Rapport de Participation de Smart Tribune à EvalLLM2024 : Quelques Usages de LLMs dans l'Univers de la Reconnaissance d'Entités Nommées

Guillaume De Murcia[1], Ilyas El-Allali[1], Ludovic Meineri[1], Laurent Gillard[1], Samy Lastmann[1]
(1) Smart Tribune R&D, Marseille, 13001, France
prenom.nom@smart-tribune.com

Résumé

Ce rapport présente les expériences de l'équipe R&D de Smart Tribune lors du challenge EvalLLM2024, menées dans un temps d'expérimentation limité. Nous avons exploré l'usage de Grands Modèles de Langages (LLMs) pour l'annotation de données, la création de jeux de données annotés et le fine-tuning sur des entités. Nous avons défini les prémices d'une méthodologie intégrant GPT-4o pour la constitution de données annotées et validé des approches modulaires, complémentaires et économiques pour la reconnaissance d'entités nommées, utilisant un dérivé du modèle CamemBERT et des fine-tunings de Mistral-7B, complétés par des méthodes plus classiques. Nos stratégies se sont concentrées sur la constitution de données annotées et leur utilisation pour le fine-tuning d'un LLM notamment sur l'entité TIME. Les résultats et les méthodes utilisées sont décrits dans ce rapport, mettant en lumière quelques perspectives d'amélioration futures.

Abstract

Participation Report of Smart Tribune at EvalLLM2024: Some Applications of LLMs in Named Entity Recognition.

This report presents the experiences of the Smart Tribune R&D team during the EvalLLM2024 challenge, conducted within a limited experimentation timeframe. We explored the use of Large Language Models (LLMs) for data annotation, the creation of annotated datasets, and fine-tuning on entities. We established the foundations of a methodology integrating GPT-4o for the creation of annotated data and validated modular, complementary, and cost-effective approaches for named entity recognition, using a derivative of the CamemBERT model and fine-tunings of Mistral-7B, complemented by more classical methods. Our strategies focused on the creation of annotated data and their use for fine-tuning an LLM, particularly for the TIME entity. The results and methods used are described in this report, highlighting some perspectives for future improvements.

Mots-clés : EvalLLM2024, Grand Modèle de Langage, LLM, Evaluation, Constitution de jeux de données annotées, Entités Nommées, CamemBERT, Fine-tuning, Few-shot, Mistral-7B, GPT-4o.

Keywords : EvalLLM2024, Large Language Model, LLM, Evaluation, Annotated dataset creation, Named entities recognition, CamemBERT, Fine-tuning, Few-shot, Mistral-7B, GPT-4o.

1 Introduction, contexte

Ce rapport décrit les expériences menées par l'équipe R&D de Smart Tribune dans le cadre de sa participation challenge EvalLLM2024 organisée par la DGA (https://evalllm2024.sciencesconf.org/resource/page/id/2)

Le challenge EvalLLM2024 rejoignait certains des enjeux de Smart Tribune liés à des problématiques d'explorations et de mise en oeuvre de Grand Modèles de Langages (LLMs) notamment sur les aspects suivants : l'utilisation de ces outils pour constituer des ressources annotées ou même annoter des corpus de données, et le fine tuning de modèles de type LLMs sur des tâches spécifiques d'annotations. C'est dans ce contexte que s'inscrit notre participation, ainsi que la comparaison avec des approches plus classiques. Cependant, des contraintes de calendrier et d'autres liées à notre activité courante ont limité le temps disponible pour cette première participation à une campagne d'évaluation de cette nature, et encore de nombreuses expériences ou évaluations restent à mener.

Ce rapport est structuré en trois grandes parties. La première partie récapitule les résultats obtenus par les différentes approches et constituants. Les deux parties suivantes approfondissent ces éléments de manière plus détaillée. La deuxième partie propose des expérimentations et une approche de création de jeux de données pour l'entité TIME, réalisée avec GPT-4o dans sa version conversationnelle. La troisième partie décrit les différents composants utilisés et leur combinaison et superposition pour nos soumissions. On y retrouve un modèle dérivé de CamemBERT, appliqué à une approche de reconnaissance d'entités nommées (NER), ainsi que des fine-tunings de Mistral-7B pour des typologies spécifiques d'entités (PER, LOC) d'un côté et TIME sur le jeu de données de la deuxième partie. Cette partie inclut également des compléments sur la superposition de ces méthodes et l'intégration de méthodes hybrides provenant de l'univers plus historique du TALN.

2 Notre participation à EvalLLM2024

2.1 Soumission et résultats

Lors de notre soumission, nous avons transmis 3 runs, le run1 était le plus complet en ce sens qu'il était un mélange de différentes approches, et offrait la plus grande couverture en termes d'entités du challenge, sans pour autant toutes les considérer (ce qui explique des résultats à zéro sur certaines typologies). Le run2 était lui uniquement constitué par les résultats obtenus depuis des fine-tuning de Mistral-7B et il était encore plus restrictif sur les typologies : seules PER, LOC et TIME sont considérées. Schématiquement, voici un bilan de superposition des approches qui sont décrites en détails ci-après :

run1 = NER1 + NER4-HYBRID + NER3-TIME
run2 = NER2 + NER3-TIME

Enfin, le run3 était un mélange des run1 et run2, cependant, nous nous sommes rendus compte très rapidement après soumission que ce run3 était corrompu par des problématiques d'ingénierie liées à ces mélanges d'approches et de constituants. Et, faute de temps supplémentaire à consacrer à ce

travail, ce run3 a été “abandonné”. D’autant que l’examen de ce dernier nous à fait prendre conscience d’autres incohérences dans les positionnement des sorties de NER1 impactant le run1, que nous avons corrigées, et que les organisateurs du challenge ont accepté d’évaluer.

Le run1 corrigé a obtenu une macro-précision de 34.12, la meilleure de notre participation. Toutefois, un macro-rappel de 21.32 et un macro-F1 de 24.62 révèlent des difficultés à capturer toutes les entités pertinentes - parmi celles qui nous étaient possibles - et celles à maintenir un bon équilibre entre précision et rappel. Nous renvoyons à l’article des organisateurs pour le détails des résultats de nos runs et ne proposons en Table 1 que les scores F1 pour les entités que nous avons considérées, même partiellement. À noter que le score RESOURCE est uniquement attribué à des montants monétaires, excluant la vaste étendue des ressources potentiellement à détecter. Comme attendu, le run 2 obtient des résultats beaucoup plus faibles avec une macro-précision de 15.36, un macro-rappel de 12.8 et un macro-F1 de 13.89, et les silences volontaires décidés impactent nécessairement ces métriques. Il faut cependant noter un meilleur score F1 sur les PERSON et la relative bonne performance de l’approche sur les TIME avec un F1 de 68.33. Pour conclure, d’autres explorations devront être menées ultérieurement pour étudier plus précisément les autres résultats. Par exemple, certaines différences observées sur les LOC entre les deux runs peuvent être intuitivement expliquées par les post-traitements effectués sur le premier, concernant les inclusions dans l’entité de la préposition introductive, comme requis par les organisateurs, bien que ces aspects n’aient pas encore été vérifiés, puisque le modèle sous-jacent du run2 avait été fine-tuné antérieurement à notre inscription à ce challenge.

run	PER	ORG	LOC	EQUIPMENT	RESOURCE	TIME	EVENT	ID
run1	87.91	59.72	39.83	8.27	20.51	36.19	0.96	66.67
run2	93.48	0.0	18.72	0.0	0.0	68.33	0.0	0.0

TABLE 1 : Scores F1 des 2 runs significatifs sur des entités considérées même partiellement

2.2 Retours d’expériences et empreinte carbone

Notre participation à ce challenge a été marquée par plusieurs défis, dont celui de la gestion du temps vis-à-vis de notre activité courante, cependant elle a ouvert des perspectives complémentaires. Elle a également mis en évidence de nombreux points d’amélioration ou nécessités d’observation, notamment le besoin d’intégrer des outils de monitoring du fine-tuning comme W&B ou MLflow dans nos expériences.

Une autre des difficultés rencontrées a concerné le très faible volume de données d’entraînement, mis à disposition eu égard au grand nombre d’entités à détecter et de la complexité de certaines nuances (comme celles des nationalités et de leur usage comme entité politique/étatique, ou celles d’entités discontinues/imbriquées que nous n’avons que peu ou pas considérées) même si ce faible volume correspondait effectivement à une logique de few-shot learning/prompting.

Cette contrainte était particulièrement bien adaptée à notre volonté de mettre en œuvre des stratégies différenciées, multiples et structurées autour d'outils complémentaires, comme autant de briques à assembler, pour la reconnaissance d'entités nommées au sein d'univers métiers spécifiques. Ce découpage en petits constituants, permettrait d'avoir à terme une modularité et une forte flexibilité, en permettant des améliorations itératives, mais aussi un coût d'usage très facile à contrôler. Cela rejoint également des problématiques de gestion de contexte des LLMs, la limitation à un petit nombre d'entités pour chaque outil permettant de bénéficier d'un contexte limité.

Concernant les approches complémentaires, nous avons un temps retenu l'approche de scrapper des sites relevant du domaine militaire et du renseignement pour augmenter la couverture en vocabulaire/lexiques mais nous sommes ravisés vu les problématiques potentielles sur les droits d'auteur, aussi nous nous sommes limités à quelques usages parcimonieux d'API et d'une ressource libre comme Wikipedia. De même, nous avons renoncé à copier/coller des parties du guide d'annotations et des données d'entraînement dans nos usages de GPT-4o en raison des clauses de confidentialité, mais il aurait été intéressant de constater la compréhension du modèle depuis ces consignes d'annotations, puisque nous avons trouvé celles-ci difficiles à interpréter même du point de vue notre compréhension et l'adaptation à des fins de prompt engineering nous a paru être un autre challenge.

Enfin, notre approche à base de superposition avec de petits outils d'étiquetage NER nous a joué quelques tours lors de l'assemblage final des sorties de l'ensemble des systèmes. En particulier, nous nous sommes rendu compte après soumission de certains décalages et d'incohérences sur les positions des entités. Par exemple, des décalages de proche en proche sont survenus en raison de normalisations destructrices des espaces séparateurs consécutifs et/ou de retours chariot "\n" sur un des premiers composants, il a été nécessaire de ré-aligner après coup ces positions. Aussi, nous remercions les organisateurs d'avoir accepté de réévaluer ce run1 corrigé. Le run3 est lui trop corrompu pour même avoir du sens.

Empreinte carbone

Cette approche modulaire, combinant divers outils et méthodes, nous a permis de valider une stratégie économique adaptée à nos capacités limitées ("low-resource") ce qui se traduit également par une empreinte carbone relativement faible, calculée ci-dessous (Lannelongue et al., 2020). Concernant le fine-tuning de Mistral 7B dans la tâche de NER multilingue sur les entités PER et LOC :

This algorithm runs in 2h and 37min on 8 GPUs NVIDIA A100 PCIe, and draws 7.63 kWh. Based in Germany, this has a carbon footprint of 2.58 kg CO2e, which is equivalent to 2.82 tree-months (calculated using Green Algorithms v2.2).

Cumul des temps de fine-tuning de Mistral 7B pour la détection des TIME et de ceux des inférences/étiquetages (ces derniers sont marginaux) sur la machine locale RTX 4090 et un processeur Intel Core I7 :

This algorithm runs in 24min on 1 GPU other and 16 CPUs other, and draws 235.97 Wh. Based in France, this has a carbon footprint of 12.10 g CO2e, which is equivalent to 1.32e-02 tree-months (calculated using Green Algorithms v2.2).

3 Le cas des entités temporelles : quelques approches exploratoires pour la constitution de ressources d'apprentissage via un LLM (GPT-4o)

Le type d'entité TIME a suscité un intérêt supplémentaire de notre part pendant ce challenge. En effet, nous n'avions qu'une gestion partielle des entités temporelles, dans quelques-unes de ses versions les plus simples, telles que les dates absolues. Au-delà des enjeux d'améliorer notre couverture des entités temporelles, nous souhaitions valider deux hypothèses : la première était que la qualité des annotations obtenues par un modèle de la famille GPT, et a fortiori sa version la plus récente GPT-4o, serait suffisante pour constituer un jeu de données qualitatif. Avec pour objectif de fine-tuner un (plus petit) LLM dans la détection des entités TIME. Pour la seconde, nous avons supposé que nous n'avions besoin que d'un nombre limité d'exemples. Nous avons fixé empiriquement ce nombre à un millier de phrases annotées.

3.1 Des approches infructueuses

En première approche, nous avons tenté d'utiliser GPT-3.5, GPT-4 puis GPT-4o comme modèles de NER en leur fournissant quelques exemples et explications sur la nature des entités temporelles. Conformément à nos attentes les résultats étaient relativement médiocres. En effet, la reconnaissance d'entités nommées est par nature une tâche complexe pour un LLM (Wang et al., 2023) et cela s'est vérifié dans nos expérimentations de prompt engineering même sur ces formes simples.

Dans un même esprit que GPT-NER de Wang et al. (2023), nous avons ensuite positionné le LLM en tant que générateur de textes annotés plutôt qu'en annotateur de textes. Avec en ligne de mire du few-shot prompting (Chen et al., 2024), nous avons extrait les entités TIME du jeu d'entraînement, puis les avons modifiées et enrichies avec des lexiques. Depuis cette liste, nous avons demandé au LLM de générer des histoires dans lesquelles les entités temporelles ont été écrites entre balises <TIME>entité</TIME> (cf. Annexe 1). Malgré de nombreuses tentatives, variant prompt engineering et ajustement des paramètres de génération comme la température notre appréciation des résultats est restée mitigée. En Annexe 2 se trouve un exemple de prompt utilisé lors de ces expérimentations. Nous avons en particulier relevé un nombre non négligeable d'annotations incomplètes ou erronées, des hallucinations et des temps de générations très longs (avec GPT-4). Le modèle GPT-4o ne s'est pas montré plus convaincant, notamment sur la consistance à inclure ou non l'ensemble de la préposition. Par exemple, il arrive au LLM de générer "<TIME>en 2024</TIME>" mais aussi mais aussi "<TIME>2024</TIME>" en omettant la préposition "au" introductive.

Nous avons également essayé de décomposer TIME en plusieurs sous-catégories comme DATE, PERIODE ou MT (cf. Annexe 3). Nous avions remarqué que des adverbes comme "rapidement" ou "fréquemment" dans les exemples annotés induisaient le modèle à considérer d'autres groupes adverbiaux comme entités temporelles. Ce dernier travail reste à approfondir, et découle de l'hypothèse que les entités temporelles peuvent être mieux modélisées en considérant différents niveaux de complexité.

3.2 Une méthodologie itérative

Nous nous proposons ici de détailler la méthodologie que nous avons envisagée et suivie pour répondre aux faiblesses décrites dans les précédentes approches. Felkner et al. (2024) soulignent les défis liés à l'utilisation des LLMs pour des tâches d'annotation et mettent en évidence la nécessité de l'annotation humaine pour des tâches complexes et sensibles. Inspirés également par Huang et al. (2024) nous avons décidé d'inclure dans notre méthodologie des étapes d'annotations humaines. L'hypothèse forte derrière notre méthodologie est que le few-shot prompting n'ayant pas suffit au LLM pour qu'il généralise correctement l'annotation des TIME, nous allions devoir aborder le procédé d'annotations automatiques de manière itérative. Nous avons donc utilisé la console ChatGPT de OpenAI avec la version GTP-4o afin de bénéficier de l'historique de la conversation avec le LLM.

Dans un premier temps, nous avons annoté manuellement une dizaine de textes issus du jeu de données WikiNER_fr (Jean-Baptiste, 2021b) traduit en français, comme par exemple "Cependant, la guerre des Six Jours culmina avec la fermeture du canal de Suez <TIME>entre 1967 et 1975</TIME>".

Nous avons ensuite demandé à ChatGPT-4o de poursuivre notre travail en annotant une centaine de phrases supplémentaires issues du WikiNER_fr. Les annotations générées par ChatGPT-4o ont été vérifiées et corrigées manuellement. Notons que nous avons ressenti que l'effort à fournir en corrections était moindre comparativement à l'annotation humaine du texte brut. Une fois corrigées nos annotations ont été partagées avec ChatGPT-4o, c'est ainsi que le LLM est passé d'une dizaine à une centaine d'exemples annotés avec un certain degré de fiabilité.

Pour augmenter la taille du jeu de données, nous avons employé deux stratégies. D'abord par remplacements et variations : nous avons demandé à ChatGPT-4o de générer des variations des textes annotés existants en remplaçant les entités temporelles par des équivalents ou des antagonistes, tout en adaptant la structure des phrases. Nous voulions ainsi augmenter la diversité des annotations en limitant le risque d'erreurs de la part du LLM. Par exemple, le LLM a transformé la phrase "<TIME>Hier</TIME>, la contestation était [...]" en une phrase <TIME>Demain</TIME>, la contestation sera [...]".

Notre seconde stratégie d'augmentation a été de procéder directement par création de textes annotés, sous l'hypothèse que par son historique de conversation ("contexte") le LLM avait acquis une compréhension et un alignement plus fin de nos attentes sur ces entités temporelles. C'est aussi en ce sens que notre approche se veut itérative. En effet nous avons défini des thématiques comme les jours de la semaine, les mois de l'année, les durées, les fréquences, etc. Nous avons successivement demandé au LLM de créer des histoires annotées autour de ces thèmes avec la consigne de varier entre le format numérique et littéral quand cela avait du sens. Une autre astuce a été de demander pour chaque série de générations de commencer par une dizaine d'exemples, de vérifier et corriger si besoin, puis de confirmer au LLM qu'il pouvait poursuivre ses générations en ce sens. Nous avons complété notre jeu de données avec des phrases ne contenant aucune entité temporelle afin d'éviter, lors du fine-tuning, d'introduire un biais sur la présence systématique d'entités à détecter. Nous avons enfin ordonné aléatoirement les textes, pour éviter à nouveau un biais de construction lié à notre approche. Nous avons ainsi obtenu un jeu de données de mille textes annotés en TIME contenant un total de 1296 entités temporelles, dont 674 étaient uniques.

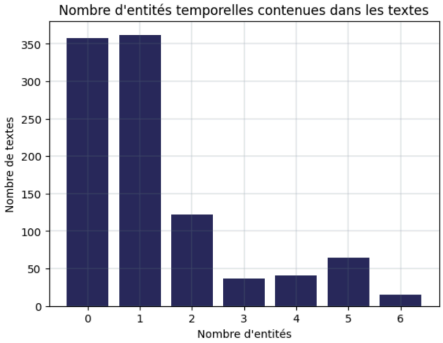

FIGURE 1 : Répartition du nombre d'entités par "texte"

Notre étude nous a montré que l'utilisation itérative d'un LLM dans un contexte de création de ressources annotées nous permettait d'obtenir une bien meilleure précision dans les annotations que lors de l'utilisation directe du LLM pour l'annotation. Cependant, des limitations subsistent, notamment la nécessité de relecture et de corrections manuelles des annotations générées par l'IA. La variabilité, la pertinence et plus généralement l'intégrité de la donnée générée par l'IA restent discutables. Néanmoins, cette méthodologie itérative s'est avérée efficace et pourrait être adaptée à d'autres types d'entités nommées. Les résultats obtenus sont encourageants et ouvrent la voie à des recherches futures visant à affiner et optimiser cette méthodologie.

4 Superposition de modèles de reconnaissance des entités nommées

Notre approche correspond à une volonté d'utiliser des modèles relativement petits afin de minimiser les temps de réponse pour garantir une utilisation en temps réel notamment au sein d'agents conversationnels et au travers d'appels d'API. Cela correspond également à une volonté d'équilibrer et faire des compromis raisonnables entre des coûts d'usage, et de fait leur empreinte carbone, sans pour autant sacrifier à leur performance. Par conséquent, nous avons envisagé une approche générale proposant une superposition des résultats de différents modèles, et des stratégies hybrides, dont nous présentons ici chacun des constituants.

Certains de ces modèles ont été complétés ou développés rapidement pour traiter quelques uns des enjeux spécifiques à ce challenge, mais l'ensemble de nos expérimentations consistaient à privilégier des approches d'usages ou de spécialisation de LLM, en conservant quelques raccourcis vers des approches plus classiques à base de TALN sur lesquelles il serait possible d'itérer ensuite. Enfin, une des problématiques était de mettre en place ces stratégies dans un temps particulièrement court/limité.

4.1 Un premier outil interne : NER1, dérivé d'un modèle CamemBERT et d'autres méthodes plus classiques

Dans le cadre de ses besoins internes, Smart Tribune a incorporé un modèle de reconnaissance d'entités nommées de type PER (PERSON dans ce challenge). Ce modèle a été dérivé des travaux de (Delestre & Amar, 2022) et ceux de (Jean-Baptiste, 2021a).; eux-même basés sur le modèle

CamemBERT (Martin et al., 2020). Ce modèle a été entraîné depuis les données du jeu de données WikiNER (Nothman et al., 2013) et une traduction en français WikiNER_fr (Jean-Baptiste, 2021b).

L'ensemble est inclus dans un outil de reconnaissance des entités nommées qui est essentiellement utilisé pour de la détection de noms de personne et de lieux. Enfin un certain nombre de pré-traitements et post-traitements correctifs sont mis en œuvre : parmi les pré-traitements, une normalisation des espaces séparateurs, un masquage des entités susceptibles de perturber le modèle comme des adresses emails (prenom.nom@quelquechose.com) ; parmi les post-traitements : quelques règles de réappropriation des étiquettes LOC depuis des entités PER, notamment dans des cas d'inclusion ou d'adjacence avec des marqueurs et constituants typiques d'une adresse comme dans l'expression *"rue Jean Mermoz", où "Jean Mermoz"* est plutôt une partie du lieu qu'une entité PER.

Il faut rappeler que cet outil permet de reconnaître les entités de type PER, ORG, LOC et MISC, et ces premières étiquettes ont été réaffectées aux entités attendues respectives PERSON, ORGANIZATION, LOCATION, alors qu'une stratégie externe au modèle à été implémentée pour les entités MISC détectées (cf. partie 4.4),

Enfin, cet outil est également capable de détecter diverses entités métiers comme des numéros de téléphone, des e-mails, des montants numériques, des monnaies et par conséquent des montants monétaires ou des entités de type DATE. La majorité de ces entités n'ont pas été représentées dans le cadre de ce challenge même si elles auraient pu faire partie de l'ensemble ID, en revanche ces montants monétaires sont devenus des RESOURCE. Cet outil permet également de *"projeter"* des lexiques et de leur affecter une étiquette d'entité spécifique. Toutes ces entités relativement classiques sont reconnues par des méthodes à base de règles et/ou motifs comme des expressions régulières.

4.2 Un modèle multilingue : NER2, Mistral-7B fine-tuné sur PER et LOC

Dans le cadre de travaux précédents, et toujours autour de ce besoin d'adapter un modèle à cette tâche de détection des entités nommées, nous avions envisagé plusieurs expérimentations de fine-tuning d'un modèle Mistral-7B (Jiang et al., 2023), un grand modèle de langage (LLM) proposé par la compagnie Mistral.AI. Ce modèle NER2 portait sur un jeu réduit d'entités : les seuls types PER et LOC. Ce fine-tuning a été réalisé indépendamment et préalablement à ce challenge par le biais de la plateforme cloud d'AWS. sur une machine *ml.p4d.24xlarg*e (durée de fint-tuning de 2h37min), enfin le modèle obtenu a été quantifié en 4 bits pour des raisons d'usabilité.

Les données d'entrainements étaient constituées d'un total de 20K lignes, sélectionnées aléatoirement parmi différents jeux de données suivant la décomposition suivante : 5K lignes en Français depuis WikiNER_fr (Jean-Baptiste, 2021b) , 1K de données internes provenant de notre univers du selfcare et de la relation client après une pseudonymisation (anonymisation par substitution "équivalente"), 3K en anglais depuis CoNLL-2003 (Tjong Kim Sang & De Meulder , 2003), et plusieurs ensembles de 3K en allemand, espagnol, en néerlandais depuis WikiNER, et enfin en langue arabe obtenu depuis une traduction automatique (services AWS translate) depuis WikiNER_fr. Ces données comprenaient également des variations de casse : majuscules

uniquement(11,3%), minuscules uniquement (7,2%), majuscules initiales sur chaque mot (3.5%), et des formes mélangées (78%). Une autre expérience aurait été d'ajouter des erreurs typographiques.

4.3 Un modèle pour les entités temporelles : NER3-TIME, Mistral-7B fine-tuné

Ce modèle a été spécifiquement entraîné en utilisant le jeu de données conçu et décrit dans la partie 3.2. Ce modèle a été entraîné localement, sans utilisation d'un service cloud, sur une machine dotée d'une carte GPU RTX 4090 avec 24 Go de VRAM. Il a également été quantifié en 4 bits via QLORA (Dettmers et al., 2023).

Contrairement au fine-tuning précédent nous nous sommes volontairement limité à un petit jeu de données d'apprentissage à 1K phrases comme déjà explicité en partie 3, probablement que les performances pourrait s'améliorer avec des données plus qualitatives et nombreuses mais cela était aussi une partie de nos expérimentations et de notre volonté de vérifier les performances obtenues depuis un jeu de données rapidement constitué.

Voici ci-dessous, le prompt très simple mis en oeuvre pour ce finetuning et qui montre clairement des pistes d'amélioration possible :

> input : Extract the TIME entities for the following text : {text}.
> output : [{'TIME', 'entité', ordre_d_apparition_entité}, [...]]
> ex : [{'TIME', 'demain', 1}, {'TIME', 'hier', 2}]

Enfin puisque ces outils NER2 et NER3-TIME proposent des prédictions sur les étiquettes de chaque token, un composant supplémentaire a été conçu pour récupérer les positions des occurrences des entités classifiées en prenant garde de gérer au mieux les chevauchements et inclusions.

4.4 NER4-HYBRID : des stratégies complémentaires et hybrides pour d'autres entités et adaptations spécifiques

Nous avions à notre disposition en sortie de l'outil interne NER1 des étiquettes MISC, et à la vue des données d'apprentissages fournies par les organisateurs du challenge, nous avons envisagé une stratégie complémentaire très basique pour ré-affecter l'étiquette MISC vers des entités en accord avec celui-ci. Faute de temps et de données annotées pour développer un véritable classifieur, un composant dédié a été développé rapidement. Il n'est pas à proprement possible de parler de classification statistique vu sa simplicité : une stratégie de vote a été implémentée depuis : des lexiques d'équipements militaire obtenu au travers du portail militaire de Wikipédia, des appels API ; des sites externes moteur de recherche avec comme mots clés *"Wikipedia + Military + Equipment + valeur de l'entité MISC"* qui transforment en EQUIPMENT ; des sites relevant de l'identification d'appareil aérien pour ID ; et enfin par défaut UNKNOWN. L'approche retenue initialement était de décider arbitrairement d'affecter tous ces MISC a l'étiquette EQUIPMENT, puisque cela semblait une correspondance prépondérante dans les données initiales. Quoi qu'il en soit, ces approches d'ingénierie restent très peu intéressantes à la fois du point de vue de nos travaux/besoins et du point de vue scientifique, puisqu'elles n'ont d'intérêt que de contribuer relativement arbitrairement à un score final.

De même une stratégie basée sur d'autres lexiques simples générés avec l'aide de l'interface web de GPT-4o avec des amorces telles que *{"attentat", "assassinat", "meurtre", "détournement d'avion"...}* ont également été projetés grâce à l'outil NER1, mais, là encore, nous n'avons pas mis d'efforts particuliers sur ces éléments, vu leur faible intérêt métier. Ils auraient cependant mérité des approches plus complexes, par exemple l'expression *"attaque aérienne"* suggérée par GPT-4o a été écartée lors de la relecture, car elle pourrait probablement autant qualifier un événement qu'un type particulier d'équipement (*"avion d'…"* ou *"missile d'…"*) et ils rappellent ainsi les limites d'une telle projection sans prise en compte du contexte sémantique.

Sur le même modèle un lexique de nationalités a été projeté avec comme étiquette *ORGANIZATION,* Ce choix était une première approximation pour compléter les ORG des entreprises, l'examen rapide des données fournies mettant en évidence une forte propension à généraliser un gentilé à son gouvernement associé comme son entité étatique. Il faut noter que notre première intention était de créer et d'étendre ce lexique en utilisant GPT-4o. Mais nous avons rapidement rencontré des limites dans ses capacités de génération, et notamment sur le concept de bi-nationalité comme *"franco-américain"* (une certaine uniformité, répétitions, et des hallucinations). Ce lexique a finalement été construit manuellement, en effet, il est plus évident de constituer ces ressources depuis des références telles que *Wikipedia* (notamment https://fr.wikipedia.org/wiki/Liste_de_gentil%C3%A9s). Enfin, cette ressource aurait pu être utilisée comme autant de marqueurs linguistiques pour d'autres systèmes à base d'apprentissage automatique ou à base de règles, mais nous n'avons pas exploré cette voie pour cette participation.

Enfin quelques adaptations spécifiques ont été faites dans le cadre de cette participation, lesquelles ont consisté principalement à intégrer certaines prépositions dans les bornes d'entités nommées comme la préposition *"au"* précédant un lieu comme dans *"au Mali"*. Cela n'était pas nos choix initiaux dans l'outil NER1, dont le rôle est d'identifier uniquement la partie "nom propre" (Pays).

5 Conclusion et perspectives

Notre participation au challenge EvalLLM2024 nous a permis d'explorer et de valider certaines de nos hypothèses concernant l'utilisation des Grands Modèles de Langages (LLMs) pour la création de jeux de données annotés, de mettre en évidence des limitations liées à l'annotation de données, mais également de valider notre stratégie de superpositions et de mélange d'outils complémentaires pour cette tâche de reconnaissance d'entité nommées (NER). Par ailleurs, nous avons confirmé des difficultés inhérentes à l'utilisation des LLMs pour la NER en termes de prompt engineering, notamment pour une mise en œuvre de few-shot learning/prompting, où demeure un évident besoin d'annotations humaines de qualité. Ce challenge rappelle en outre les difficultés liées à l'évaluation des résultats générés par un LLM ainsi que l'évident besoin de méthodologies et de métriques en adéquation. En conséquence, ces travaux nous permettent d'envisager de nouvelles perspectives dans ces différents domaines.

Enfin, les résultats obtenus l'ont été en considérant un petit sous-ensemble d'entités nommées, et auraient probablement été améliorés en intégrant davantage de typologies d'entités, ce que nous envisagerions dans le cadre d'une future participation. D'autant que, comme ces extensions de couvertures, d'autres travaux et explorations restent à mener.

Références

Chen B., Zhang Z., Langrené N., & Zhu S. (2024). Unleashing the potential of prompt engineering in Large Language Models: a comprehensive review. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2310.14735

Delestre C., & Amar A. (2022, juillet). DistilCamemBERT : une distillation du modèle français CamemBERT. CAp (Conférence Sur l'Apprentissage Automatique). Retrieved from https://hal.archives-ouvertes.fr/hal-03674695

Dettmers T., Pagnoni A., Holtzman A., & Zettlemoyer L. (2023). QLoRA: Efficient Finetuning of Quantized LLMs (Version 1). *arXiv*. https://doi.org/10.48550/ARXIV.2305.14314

Felkner V. K., Thompson J. A., & May J. (2024). GPT is Not an Annotator: The Necessity of Human Annotation in Fairness Benchmark Construction. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2405.15760

Huang Y., Tang K., & Chen M. (2024). Leveraging Large Language Models for Enhanced NLP Task Performance through Knowledge Distillation and Optimized Training Strategies. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2402.09282

Jean-baptiste (2021a). CamemBERT-NER. https://huggingface.co/Jean-Baptiste/camembert-ner

Jean-baptiste (2021b). WikiNER_fr. https://huggingface.co/datasets/Jean-Baptiste/wikiner_fr

Jiang A. Q., Sablayrolles A., Mensch A., Bamford C., Chaplot D. S., Casas D. De Las, Bressand F., Lengyel G., Lample G., Saulnier L., Lavaud L. R., Lachaux M.-A., Stock P., Scao T. L., Lavril T., Wang T., Lacroix T., & Sayed W. E. (2023). Mistral 7B (Version 1). *arXiv*. https://doi.org/10.48550/ARXIV.2310.06825

Lannelongue L., Grealey J., & Inouye M. (2020). Green Algorithms: Quantifying the carbon footprint of computation (Version 5). *arXiv*. https://doi.org/10.48550/ARXIV.2007.07610

Martin L., Muller B., Suárez P. J. O., Dupont Y., Romary L., De La Clergerie É. V., … Sagot B. (2020). CamemBERT: a Tasty French Language Model. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

Nothman J., Ringland N., Radford W., Murphy T., & Curran J. R. (2013). Learning multilingual named entity recognition from Wikipedia. In Artificial Intelligence (Vol. 194, pp. 151–175). Elsevier BV. https://doi.org/10.1016/j.artint.2012.03.006

Tjong Kim Sang E. F., De Meulder F.. (2003). Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003, pages 142– 147.

Wang S., Sun X., Li X., Ouyang R., Wu F., Zhang T., … Wang, G. (2023). GPT-NER: Named Entity Recognition via Large Language Models. *arXiv [Cs.CL]*. Retrieved from http://arxiv.org/abs/2304.10428

Annexes

System	Human	Substitution
Ton rôle est d'écrire une ou plusieurs histoires en utilisant l'intégralité des mots dans la liste ci-dessous : {entities} Cette liste est une liste d'entités du type TIME.	Dans ces histoires encadre les entités de la liste fournie avec les balises <TIME></TIME>	{entities} est remplacé par une liste d'entités temporelles contenant notamment des entités du jeu d'entraînement.

ANNEXE 1 : Prompt de génération de textes à partir d'une liste d'entités donnée.

System	Human	Substitution
En tant qu'expert dans la tâche de NER (Named Entity Recognition), tu te spécialises dans la détection des entités de type TIME. {rules} Réalise une analyse méticuleuse de l'exemple ci-dessous afin de comprendre ce que sont les entités de type TIME. Les entités de type TIME sont tous les mots ou groupes de mots encadrés par les balises <TIME></TIME>. <EXEMPLE> {text_ex} </EXEMPLE>	Utilise ta connaissance des entités TIME pour annoter, à l'aide des balises <TIME></TIME>, toutes les entités de type TIME dans le texte suivant : <TEXTE> {text} </TEXTE>"	{rules} est remplacé par notre description de l'entité TIME, inspirée par le guide d'annotation. {text_ex} est le texte généré à l'aide du Prompt 1 et dans lequel les entités temporelles sont précédées de la balise <TIME> et suivies de la balise </TIME>. {text} est le texte à annoter par le LLM.

ANNEXE 2 : Prompt de génération de textes annotés.

Rules
Les entitées temporelles peuvent être : - DATE : toutes les dates absolues ou relatives - PERIODE : les durées et les périodes de temps. - MT : pour Marqueur Temporel, ce sont toutes les marques de fréquence comme "souvent", "parfois", "régulièrement" et les connecteurs logiques comme "d'abord", "premièrement", "puis", "enfin", "ensuite". Lorsque l'entité est introduite par une préposition, on annote l'ensemble de la préposition.

ANNEXE 3 : Extrait de prompt avec subdivisions de TIME

https://www.jetphotos.com/photo/keyword/
https://www.flightradar24.com/data/aircraft/

ANNEXE 4 : Site d'identification/immatriculation d'appareils aérien (MISC vers ID)

https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:V%C3%A9hicule_militaire_post-guerre_froide
https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:V%C3%A9hicule_militaire_l%C3%A9ger
https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:V%C3%A9hicule_blind%C3%A9_de_transport_de_troupes
https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:V%C3%A9hicule_militaire_polonais

ANNEXE 5 : Quelques pages de Wikipedia "Portail Militaire" (MISC vers EQUIPMENT)