Apprentissage Actif à l'ère des Grands Modèles de Langue (LLMs)

Shami Thirion Sen[1,2] Rime Abrougui[2] Guillaume Lechien[2] Damien Nouvel[1]
(1) Inalco, ERTIM, 2, rue de Lille, 75007 Paris, France
(2) Aday, 104, Bd du Montparnasse, 75014 Paris, France
shami.contact@gmail.com, rabrougui@aday.fr, glechien@aday.fr,
damien.nouvel@inalco.fr

RÉSUMÉ

En TAL, la performance des modèles dépend fortement de la qualité et de la quantité des données annotées. Lorsque ces ressources sont limitées, l'apprentissage actif (*Active Learning*) offre une solution efficace en sélectionnant les échantillons les plus pertinents à annoter. Traditionnellement, cette tâche est réalisée par des annotateurs humains, mais nous explorons ici le potentiel du grand modèle de langue Mixtral-8x7B pour générer automatiquement ces annotations. En outre, nous analysons l'influence de l'augmentation des données dans un processus d'apprentissage actif pour la reconnaissance d'entités nommées afin d'améliorer la performance des catégories sous-représentées, ainsi que l'impact du prompt et des hyper-paramètres sur la qualité des annotations générées. Les évaluations conduites sur le corpus WiNER montrent que, malgré l'absence d'annotations manuelles, cette approche permet d'obtenir des performances comparables à notre baseline, tout en réduisant de 80 % la quantité des données annotées.

ABSTRACT

Active Learning in the Time of LLMs.

For NLP, model performance heavily depends on the quality and quantity of annotated data. When these resources are limited, Active Learning offers an efficient solution by selecting the most relevant samples for annotation, traditionally performed by human annotators. In our study, we explore the potential of the large language model Mixtral-8x7B to automatically generate these annotations. Furthermore, we analyze the influence of data augmentation for certain under-represented categories within an Active Learning process for Named Entity Recognition, as well as the impact of the prompt and hyperparameters on the quality of the annotations generated. Evaluations on the WiNER corpus show that, despite the absence of manually annotated resources, this approach achieves performance comparable to our baseline while reducing the amount of annotated data by 80%.

MOTS-CLÉS : Apprentissage actif, annotation automatique, Reconnaissance d'Entités Nommées, Grands Modèles de Langue (LLMs), Augmentation des Données.

KEYWORDS: Active Learning, automatic annotation, Named Entity Recognition, Large Language Models (LLMs), Data Augmentation .

1 Introduction

L'efficacité de l'apprentissage automatique supervisé dépend notamment des quantités de données annotées. Cependant, l'annotation, bien que cruciale, reste complexe, chronophage et requiert une expertise pour garantir la qualité des résultats. Ainsi, en absence des ressources suffisantes pour l'annotation manuelle des données, l'apprentissage actif (en anglais *Active Learning*), s'avère une solution pour optimiser ce coût. Cette approche d'annotation permet d'atteindre des performances acceptables avec un nombre réduit d'annotations pour l'entraînement. Son principe repose sur un processus itératif où, à chaque cycle d'apprentissage, l'algorithme sélectionne les échantillons les plus pertinents à annoter pour améliorer la performance du modèle (Settles, 2009). Traditionnellement, cette annotation est réalisée par un annotateur humain. Dans ce travail, nous cherchons à déterminer si les grands modèles de langue peuvent générer des annotations exploitables pour l'apprentissage actif.

Nous expérimentons cette méthodologie pour la reconnaissance d'entités nommées (NER), tâche pour laquelle les modèles basés sur des transformers, souvent entraînés sur des corpus génériques comme WikiNER, se révèlent insuffisants pour les besoins industriels. En effet, une grande partie de ces modèles ne reconnaissent que quatre types d'entités standards (PER, LOC, ORG, MISC), ce qui limite fortement leur applicabilité à des corpus industriels nécessitant l'identification d'entités dans des domaines spécifiques. Dans notre contexte, le besoin de NER s'inscrit dans un cadre médiatique francophone propre à l'entreprise, avec un intérêt restreint à certaines catégories d'entités spécifiques. Or, les données disponibles sont limitées, et les contraintes de confidentialité ainsi que le manque de ressources humaines rendent difficile une annotation manuelle ciblée. Pour pallier ces difficultés, nous avons exploré la possibilité de réannoter un corpus existant selon l'ontologie d'un autre corpus jugé plus pertinent pour nos cas d'usage. Ces contraintes renforcent l'intérêt d'une approche par apprentissage actif.

Face à ces limitations, nous proposons une approche combinant l'apprentissage actif avec la génération et l'augmentation des données par LLMs pour enrichir le nombre de types d'entités. D'une part, nous générons des annotations en entités et des données synthétiques pour enrichir notre corpus d'affinage (ou d'apprentissage), permettant ainsi d'étendre le jeu d'étiquettes au-delà des entités classiques. D'autre part, nous intégrons ces données dans une boucle d'apprentissage actif, là où l'annotation repose traditionnellement sur un oracle humain. Notre étude exploite le modèle `Mixtral-8x7B` (Jiang *et al.*, 2024) pour la réalisation d'annotations des données en entités nommées, et l'intégration de ces données au processus d'apprentissage actif.

Notre contribution principale concerne la validation de cette méthodologie pour affiner des modèles d'annotation en entités nommées, avec une réduction importante de la quantité de données nécessaire pour l'apprentissage. Nous analysons, en premier lieu, l'impact du prompt et des hyper-paramètres sur la génération d'annotations en entités nommées ; nous observons ensuite, la performance du modèle NER (CamemBERT) final affiné avec ces données annotées dans un processus d'apprentissage actif. Notre approche atteint des performances comparables à l'état de l'art, malgré un corpus d'entraînement de taille réduite de 80%, démontrant ainsi l'efficacité de l'intégration des annotations générées par des LLMs dans l'apprentissage actif.

2 État de l'art

L'apprentissage actif est une approche d'apprentissage supervisée dont l'objectif est de minimiser la quantité de données annotées requises, tout en optimisant la performance d'un modèle (Settles,

2009). Dans ce processus cyclique, à chaque itération, l'algorithme détermine, selon des critères de sélection, quels échantillons non annotés d'un jeu de données doivent être annotés et ajoutés au corpus d'apprentissage. Classiquement, cette annotation est réalisée par un « oracle » ou un annotateur humain. Une fois les échantillons annotés, ils sont ajoutés à l'ensemble des données d'entraînement, et retirés du jeu de données non annotées, puis le modèle est ré-entraîné. Ce processus se poursuit jusqu'à atteindre un critère d'arrêt prédéfini (voir l'algorithme en Figure 1). Deux éléments fondamentaux structurent l'apprentissage actif : le critère de sélection des échantillons à annoter et le processus d'annotation lui-même.

La sélection des échantillons à annoter est essentielle en apprentissage actif, car elle détermine la performance du modèle entraîné. La méthode de sélection détermine quelles sont les données non annotées les plus utiles selon trois stratégies principales (Zhang *et al.*, 2023) : l'informativité, qui sélectionne les échantillons les plus incertains ou sujets à désaccord ; la représentativité, qui garantit une couverture équilibrée du corpus ; et une approche hybride, combinant ces deux critères pour maximiser la diversité et la pertinence des annotations.

Traditionnellement, l'annotation des échantillons sélectionnés est réalisée par des annotateurs humains. Toutefois, afin de réduire les coûts et l'effort requis, des alternatives sont actuellement explorées, notamment le recours aux grands modèles de langue (LLMs) comme annotateurs automatiques. Bien que l'utilisation des LLMs présente certaines limites, elle suscite un intérêt croissant dans la communauté scientifique. Dans (Tan *et al.*, 2024), les auteurs analysent la qualité des annotations générées par les LLMs, leur évaluation et leur intégration dans des tâches d'apprentissage. Pour la reconnaissance d'entités nommées, (Kholodna *et al.*, 2024) applique l'apprentissage actif à des langues peu dotées en utilisant exclusivement des annotations générées par des LLMs. Malgré le bruit présent dans les annotations générées et le coût de calcul associé aux LLMs, leur intégration dans l'apprentissage actif représente une piste prometteuse pour réduire les efforts d'annotations tout en améliorant la performance des modèles.

Pour la tâche de Reconnaissance d'Entités Nommées, l'utilisation de données annotées par des LLMs a déjà été proposée, notamment dans les travaux de (Zaratiana *et al.*, 2023). Leur modèle, `GLiNER`, propose de traiter plus particulièrement la reconnaissance ouverte d'entités nommées (Open NER) où l'objectif est de pouvoir annoter avec un jeu de type de données variables. Ce type de modèle, permettant de répondre également à nos contraintes, servira de point de comparaison dans nos évaluations finales.

3 L'apprentissage actif avec des données annotées par un LLM

Notre objectif est de dépasser les limitations des modèles existants pour la tâche d'extraction d'entités nommées en français, lesquels se limitent souvent à quatre types d'entités. Nous visons ici à élargir le nombre de catégories d'entités reconnues par le modèle, tout en réduisant les coûts d'annotation grâce à l'apprentissage actif.

Pour cela, nous mettons en place un protocole expérimental basé sur des itérations successives d'Active Learning. Chaque cycle repose sur une sélection progressive des données à annoter et sur l'utilisation du modèle `Mixtral-8x7b` pour effectuer les annotations.

Dans le cadre des présents travaux, nous avons souhaité choisir un LLM aussi performant que possible étant données nos ressources matérielles en GPU, ne disposant que de 48 Go de VRAM. Par ailleurs,

le modèle devait également supporter la langue française. A cet égard, à l'époque de nos expériences, le modèle `Mixtral-8x7B` était un des rares modèles à proposer le mécanisme de Mixture of Expert qui permet de réduire le nombre de paramètres utilisés à l'inférence. Ainsi, cela permet de profiter d'un modèle de 47B paramètres n'activant que 13B paramètres à chaque inférence. Par ailleurs, la performance d'un LLM étant corrélée avec sa taille, ce mécanisme propose ainsi un compromis intéressant entre vitesse et qualité espérée des annotations.

3.1 Jeux de données

Pour nos expérimentations de base nous utilisons deux corpus : WiNER (Dupont, 2019) et WikiNER (Nothman *et al.*, 2013).

1. **WiNER** (Dupont, 2019) est un corpus annoté en entités nommées construit à partir des articles de Wikinews en français, couvrant la période de 2016 à 2018. Il comporte un total de 1191 articles annotés en sept entités nommées, notamment en personne, lieu, organisation, date, heure, événement et produit (**Person**, **Location**, **Organisation**, **Date**, **Hour**, **Event**, **Product**). Le corpus a été annoté par un annotateur spécialiste du domaine. Étant donnée que le jeu d'étiquettes s'approchent de nos besoins, nous nous en servons comme corpus d'évaluation pour nos expérimentations.
2. **WikiNER** est un corpus silver automatiquement annoté en entités nommées, extrait à partir des données et des structures de Wikipedia (Nothman *et al.*, 2013). C'est un corpus multilingue (anglais, allemand, français, polonais, italien, espagnol, néerlandais, portugais et russe) dont nous nous servons uniquement de la version française de ce corpus.
3. **WikiNER dev** est un sous-ensemble du corpus WikiNER, composé de 98 énoncés en français manuellement annotés en sept entités nommées. Ce jeu de données nous sert de corpus de validation pour l'affinage du modèle CamemBERT pour nos boucles d'apprentissage actif. Il nous permet également d'évaluer les sorties du modèle Mixtral 8x7B afin de déterminer les meilleurs hyperparamètres pour la génération d'annotations lors de nos boucles d'active learning. Nous présentons les résultats dans le tableau 4.

3.2 Présentation de notre méthodologie d'apprentissage actif

Pour nos boucles d'apprentissage actif, nous travaillons sur un sous-ensemble représentant 20% du corpus d'entraînement `wikiner_train`, soit 24 136 énoncés, afin d'optimiser le temps d'apprentissage. Le choix des articles à annoter s'appuie sur une stratégie de sélection par moindre confiance. Après un calcul de score de confiance de chaque énoncé de l'ensemble non annoté `U`, nous sélectionnons 10% des énoncés ayant le score de confiance le plus bas. Ces énoncés sont ensuite annotés par le modèle `Mixtral-8x7B`, concaténé avec l'ensemble annoté `L`, et retiré de l'ensemble non-annoté `U`. Ce processus est illustré dans la figure 1.

A la fin de chaque boucle, nous affinons le modèle `camembert-base` avec l'ensemble de données annotées `L`, et nous évaluons notre modèle affiné sur notre corpus d'évaluation WiNER.

Afin d'optimiser le temps et la qualité de la génération de l'annotation des énoncés sélectionnés par l'algorithme, nous menons des expérimentations sur le **prompt** et les **hyperparamètres**.

Algorithme d'Apprentissage Actif

Input: Un ensemble de données non étiqueté $\mathcal{U}$
Output: L'ensemble de données étiqueté final $\mathcal{L}$ et le modèle entraîné $\mathcal{M}$

$\mathcal{L}, \mathcal{U} \leftarrow initialiser(\mathcal{U})$ ▷ Début
$\mathcal{M} \leftarrow entraîner(\mathcal{L}, \mathcal{U})$ ▷ Apprentissage du Modèle
while not critère_d'arrêt() **do**
    $\mathcal{I} \leftarrow interroger(\mathcal{M}, \mathcal{U})$ ▷ Requête
    $\mathcal{I}' \leftarrow annoter(\mathcal{I})$ ▷ Annotation
    $\mathcal{U} \leftarrow \mathcal{U} - \mathcal{I}$; $\mathcal{L} \leftarrow \mathcal{L} \cup \mathcal{I}'$
    $\mathcal{M} \leftarrow entraîner(\mathcal{L}, \mathcal{U})$ ▷ Apprentissage du Modèle
end while
retourner $\mathcal{L}, M_f$

FIGURE 1 – Les boucles d'apprentissage actif (Zhang *et al.*, 2023)

3.3 Le *prompt* ou le requêtage

Un «prompt» ou une requête est une instruction en langage naturel, parfois combiné avec des données, fourni au LLM pour la réalisation d'une tâche générative (Zamfirescu-Pereira *et al.*, 2023). Dans notre approche, ces prompts jouent un rôle central pour guider le modèle dans l'annotation des énoncés sélectionnés. Nous spécifions dans le prompt les types d'entités nommées (**Person**, **Location**, **Organisation**, **Date**, **Hour**, **Event**, **Product**) à utiliser pour annoter l'énoncé.

Nous expérimentons deux types de prompts : un prompt zero-shot, qui précise uniquement les catégories d'entités à générer ainsi que leur définition, et un prompt basé sur l'*in-context learning* (few-shot), qui intègre des exemples et des contre-exemples pour chaque catégorie d'entités nommées à annoter. Des exemples des deux prompts sont disponibles en annexe.

Le tableau 1 présente un comparatif, comme expérience préliminaire, des évaluations de ces deux prompts sur le corpus WiNER. Nous voyons que l'annotation est plus précise lorsque nous utilisons un prompt few-shot, intégrant à la fois des exemples et des contre-exemples. Cette expérimentation démontre également que l'annotation directe par `Mixtral-8x7B` ne fournit pas des résultats satisfaisants.

Prompt	**F1-macro**	**F1-micro**
prompt zero-shot	13,65	17,15
prompt few-shot	**21,85**	**26,48**

TABLE 1 – Évaluation des prompts sur WiNER

Structure du prompt few-shot

Nous vous présentons ici la structure du prompt few-shot que nous allons utiliser dans les expérimentations qui seront illustrées dans les prochaines sections. Il se décompose en plusieurs parties comme présenté dans le tableau 2 .

Il est à noter que ce prompt reprend une stratégie que certains travaux comme (Zamai *et al.*, 2024) ont

Persona
Description de la tâche
Manuel d'annotations (guidelines)
Contraintes d'annotations
Format
Contraintes sur le format
Exemplar

TABLE 2 – Structure du prompt few-shot

déjà utilisé et qui consiste à introduire un guide d'annotation pour décrire la connaissance nécessaire à une tâche, en donnant des exemples et des contre-exemples pour chaque type d'entité. Ce bloc d'instruction suit directement notre bloc de description de tâche. Il est également encadré par des délimiteurs afin de mettre en emphase cette information. Par ailleurs, le format de sortie demandé est une structure en JSON (Villena *et al.*, 2024). Le format demandé fera l'objet d'évaluations que nous présentons dans la section 3.4.

3.4 Les hyperparamètres et la structure de génération de données

Dans l'ingénierie de requêtage (*prompt engineering*), le réglage des hyperparamètres est important pour le fonctionnement efficace des LLMs (Renze & Guven, 2024). Tout d'abord, nous fixons la température à *0.0* afin d'obtenir des générations plus déterministes. Quant à la taille maximale de tokens produit par le modèle de langage (`max_token`), elle peut aider à optimiser le temps de traitement du prompt. Pour vérifier cette hypothèse, nous mesurons l'impact de la taille de la sortie sur des valeurs de 100 à 1000 tokens par paliers de 100.

Par ailleurs, la structure de données spécifiée dans le prompt peut avoir une influence sur la qualité et l'efficacité de la génération. Pour mesurer son impact, nous faisons ainsi varier les structures demandées sur deux critères : la sémantique (favorisant l'emploi de structures clé-valeur) et la possibilité de produire des structures imbriquées. Les variations utilisées sont présentées dans la table 3.

Structure de génération	**Format**
Clés ***text*** et ***entities***	`{"text": "CORIA-TALN 2025 aura lieu à Marseille.", "entities": [{"entity": "CORIA-TALN 2025", "type": "Event"} ...]}`
Uniquement **entités**	`{"entities": [{"entity": "CORIA-TALN 2025", "type": "Event"}], ...}`
Liste de dictionnaires	`[{"entity": "CORIA-TALN 2025", "type": "Event"}, ...]`
Liste de listes d'entités	`[["CORIA-TALN 2025", "Event"], ...]`

TABLE 3 – Structure de génération des entités

Notre objectif principal est de maximiser le compromis entre la **f1-mesure** et le **temps de génération** des entités. Nous évaluons chacune des configurations sur un petit ensemble du corpus WikiNER (98 énoncés), annoté manuellement, que nous nommons ici WikiNER dev. Nous aurons ainsi un corpus dont la granularité des entrées correspond à ce que le LLM doit annoter lors des boucles d'AL,

c'est-à-dire la phrase. La quantité d'énoncés annotés est motivée par les contraintes de ressources matérielles limitées. Aussi nous cherchons à pouvoir tester plusieurs configurations rapidement. Nous présentons les résultats saillants obtenus, en fonction des axes "format de sortie" (structure de génération) et "taille de la sortie" (max_token), selon la f1-micro dans le tableau 4.

max_tokens	Format de sortie	F1-macro	F1-micro	Temps
1000	Tout format confondu	54,5 - 55,2	61,8 - 63,6	7m39s - 9m36s
200	**entités uniquement**	55,3	**63,6**	**5m36s**
200	liste de dictionnaires	55,0	63,6	5m54s
200	original : avec clés texte et entités	55,5	63,2	6m54s
100	liste de liste	50,3	58,9	3m45s
100	liste de dictionnaires	47,1	55,8	4m21s
100	entités uniquement	47,8	54,5	4m09s
100	original : avec clés texte et entités	21,0	25,1	4m31s

TABLE 4 – Résultats en fonction de **max_tokens** et **format de sortie** des entités - Evaluations sur WikiNER dev

Ces résultats nous permettent de déterminer les hyper-paramètres optimaux pour générer les annotations en entités nommées avec notre modèle avec un temps d'inférence acceptable afin de réduire le coût de l'annotation. Nous adoptons les paramètres suivants : une **température** fixée à **0.0**, l'utilisation d'un **prompt few-shot** incluant des exemples, une **génération** des annotations au format structuré contenant les entités, leurs types et un exemple, et une fenêtre maximale du contexte (**max_token**) de **200** tokens.

3.5 Résultats des boucles

Nous présentons dans le tableau 5 les scores de la boucle 9, notre meilleure boucle d'apprentissage selon le f1-score. Le détail des résultats de ces expérimentations est consultable en annexe dans le graphique 4. Nous observons que notre modèle a réussi partiellement à traiter un certain nombre d'entités, mais il présente de faibles performances pour les catégories **Event** et **Product**, et une performance nulle pour la catégorie **Hour**, pour laquelle aucune entité n'est repérée. Ce problème provient de la quasi-absence de ces entités annotées avec ces étiquettes dans le corpus WikiNER. Afin de pallier le manque de ces catégories, nous menons des expérimentations avec l'augmentation des données synthétiques pour **Event**, **Product** et **Hour**.

Entité	Rappel	Precision	F1-score
Person	81,10	69,30	74,70
Location	74,30	52,50	61,50
Date	41,30	64,70	50,40
Organization	45,30	46,10	45,70
Event	16,80	48,30	**24,90**
Product	11,50	22,60	**15,20**
Hour	0,00	0,00	**0,00**

TABLE 5 – Performance du modèle camembert-affiné par catégorie d'entités nommées. Evaluations sur WiNER

4 Augmentation des données

Nos expérimentations initiales ont montré de faibles performances pour les catégories ***Hour***, ***Product*** et ***Event***. L'analyse de ces catégories a révélé que cette baisse de performance est due au manque d'exemples étiquetés par ces labels dans le jeu de donnée WikiNER. Étant donné que nous nous intéressons à ces labels, nous avons envisagé la possibilité d'augmenter le nombre d'exemples correspondants. Nous avons donc testé une augmentation des données pour ces trois catégories sous-représentées afin de renforcer leurs performances.

Pour ce faire, nous demandons au modèle GPT-4o de générer des données. Le modèle nous propose un script Python qui utilise une liste des heures et des patrons (chaînes de caractères) pour générer des phrases formatées permettant l'insertion des entités **Hour**, **Event** et **Product**. À noter que nous n'avons fourni aucun jeu de données confidentiel dans ce processus expérimental. Le modèle GPT-4o s'est limité à générer un script Python permettant la création de templates pour l'insertion des entités.

Le prompt ainsi que le script est disponible en annexe (voir 5.1.3). Grâce à ce script, nous générons 3500 énoncés, dont 1200 occurrences pour chaque entité.

Nous procédons ensuite à un affinage du modèle `camembert-base` avec les 3500 énoncés générés par le LLM pour notre première boucle. Par la suite, nous appliquons les mêmes étapes de notre système d'apprentissage actif, avec la sélection des énoncés et l'annotation de ceux-ci par `Mixtral-8x7B`.

	Person	**Location**	**Date**	**Organization**	**Event**	**Product**	**Hour**
AL	74,7	61,5	50,4	45,7	24,9	15,2	0,00
AL + aug. données	75,8	63,3	52,8	45,3	19,8	16,0	**38,7**
AL + hyper-param	75,9	65,9	59,9	**48,5**	**38,0**	1,0	0,0
AL + aug. donnée + hyper-param	**79,2**	**68.4**	**61,6**	44,3	34,9	**22,4**	37,3

TABLE 6 – Résultat par type des différentes expériences - F1-mesure (micro) - Evaluations sur WiNER

Nous observons une amélioration significative des performances dès la première boucle. Les résultats du modèle de la boucle 9, qui affiche les meilleurs f1-scores, présentés dans le tableau 6, confirment notre hypothèse. En effet, pour **Hour**, nous atteignons jusqu'à 38% de F1-score. L'augmentation des données a également eu un effet positif sur d'autres étiquettes, telles que **Person**, **Location** et **Date**. Toutefois, pour **Event**, nous remarquons une baisse, notamment au niveau du rappel. Cela peut être expliqué par la complexité sémantique de ce label, qui regroupe diverses entités, telles que les « tournois sportifs », les « congrès », les « événements annuels » et les « fêtes », ainsi que les « événements climatiques » et les « affaires politico-juridiques », pour lesquelles la génération de données par augmentation peut ne pas correspondre aux entités à annoter dans le corpus d'évaluation.

Nous rapportons les résultats de notre modèle final, comparé au modèle de l'état de l'art, dans le tableau 7. Malgré une quantité de données relativement faible, notre meilleur modèle obtient des résultats très proches de ceux du modèle `GLiNER` dans sa version multilingue, laquelle a été entraînée sur le français. Nous constatons également que l'augmentation des données a eu un impact positif sur le modèle final à la fois sur le F1-score mais également sur le besoin en données d'annotation.

Modèle	Corpus d'entraînement	Taille corpus (nb. de phrases)	F1-score
GLiNER multilingue	Pile NER	364 773	**70,8**
camembert-affiné-AL	WikiNER_fr réannoté	21 401	60,0
camembert-affiné-AL-avec-augmentation	WikiNER_fr réannoté	**17 638**	**70,7**

TABLE 7 – Résultats finaux et comparaison avec d'autres modèles - Evaluations sur WiNER

5 Conclusion

Ce travail présente nos résultats expérimentaux sur l'application de l'apprentissage actif pour la reconnaissance d'entités nommées dans un contexte industriel. Compte tenu des contraintes spécifiques à ce contexte, nous nous intéressons à un nombre défini d'étiquettes. La confidentialité des données étant une condition à respecter impérativement, nous avons utilisé le grand modèle de langue `Mixtral-8x7B` que nous avons pu déployer librement dans notre infrastructure. Afin d'étudier la possibilité d'annoter nos données d'entreprise avec une ontologie bien définie provenant du corpus WINER, nous avons exploré l'annotation automatique par le LLM dans un contexte aux ressources matérielles limitées. Nos résultats montrent que l'apprentissage actif permet d'obtenir de bonnes performances avec une quantité limitée de données annotées automatiquement, réduisant ainsi considérablement l'effort d'annotation manuelle. Par ailleurs, nous montrons que l'augmentation des données pour les catégories affichant des performances modestes permet d'obtenir des gains significatifs, rapprochant les résultats des performances de l'état de l'art.

Néanmoins, certaines entités, telles que **Event**, demeurent problématiques en raison de la complexité sémantique associée à ce label. Nous souhaitons approfondir l'analyse des erreurs associées à cette catégorie et améliorer la qualité des exemples générés, notamment en affinant les prompts. Plusieurs pistes d'améliorations peuvent être explorées pour prolonger ce travail, notamment la comparaison des résultats avec des grands modèles de langage (LLMs), de tailles différentes, utilisés en tant qu'oracle. Nous envisageons également d'appliquer cette approche à d'autres tâches de classification.

Références

BENAMARA F., HATOUT N., MULLER P. & OZDOWSKA S., Éds. (2007). *Actes de TALN 2007 (Traitement automatique des langues naturelles)*, Toulouse. ATALA, IRIT.

DIAS G., Éd. (2015). *Actes de TALN 2015 (Traitement automatique des langues naturelles)*, Caen. ATALA, HULTECH.

DUPONT Y. (2019). Un corpus libre, évolutif et versionné en entités nommées du français (a free, evolving and versioned french named entity recognition corpus). In E. MORIN, S. ROSSET & P. ZWEIGENBAUM, Éds., *Actes de la Conférence sur le Traitement Automatique des Langues Naturelles (TALN) PFIA 2019. Volume II : Articles courts*, p. 437–446, Toulouse, France : ATALA.

JIANG A. Q., SABLAYROLLES A., ROUX A., MENSCH A., SAVARY B., BAMFORD C., CHAPLOT D. S., DE LAS CASAS D., HANNA E. B., BRESSAND F., LENGYEL G., BOUR G., LAMPLE G., LAVAUD L. R., SAULNIER L., LACHAUX M.-A., STOCK P., SUBRAMANIAN S., YANG S., ANTONIAK S., SCAO T. L., GERVET T., LAVRIL T., WANG T., LACROIX T. & SAYED W. E. (2024). Mixtral of experts.

KHOLODNA N., JULKA S., KHODADADI M., GUMUS M. N. & GRANITZER M. (2024). Llms in the loop : Leveraging large language model annotations for active learning in low-resource languages. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, p. 397–412 : Springer.

LAIGNELET M. & RIOULT F. (2009). Repérer automatiquement les segments obsolescents à l'aide d'indices sémantiques et discursifs. In A. NAZARENKO & T. POIBEAU, Éds., *Actes de TALN 2009 (Traitement automatique des langues naturelles)*, Senlis : ATALA LIPN.

LANGLAIS P. & PATRY A. (2007). Enrichissement d'un lexique bilingue par analogie. In (Benamara *et al.*, 2007), p. 101–110.

NOTHMAN J., RINGLAND N., RADFORD W., MURPHY T. & CURRAN J. R. (2013). Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligence*, **194**, 151–175. DOI : 10.1016/j.artint.2012.03.006.

RENZE M. & GUVEN E. (2024). The effect of sampling temperature on problem solving in large language models.

SERETAN V. & WEHRLI E. (2007). Collocation translation based on sentence alignment and parsing. In (Benamara *et al.*, 2007), p. 401–410.

SETTLES B. (2009). Active learning literature survey.

TAN Z., BEIGI A., WANG S., GUO R., BHATTACHARJEE A., JIANG B., KARAMI M., LI J., CHENG L. & LIU H. (2024). Large language models for data annotation : A survey. *arXiv e-prints*, p. arXiv–2402.

VILLENA F., MIRANDA L. & ARACENA C. (2024). llmner : (zerolfew)-shot named entity recognition, exploiting the power of large language models.

ZAMAI A., ZUGARINI A., RIGUTINI L., ERNANDES M. & MAGGINI M. (2024). Show less, instruct more : Enriching prompts with definitions and guidelines for zero-shot ner.

ZAMFIRESCU-PEREIRA J., WONG R. Y., HARTMANN B. & YANG Q. (2023). Why Johnny Can't Prompt : How Non-AI Experts Try (and Fail) to Design LLM Prompts. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, p. 1–21, Hamburg Germany : ACM. DOI : 10.1145/3544548.3581388.

ZARATIANA U., TOMEH N., HOLAT P. & CHARNOIS T. (2023). Gliner : Generalist model for named entity recognition using bidirectional transformer.

ZHANG Z., STRUBELL E. & HOVY E. (2023). A survey of active learning for natural language processing.

ANNEXES

5.1 Les prompts

5.1.1 *Prompt zero-shot*

```
Perform very specialized Named Entity Recognition (NER) for the following French
    text.
```

```
Entities must have the following fields: the entity itself as "entity", and the
    entity type as "type", chosen from the following categories: Date, Event, Hour
    , Location, Organization, Person, and Product. The entity types should adhere
    to the following categorization:

- Date: Absolute dates.
- Event: Conferences, sports events, annual events, celebration days, named
    climatic events, etc.
- Hour: Absolute hours. If present, they include time zones, UTC, GMT, etc.
- Location: Countries, towns, regions, addresses, astrophysical objects,
    hydrophysical objects, etc.
- Organization: Non-profit organizations, companies, media, etc.
- Person: Human individuals, without their title or function. They can be actual
    people or fictional characters.
- Product: Physical objects, brands, software.

Generate duplicate entities in the same text. Do not change the uppercase or
    lowercase of the tokens.

Generate the entities in a SINGLE JSON format: {"text": text, "entities": [
    entities]}

Do not generate any comments, as they disrupt the JSON format.
```

Listing 1 – Prompt zero-shot

5.1.2 *Prompt few-shot*

```
You are a named entity recognition (NER) extraction model for French.
Follow these instructions as accurately as possible.

Only extract annotations from the following entity types:
[Date, Event, Hour, Location, Organization, Person, Product].
Do not generate other entities.

#### Definitions ####
1. Date: Only specific calendar dates. For relative dates, annotate only the
    absolute part.
   - Examples: 23 septembre 2024, 1er janvier
   - Counter-examples: "20345" is not a Date, "cinq jours" is not a Date

2. Event: Named occurrences like sports tournaments, conferences, annual events,
    holidays, climatic events, and political/legal matters.
   - Examples: championnats du monde de volley-ball 2023
   - Counter-examples: "soirée dansante" is not an Event

3. Hour: Only specific absolute times.
   - Examples: 15h00
   - Counter-examples: 880, 12 are not Hour

4. Location: Names of countries, towns, regions, addresses, astrophysical objects,
    or hydrophysical objects.
   - Examples: Université Leland Stanford Junior, Bali
   - Counter-examples: "maison", "école" are not locations

5. Organization: Named groups or businesses like non-profits, companies, or media
    organizations.
```

```
   - Examples: TCS
   - Counter-examples: "école", "hôpital" are not organizations

6. Person: Named human individuals (first names, last names, nicknames, fictional
    characters). Excludes generic groups.
   - Examples: Leland Stanford Junior
   - Counter-examples: "médecin", "personne", functions, jobs, or nationalities
       are not Person

7. Product: Named objects like software, media, or manufactured goods.
   - Examples: Airbus A380
   - Counter-examples: "film", "jeu" are not Product

#### Rules ####
- Do not annotate general terms like countries, regions, or generic objects.
- Annotate ONLY named entities.
- Repeat entities along with their types, even if they appear multiple times.
- Preserve the original case of tokens. Do not add extra spaces.

#### JSON Output Format ####
Generate the entities in a **single JSON format**:

{
  "text": "The input text goes here",
  "entities": [
    {
      "entity": "entity_name",
      "type": "entity_type"
    }
  ]
}

#### Formatting Rules ####
1. Use **only double quotes** for all string values and keys.
2. Use a **single quote** only for French apostrophes within the text.
3. Do not generate any extra comments or text unless asked to.
4. If comments are generated, place them **after the JSON**, separated by "####".

#### Example Output ####
{
  "text": "Here is an example text",
  "entities": [
    {
      "entity": "example_entity",
      "type": "example_type"
    }
  ]
}
```

Listing 2 – Prompt few-shot

5.1.3 Prompt - génération des données synthétiques (Event, Product, Hour)

```
Generate NER train data:
```

```
Can you create 3  json files each with 1000 varied texts each such that each
    contain the following NER categories . One  category per file for the
    following:
1. Hour: one or several instances of French Hour format per sentence, such that
    where the definition of Hour is quite strict: Hour (absolute hours. If present
    , they include time zones, UTC, GMT, etc.)
2. Event: conferences, sports events, annual events, celebration days, named
    climatic events, etc.
3. Product: Only refers to NAMED physical objects, software, video games, and
    media products. : example Airbus A380.
Generate a few longer sequences and avoid annotating common nouns as Named Entity,
     which will also help the model learn the entity vs. non entity pattern...

The structure of json should be the following:
{text: text, entities: [entity: entity, type : "Hour or Event or Product"] }
 I will use this silver data to fine tune a bert model for NER.
Thanks a lot!
```

Listing 3 – Prompt Data Augmentation

Script généré par GPT-4o

— **Exemple des entités et des modèles de phrase pour catégories HOUR**

```
    hours = [
    "15h00", "17h30", "18h00", "20h", "6h45", "9h20", "20h00 UTC", "
        22h00", "14h GMT", "12h30", "8h00 UTC", "23h45", "7h GMT", "
        10h", "11h30", "00h00", "5h15", "3h00", "16h45", "2h00", "19
        h", "4h30", "13h UTC","9h UTC", "22h30", "19h15", "8h45", "
        14h00", "23h00", "5h00", "12h00", "10h15", "16h00", "1h20",
        "0h30", "18h45", "9h50", "21h", "2h30", "11h GMT"
    ]

    hour_templates = [

    "La réunion commence à {} et se termine à {}.",
    "Je t'appelle demain vers {}, peut-être même à {} si je finis
        plus tard.",

    ...
    ]
```

— **Exemple des entités et des modèles de phrase pour catégories PRODUCT**

```
    products = [

    "Airbus A380", "iPhone 14", "Samsung Galaxy S21", "PlayStation 5
        ", "Windows 11", "Adobe Photoshop", "The Legend of Zelda:
        Breath of the Wild", "Netflix", "Sony WH-1000XM4", "MacBook
        Pro", "Microsoft Surface Laptop", "Call of Duty: Modern
        Warfare", "Nvidia GeForce RTX 3080", ...
    ]
```

```
product_templates = [

"Le casque {} est incontournable pour les professionnels du son,
     surtout comparé aux autres options du marché.",
"J'ai utilisé {}, et il est bien plus robuste que son prédé
    cesseur.",
"La montre connectée {} est la meilleure sur le marché pour
    suivre les performances sportives.",
...

]
```

— **Exemple des entités et des modèles de phrase pour catégories EVENT**

```
events = [
    "Festival de Cannes",
    "Tour de France",
    "Roland-Garros",
    "Fête de la Musique",
    "Journée Internationale des Droits des Femmes",
    "Coupe du Monde",
    "Jeux Olympiques",
    ...

]

event_templates = [

"Le sommet international sur le climat à Paris a été annoncé
    comme une extension du fameux {}.",

"Lors de {}, la question des droits de l'homme a été abordée,
    mais ce n'était pas un des sujets principaux.",

"La conférence de {} se tiendra à Bruxelles, mais contrairement
    aux élections législatives, elle ne sera pas retransmise en
    direct.", ...
]
```

5.2 Les graphiques - boucles d'apprentissage actif

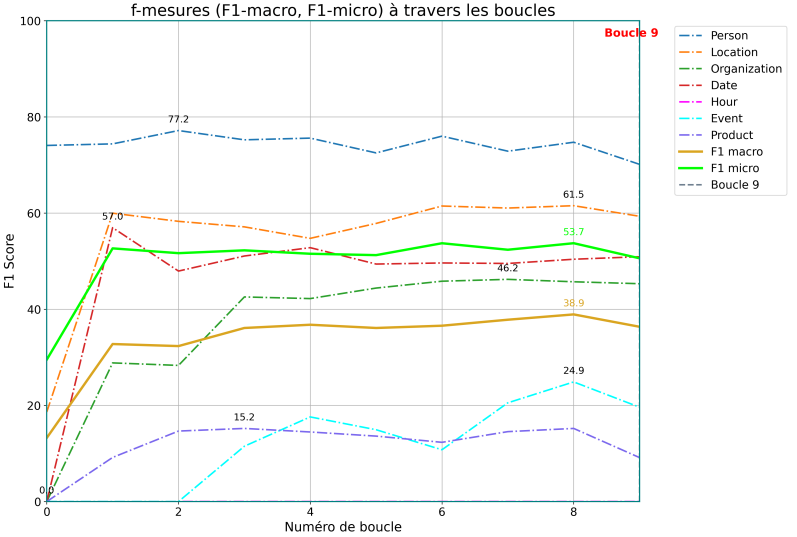

FIGURE 2 – Apprentissage actif

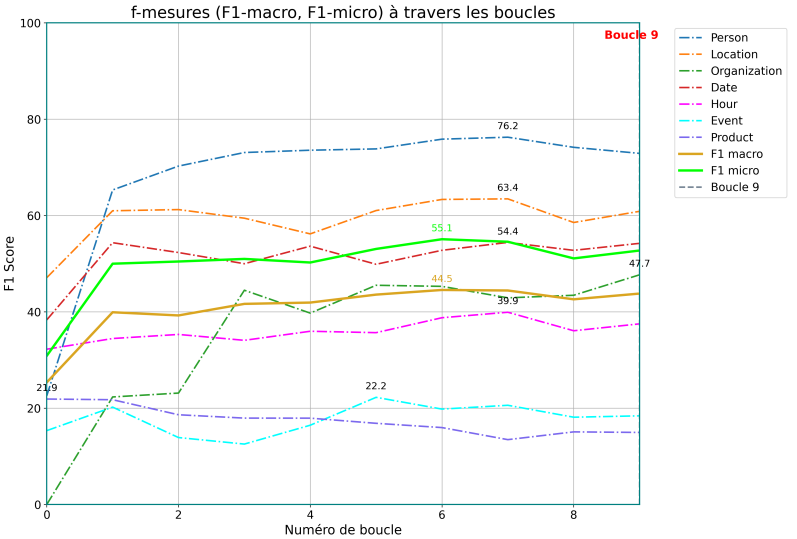

FIGURE 3 – Apprentissage actif - avec augmentation des données

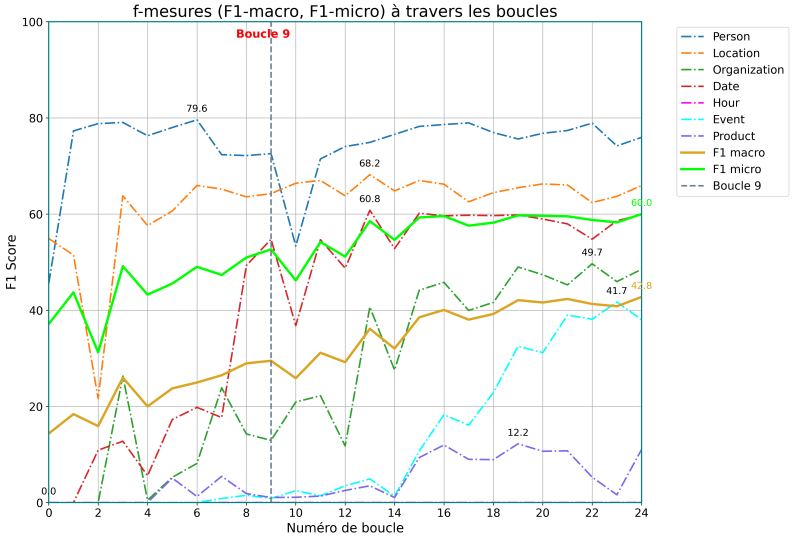

FIGURE 4 – Apprentissage actif sur 24 boucles - avec optimisation des hyper-paramètres

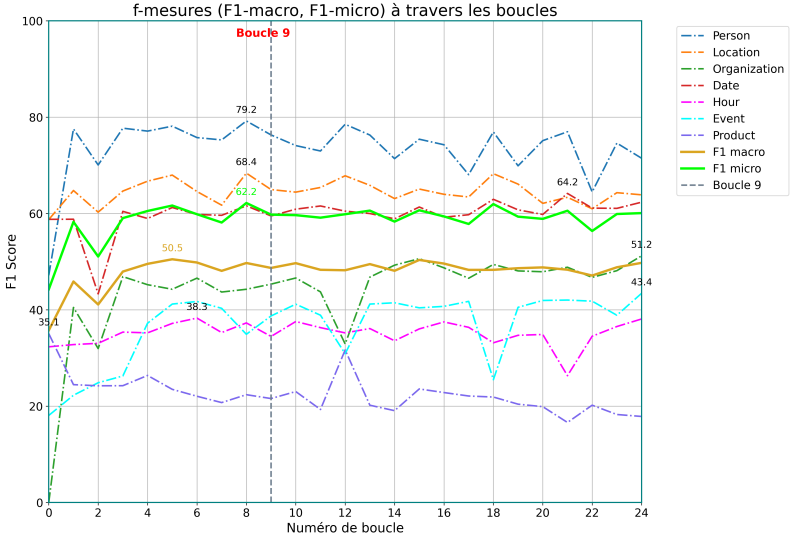

FIGURE 5 – Apprentissage actif sur 24 boucles - avec augmentation des données et optimisation des hyper-paramètres