. (2024). jina-embeddings-v3: Multilingual Embeddings With Task LoRA. *arXiv*, 1-20.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., N. Gomez, A., . . . Polosukhin, I. (2023). Attention Is All You Need. *arXiv*, 115.

Xu, Z., Chen, M., Q. Weinberger, K., & Sha, F. (2013). An alternative text representation to TF-IDF and Bag-ofWords. *arXiv*, 1-6.