Actes de CORIA-TALN 2026
Nantes, France, 29 juin au 3 juillet 2026

Approche guidée par la confiance pour l'annotation automatique d'un corpus de pré-entraînement en extraction d'événements

Salim Abdou Daoura Sondes Bannour Souihi Romaric Besançon
Olivier Ferret
Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
{salim.abdoudaoura,sondes.souihi,romaric.besancon,olivier.ferret}@cea.fr

RÉSUMÉ

Les méthodes d'extraction d'information en zero-shot ou few-shot dépendent de vastes corpus annotés, qui restent rares et coûteux. Nous proposons une méthode d'annotation automatique utilisant les grands modèles de langage (LLM) pour annoter des données réelles et évaluer leur fiabilité via des scores de confiance. Nous avons ainsi créé Omnivent, un corpus généraliste d'événements en anglais couvrant 38 859 types d'événements et 9 981 rôles d'arguments. Nous introduisons également GLEE, un modèle transformeur bidirectionnel pour la détection d'événements (DE) et l'extraction d'arguments (EAE). Évalué sur neuf benchmarks couvrant sept domaines, GLEE surpasse en zero-shot les meilleures approches à base de LLM, tout en étant environ 25 fois plus compact.

ABSTRACT

A confidence-guided approach for annotating a pretraining corpus for event extraction.

Zero-shot or few-shot information extraction methods rely on extensive annotated corpora, which remain rare and costly. We propose an automatic annotation method using Large Language Models to annotate real data and assess their reliability through confidence scores. This led us to create Omnivent, an English generalist event corpus covering 38,859 event types and 9,981 argument roles. We also introduce GLEE, a bidirectional transformer model for event detection (ED) and argument extraction (EAE). Evaluated on nine benchmarks across seven domains, GLEE outperforms the best LLM-based approaches in zero-shot settings, while being approximately 25 times more compact.

MOTS-CLÉS : Extraction d'événements, annotations synthétiques, zero-shot.

KEYWORDS: Event extraction, synthetic annotations, zero-shot.

1 Introduction

L'extraction d'événements (EE), divisée en détection de déclencheurs (*triggers*) (DE) et extraction d'arguments (EAE), constitue une tâche fondamentale en extraction d'information. Malgré son importance, elle reste limitée par la dépendance aux corpus annotés manuellement, coûteux et peu extensibles. Si des ressources comme ACE 2005 (Doddington *et al.*, 2004) ou MAVEN (Wang *et al.*, 2020) ont posé les bases du domaine, leur couverture thématique reste modeste. Des efforts récents comme GLEN (Li *et al.*, 2023) augmentent la diversité des types d'événements mais souffrent de bruit au niveau de leurs annotations et d'un manque de données sur les arguments.

L'introduction des grands modèles de langue de type décodeur (LLM) a ouvert de nouvelles perspec-

Actes des 33ème Conférence sur le Traitement Automatique des Langues Naturelles.
Volume 1 : articles scientifiques originaux, pages 145–164.
Cette œuvre est mise à disposition sous licence Attribution 4.0 International.

tives pour l'EE, à commencer par leurs capacités de réalisation de tâches en zero ou few-shot. De ce point de vue, beaucoup de travaux se sont focalisés sur l'optimisation des instructions de définition des tâches, que ce soit en une passe (Gao *et al.*, 2023; Huang *et al.*, 2024), via une série de questions binaires (Lyu *et al.*, 2021) ou, comme ChatIE, par des échanges multi-tours (Wei *et al.*, 2024). Cai *et al.* (2024) se concentrent quant à eux sur l'intégration d'une définition des événements tandis que Parekh *et al.* (2025) s'appuient sur les capacités de raisonnement des LLM récents.

Mais l'intérêt des LLM réside également dans leur capacité de génération de données synthétiques (Ma *et al.*, 2024; Chen *et al.*, 2024). Toutefois, les approches développées dans ce contexte reposent principalement sur la production de textes synthétiques et de leurs annotations associées, sans exploiter les données textuelles disponibles, et peuvent en outre introduire des redondances, voire des biais, dans les ensembles d'entraînement. Parallèlement, des approches en reconnaissance d'entités nommées (REN) ont montré que le pré-entraînement sur des données réelles annotées automatiquement favorise une meilleure généralisation (Zhou *et al.*, 2024; Zaratiana *et al.*, 2024).

En nous appuyant sur la capacité des LLM à auto-évaluer la fiabilité de leurs sorties (Xiong *et al.*, 2024), nous proposons une méthodologie d'annotation automatique en deux étapes, génération et sélection par score de confiance, appliquée à des textes bruts, ce que l'on peut voir également comme une forme de distillation des LLM utilisés. Cette approche nous a permis de constituer Omnivent, un corpus généraliste de grande taille couvrant la DE et l'EAE sans inventaire de types prédéfini. Pour exploiter cette ressource en tant que corpus de pré-entraînement, nous introduisons GLEE (*Generalist Label-semantic Event Extraction*), un modèle transformeur bidirectionnel fondé sur les segments (*spans*). GLEE traite à la fois la DE et l'EAE en alignant la sémantique des étiquettes avec des segments candidats. Il s'inscrit ainsi dans la lignée d'approches fondées sur la sémantique des étiquettes, initialement développées pour la classification de textes (Mueller *et al.*, 2022) puis appliquées à la REN en la formulant comme un problème d'appariement entre représentations d'étiquettes et de segments textuels. Ma *et al.* (2022) et Borovikova *et al.* (2024) ont développé cette approche dans un contexte few-shot tandis que Zaratiana *et al.* (2024) s'est intéressé au pré-entraînement d'un modèle de REN zero-shot couvrant un large spectre d'entités.

Nos évaluations sur neuf corpus montrent que le pré-entraînement sur Omnivent permet à GLEE de surpasser les LLM en configuration *zero-shot*, c'est-à-dire sans données d'entraînement liées aux données de test. De ce point de vue, le pré-entraînement sur Omnivent peut aussi être vu comme la mise en œuvre d'une approche cross-domaine (entre le corpus de pré-entraînement et les données de test), même la qualification zero-shot est plus usitée dans ce contexte. Nos contributions sont triples :

— une méthode d'annotation automatique de données réelles reposant sur des scores de confiance ;
— le corpus Omnivent, une ressource de grande taille en anglais pour le pré-entraînement en EE ;
— le modèle GLEE, fondé sur l'exploitation de la sémantique des étiquettes et atteignant l'état de l'art en *zero-shot* pour la DE et l'EAE.

2 Méthode

2.1 Construction du corpus Omnivent

Des études récentes (Chen *et al.*, 2024; Ma *et al.*, 2024) ont exploré l'utilisation des LLM pour la génération de données d'entraînement synthétiques dédiées à l'extraction d'événements dans des

contextes supervisés. À l'inverse, nous proposons une stratégie s'appuyant sur des corpus réels, où les LLM sont utilisés pour annoter le texte de manière comparable à des annotateurs humains, évitant ainsi le recours à la génération synthétique. Pour construire Omnivent, nous avons extrait un sous-ensemble du jeu de données CC-News, constitué d'articles de presse anglophones publiés entre janvier 2017 et décembre 2019. Notre objectif est d'extraire à la fois les déclencheurs et leurs arguments. Notre approche se décline en deux étapes principales. Premièrement, lors d'une phase de classification, le modèle détermine si un segment de texte donné décrit un événement. Deuxièmement, durant la phase d'annotation, les segments pertinents sont soumis à un LLM qui identifie les déclencheurs d'événements et annote les arguments associés. Afin d'optimiser la qualité de l'annotation, nous demandons au LLM de fournir un score de confiance pour chaque prédiction. Ces scores permettent de sélectionner les annotations les plus probantes, améliorant ainsi la fiabilité et la précision globales des données produites. Nous utilisons Llama-3-70B-Instruct (Dubey *et al.*, 2024) pour les phases de classification et d'annotation. Les instructions pour chaque phase sont présentées en annexe A.1.

2.1.1 Phase de classification

La phase de classification vise à identifier les phrases décrivant un événement. À partir d'une phrase donnée, nous sollicitons le LLM pour obtenir une décision binaire indiquant la présence ou l'absence d'un événement, accompagnée d'un score de confiance auto-évalué compris entre 0 et 1. Notre approche repose sur l'hypothèse que les LLM sont capables d'exprimer une incertitude calibrée concernant leurs propres réponses (Xiong *et al.*, 2024; Lin *et al.*, 2022). Afin de garantir une précision d'annotation élevée, nous ne conservons que les phrases pour lesquelles le LLM produit une décision positive avec un score de confiance d'au moins 0,9. Ce seuil haut permet de compenser la tendance avérée des LLM à manifester une confiance excessive dans leurs réponses (Lin *et al.*, 2022). L'efficacité de cette stratégie de filtrage fondée sur la confiance est par ailleurs étayée par les résultats d'une évaluation humaine, détaillés à la section 2.1.3.

2.1.2 Phase d'annotation

Pour les phrases identifiées comme contenant des événements, nous sollicitons le LLM afin d'extraire les déclencheurs ainsi que leurs types d'événements correspondants, en plus des arguments et de leurs rôles associés. À l'instar de la phase de classification, le LLM est interrogé de façon à fournir un score de confiance pour chaque événement extrait. Nous ne conservons que les annotations présentant un score de confiance supérieur ou égal à 0,9. Pour structurer ces annotations, nous adoptons un schéma hiérarchique à deux niveaux pour les types d'événements, comprenant une catégorie de haut niveau et un sous-type plus fin. Cette conception permet de saisir à la fois les schémas d'événements généraux et les distinctions plus subtiles entre des sous-types d'événements apparentés. Notre nomenclature reflète la nature hiérarchique de la sémantique événementielle, en cohérence avec les référentiels établis tels que ACE 2005, MEE (Veyseh *et al.*, 2022) et FewEvent (Deng *et al.*, 2020).

2.1.3 Caractérisation de la qualité du jeu de données

Nous avons fait appel à 4 annotateurs familiers avec l'extraction d'événements pour évaluer les données annotées en utilisant notre méthodologie pour différentes valeurs de seuil. Nous avons choisi 100 passages classifiés et annoté les événements ayant des scores de confiance de 0,8 et 0,9.

	Questions de qualité	0.8	0.9
(1)	La phrase porte-t-elle sur un ou plusieurs événements ?	57.6	**87.1**
(2)	Le déclencheur décrit-il une occurrence d'événement ?	65.2	**84.1**
	La catégorie de l'événement est-elle pertinente ?	66.7	**81.1**
	Les arguments annotés sont-ils pertinents par rapport à l'événement ?	94.6	**95.5**
	Tous les arguments pertinents sont-ils annotés ?	74.2	**84.8**

TABLE 1 – Évaluations par les experts. 0.8 et 0.9 : niveau de confiance lors des expériences (1) et (2).

Nous avons réalisé une expérience pour chaque étape de notre processus d'annotation (cf. interfaces d'évaluation à l'annexe A.5).

Dans l'expérience 1, qui évalue la classification sur une seule question, nous rapportons l'accord moyen entre la classification du modèle et les jugements des annotateurs pour un score de confiance donné. Dans l'expérience 2, qui évalue l'annotation à travers quatre questions, les annotateurs peuvent répondre positivement ou négativement à la question proposée. Nous rapportons la moyenne des réponses pour chaque question. Les questions et résultats sont présentés dans le tableau 1.

Les annotateurs humains étaient d'accord avec la classification proposée à 87,1 % pour un score de confiance de 0,9 et à 57,6 % pour un score de confiance de 0,8. De même, dans l'expérience 2, les annotateurs ont systématiquement répondu plus positivement à toutes les questions pour l'annotation d'événements avec un score de confiance de 0,9. Ces résultats démontrent une amélioration nette de la qualité perçue de l'annotation aux niveaux de confiance les plus élevés.

2.1.4 Caractéristiques du jeu de données Omnivent

Notre jeu de données annoté de manière automatique, Omnivent, comprend 101 000 textes, incluant 199 716 déclencheurs d'événements répartis entre 38 859 types d'événements distincts. Cette envergure rend notre corpus dix fois plus important, en nombre de types d'événements, que les jeux de données précédents (Li *et al.*, 2023). Concernant les arguments, notre jeu de données en dénombre 522 703, couvrant 9 981 rôles distincts. Des statistiques complémentaires sont disponibles dans le tableau 2 et une analyse de la distribution des types d'événements est donnée à l'annexe A.3.

Omnivent englobe une vaste gamme de types d'événements, dont les cinq plus fréquents sont Communication:Statement, Conflict:Attack, Competition:Victory, State:Existence et Life:Death. Ces cinq types représentent à eux seuls 5 % du total des instances d'événements. Par ailleurs, les cinq arguments les plus récurrents sont Location, Person, Time, Organization et Event, totalisant 30 % des instances d'arguments. Les types d'événements comme les rôles d'arguments présentent une distribution caractéristique en « longue traîne », conformément à la répartition observée avec les annotations humaines. Quelques exemples illustratifs d'Omnivent sont donnés à l'annexe A.4.

Jeu de données	DE	EAE	Fin	Doc.	Phr.	ÉvTypes	Évt.	Rôles	Arg.
ACE05 (Doddington *et al.*, 2004)	✓	✓	✓	599	20 920	33	5 348	22	8097
MAVEN (Wang *et al.*, 2020)	✓	✗	✓	3 623	40 473	168	96 897	-	-
GLEN (Li *et al.*, 2023)	✓	✗	✓	6 224	208 454	3 465	204 065	-	-
DocEE (Tong *et al.*, 2022)	✓	✓	✗	27 485	749 568	59	27 485	356	180 528
Omnivent	✓	✓	✓	58 064	101 000	38 859	199 716	9 981	522 703

TABLE 2 – Statistiques concernant notre jeux de données comparées à celles des jeux de données existants. **Doc.** : nombre de documents ; **Phr.** : nombre de phrases ; **ÉvTypes** : nombre de types d'événements ; **Évt** : nombre de déclencheurs d'événements ; **Rôles** : nombre de types d'arguments ; **Arg.** : nombre d'arguments ; **Fin** : annotation à grain fin.

2.2 Modèle d'extraction d'événements

Nous proposons GLEE, une architecture conçue pour effectuer conjointement la DE et l'EAE selon une approche en séquence. GLEE est un modèle à base de segments sensible à la sémantique des étiquettes, qui calcule un score de similarité entre des segments de texte et des étiquettes prédéfinies. Il se compose de deux modules distincts, l'un dédié à la DE et l'autre à l'EAE. GLEE s'appuie sur les avancées récentes en REN fondées sur les segments (Ma *et al.*, 2022; Zaratiana *et al.*, 2024).

2.2.1 Modèle de détection d'événements

Nous abordons la DE comme une tâche d'appariement entre segments et étiquettes dont l'objectif est d'identifier les segments de texte correspondant à des catégories d'événements spécifiques. Les catégories d'événements ainsi que les segments de texte sont projetés dans un espace sémantique partagé par le biais d'un encodeur de type transformeur. Cette architecture s'appuie sur les principes introduits par GLiNER (Zaratiana *et al.*, 2024) que nous adaptons pour répondre aux spécificités de la DE, en particulier du point de vue de ses entrées.

Entrées de la détection d'événements. L'entrée de notre modèle de DE est constituée de la concaténation des trois éléments suivants :

— une instruction composée de tous les types d'événements candidats séparés par le token spécial [EVT] ;
— le token spécial [SEP], agissant comme un délimiteur entre l'instruction et le texte à analyser ;
— les tokens du texte à analyser.

$$[\text{EVT}]\ t_0\ [\text{EVT}]\ t_1\ \ldots\ [\text{EVT}]\ t_{M-1}\ [\text{SEP}]\ x_0, x_1, \ldots x_{N-1}$$

2.2.2 Modèle d'extraction d'arguments d'événements

GLEE formule l'EAE comme une tâche d'appariement fondée sur la similarité entre des paires (déclencheur, segment) et des représentations de rôles d'arguments. Le modèle détermine quels segments textuels font office d'arguments pour un déclencheur d'événement donné et leur attribue les rôles sémantiques appropriés. L'architecture se compose de trois composantes principales :

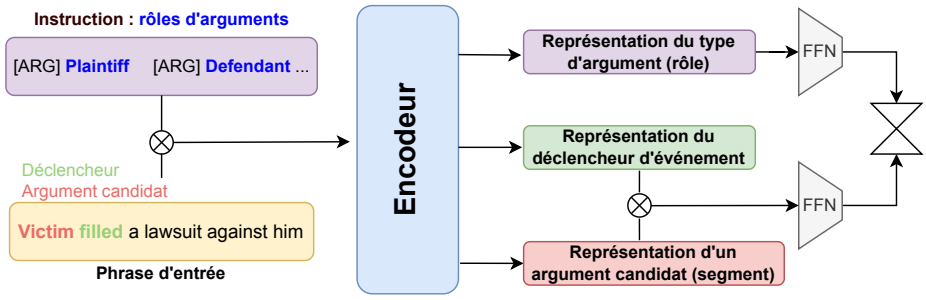

FIGURE 1 – Architecture du modèle GLEE pour l'EAE.

— un encodeur bidirectionnel encodant de façon jointe le texte d'entrée et les étiquettes de rôles d'arguments dans un espace de représentation commun ;
— une couche de représentation des paires (segment, déclencheur) ;
— une couche de représentation des rôles d'arguments.

Le modèle calcule un score de similarité entre chaque paire (déclencheur, segment) et la représentation de chaque rôle candidat et retient finalement le rôle ayant le plus haut score. La figure 1 illustre l'architecture de ce modèle et les interactions entre ses différents constituants.

Séquence d'entrée pour l'extraction d'arguments. La séquence d'entrée du modèle est la concaténation de 3 éléments :

— une liste des rôles candidats séparés par un token spécial appris [ARG] ;
— un token spécial [SEP] délimitant les rôles candidats du texte ;
— la liste des tokens du texte d'entrée.

[ARG] t_0 [ARG] t_1 … [ARG] $t_{M\text{-}1}$ [SEP] $x_0, x_1, \ldots x_{N\text{-}1}$

Représentation des tokens et des segments. La séquence d'entrée est passée à l'encodeur pour produire une représentation de chacun de ses tokens. Notons h la partie correspondant à la représentation du texte d'entrée :

$$h = \{h_i\}_0^{N-1} \in \mathbb{R}^{N \times D} \tag{1}$$

où N est le nombre de mots du texte d'entrée et D la dimension de représentation du modèle (pour les mots découpés en plusieurs tokens, nous utilisons la représentation du premier token). Un segment commençant au i^e mot et se terminant au j^e mot est représenté par :

$$S_{ij} = FFN(h_i \otimes h_j) \quad \forall (i,j) \in S \tag{2}$$

où $\otimes$ désigne l'opération de concaténation et FFN est un réseau *Feed Forward* à deux couches. Étant donné $T = S_{ab}$ le segment correspondant au déclencheur pour un événement donné, S est défini par :

$$S = \{(i,j) \mid [i,j] \cap [a,b] = \emptyset\} \tag{3}$$

Représentation des paires. La représentation d'une paire associant un segment S_{ij} et un déclencheur T est faite par la simple concaténation des deux représentations.

$$U_{ijT} = FFN(S_{ij} \otimes T) \quad \forall (i,j) \in S \tag{4}$$

Représentation des arguments. Comme la représentation des segments, la représentation des arguments est obtenue à partir de l'encodeur : en notant p la représentation de l'argument candidat par l'encodeur, sa représentation finale est obtenue par l'application d'un réseau FFN à 2 couches :

$$q = FFN(p) = \{q_i\}_0^{M-1} \in \mathbb{R}^{M \times D} \tag{5}$$

où M désigne le nombre de rôles candidats et D, la dimension cachée.

Score de similarité. Un score de similarité est ensuite calculé entre chaque représentation d'argument et chaque représentation de paire :

$$score(q_i, U_{ijT}) = \sigma(q_i^T \cdot U_{ijT}) \in \mathbb{R} \tag{6}$$

avec σ, une fonction d'activation sigmoïde. Chaque score représente la probabilité qu'un segment S_{ij} soit un argument pour le rôle q_i associé au déclencheur T.

3 Cadre d'évaluation

Jeux de données. Nous avons mené une évaluation exhaustive sur 9 jeux de données d'extraction d'événements couvrant 7 domaines distincts. Les jeux de données incluent : ACE 2005, couvrant des textes d'actualités, MLEE (Pyysalo *et al.*, 2012) et Genia2011 (Kim *et al.*, 2011) dans le domaine biomédical, M2E2 (Li *et al.*, 2020) pour du contenu multimédia, CASIE (Satyapanich *et al.*, 2020) axé sur la cybersécurité, PHEE (Sun *et al.*, 2022) ciblant la pharmacovigilance, SPEED (Parekh *et al.*, 2024) centré sur l'épidémiologie, MEE (Pouran Ben Veyseh *et al.*, 2022) construit à partir d'entrées Wikipedia et MAVEN qui offre une couverture large de domaines. Pour assurer la comparabilité avec les travaux antérieurs, nous suivons les protocoles de prétraitement et d'évaluation introduits dans TextEE (Huang *et al.*, 2024), ce qui maintient la cohérence sur toutes ces évaluations.

Modèles de référence. Nous évaluons GLEE vis-à-vis de plusieurs modèles de référence adaptés à chaque configuration de tâche en retenant des modèles véritablement zero-shot [1]. En outre, nous prenons comme référence les résultats des modèles les plus proches en taille de nos modèles. Pour la détection zero-shot d'événement, nous considérons deux approches récentes : 1. DiCoRe (Parekh *et al.*, 2025), fondé sur une approche de raisonnement LLM divergent-convergent ; 2. DivED (Cai *et al.*, 2024), qui formule la détection d'événements comme un processus conversationnel multi-tours.

1. (Lyu *et al.*, 2021) s'affiche par exemple zero-shot mais utilise la partie développement des données d'évaluation pour fixer la valeur de ses hyperparamètres.

Pour la détection zero-shot d'arguments, nous considérons trois approches récentes : 1. GuidelineEE (Srivastava *et al.*, 2025), qui exploite les directives d'annotation, convertissant l'extraction d'événements en une tâche structurée de génération de code Python guidée par des descriptions de schémas textuels ; 2. DecomposeEE (Shiri *et al.*, 2024), qui décompose l'extraction d'arguments en phases de détection utilisant une augmentation dynamique et sensible au schéma de la récupération afin de réduire les hallucinations ; 3. AEC (Guo *et al.*, 2025), qui traite l'extraction d'événements comme un processus structuré et itératif de génération de code avec des agents.

Métriques. Nous adoptons les mêmes métriques d'évaluation que dans les travaux antérieurs (Lin *et al.*, 2020). Pour les déclencheurs d'événements, TI (*trigger identification*) mesure la justesse de la détection des segments, tandis que TC (*trigger classification*) évalue à la fois l'identification des segments et la classification du type d'événement. Pour l'évaluation des arguments, nous utilisons AI (*argument identification*), qui évalue l'identification correcte des segments d'arguments associés aux déclencheurs prédits, et AC (*argument classification*), qui nécessite en outre l'assignation correcte des rôles sémantiques. Toutes les métriques sont rapportées en utilisant des scores F1 micro-moyennés. Pour mieux capturer les performances à travers divers types d'événements, nous rapportons également des scores F1 en macro-moyenne pour TC et AC (notés macTC et macAC).

Détails de l'entraînement. Nous adoptons DeBERTa-v3-large (He *et al.*, 2023) comme encodeur, étant donné ses bonnes performances générales sur des tâches d'extraction d'information. Chaque composant de notre architecture est entraîné indépendamment. Le pré-entraînement sur le jeu de données Omnivent nécessite environ 6 heures sur une seule carte graphique NVIDIA A100. Le modèle DE a été entraîné pendant 8 époques, tandis que le modèle EAE a été entraîné pendant 3 époques, tous deux avec une taille de *batch* de 16 et un taux d'échantillonnage négatif de 1. Nous avons utilisé une stratégie de taux d'apprentissage à deux niveaux, assignant un taux d'apprentissage plus bas à l'encodeur (1e-5) et un taux plus élevé aux paramètres restants du modèle (3e-5).

4 Évaluation

4.1 Efficacité du pré-entraînement sur Omnivent

Pour évaluer l'efficacité de notre méthodologie de génération de données, nous montrons que le pré-entraînement sur Omnivent mène à une performance supérieure en détection d'événements en zero-shot par rapport à des données annotées par des humains.

Pré-entraînement sur Omnivent *vs* données annotées manuellement. Pour évaluer la qualité de nos annotations synthétiques, nous comparons le pré-entraînement sur Omnivent au pré-entraînement sur deux ensembles de données annotées par des humains largement utilisés : ACE05 et MAVEN. La performance en zero-shot de GLEE pré-entraîné sur chacun de ces jeux de données annotés est alors évaluée sur les autres benchmarks sans ajustement spécifique à la tâche. Comme le montre le tableau 3, le modèle pré-entraîné sur Omnivent obtient une performance en zero-shot supérieure à celle de ses homologues pré-entraînés sur ACE05 et MAVEN. Plus précisément, Omnivent amène une amélioration moyenne de 2,7 points de F1 par rapport à MAVEN (à une échelle de données

Dataset	ACE05			MAVEN			Omnivent (28k)			Omnivent (full)		
	TI	**TC**	**macTC**	**TI**	**TC**	**macTC**	**TI**	**TC**	**macTC**	**TI**	**TC**	**macTC**
ACE05	–	–	–	36,6	27,3	25,1	33,1	27,6	28,6	**36,7**	**31,0**	**27,4**
M2E2	**36,2**	**34,5**	**35,3**	19,7	17,4	14,8	25,7	22,0	21,5	28,4	25,8	25,8
MLEE	14,2	4,2	5,4	**34,6**	**21,8**	**14,7**	32,8	14,7	9,5	33,6	16,2	9,8
Genia2011	19,3	10,6	15,6	**30,0**	**24,2**	**24,1**	25,8	17,9	25,8	28,3	20,3	20,5
SPEED	42,5	38,3	29,5	**48,1**	**38,8**	**34,6**	37,5	33,1	30,3	35,3	29,9	27,3
MEE-en	**65,5**	**60,8**	**54,1**	40,7	34,7	25,8	44,3	40,1	30,0	44,6	40,6	32,9
CASIE	9,8	8,2	8,0	11,6	10,2	9,7	16,4	14,5	14,9	**17,6**	**16,3**	**16,4**
PHEE	10,8	6,6	8,0	24,7	15,3	14,4	**45,8**	**43,6**	**36,0**	45,2	42,7	33,0
MAVEN	41,9	11,8	9,4	–	–	–	**61,4**	24,1	24,2	59,0	**25,3**	**24,2**
Moyenne	30,0	21,9	20,7	30,8	23,7	20,7	35,9	26,4	23,6	**36,5**	**27,6**	**24,1**

TABLE 3 – Performances (score F1) de la DE zero-shot pour **ACE05** et **MAVEN** : pré-entraînement de GLEE sur l'ensemble d'entraînement de ces jeux de données. **Omnivent (28k)** : pré-entraînement de GLEE sur un sous-ensemble d'Omnivent de la même taille que l'ensemble d'entraînement de MAVEN. **Omnivent (full)** : pré-entraînement de GLEE sur la totalité d'Omnivent.

équivalente) et de 5,7 points de F1 par rapport à ACE05. Ces gains suggèrent qu'Omnivent favorise une meilleure généralisation et une robustesse inter-domaine, probablement en raison de sa diversité syntaxique. Bien que le pré-entraînement sur ACE05 montre une forte performance sur le benchmark MEE intra-domaine, ce gain ne se retrouve pas dans des domaines plus spécifiques tels que la cybersécurité (CASIE) et la pharmacovigilance (PHEE), ce qui indique des signes de surapprentissage potentiel sur le domaine. En revanche, l'approche d'annotation ouverte adoptée pour Omnivent atténue un tel surapprentissage en exposant le modèle à une gamme plus large de motifs linguistiques pour les types d'événements, le rendant ainsi bien adapté au pré-entraînement. On remarque que l'entraînement sur la totalité d'Omnivent amène un gain par rapport à l'utilisation d'une partie seulement des données, même si ce gain est moins marqué que la différence entre Omnivent et les autres jeux de données. Une étude plus détaillée de la dynamique de performance par rapport à la taille des données de pré-entraînement est fournie en annexe A.2.

Détection d'événements zero-shot. Les résultats du tableau 4 indiquent que le pré-entraînement sur Omnivent surpasse les approches DiCoRe et DivED, fondés sur des LLM. GLEE surpasse systématiquement DivED et DiCoRe sur 3 des 5 jeux de données, malgré le fait d'être significativement plus léger (environ 25 fois plus petit que DiCoRe, qui est construit sur LLaMA 8B). De plus, GLEE ne nécessite aucune entrée supplémentaire, telle que des définitions ou descriptions d'événements, qui demandent des informations spécifiques à la tâche. Il fonctionne uniquement en utilisant des étiquettes de types d'événements, ce qui le rend beaucoup plus léger, évolutif et plus facile à intégrer dans des cadres applicatifs réels où de telles informations supplémentaires peuvent ne pas être disponibles.

Impact des scores de confiance. La construction d'Omnivent s'appuie sur des scores de confiance directement produits par les LLM, qui pourraient être mal calibrés par rapport à la qualité réelle des annotations. Afin d'évaluer l'impact de ce choix, nous avons supprimé indépendamment puis conjointement le filtrage basé sur la confiance aux étapes de classification et d'annotation. Pour chaque configuration, nous avons généré un nouveau jeu de données, pré-entraîné GLEE dessus, puis évalué les performances en DE zero-shot sur les neuf benchmarks.

Dataset	GLEE		DiCoRE		DivED	
	TI	TC	TI	TC	TI	TC
ACE05	36,7	31,0	**40,3**	**36,3**	34,2	29,1
M2E2	**28,4**	**25,8**	–	–	23,4	22,0
MLEE	33,6	**16,2**	–	–	–	–
Genia2011	**28,3**	**20,3**	25,8	15,4	–	–
SPEED	35,3	**29,9**	**35,5**	23,6	–	–
MEE-en	**44,6**	**40,6**	–	–	27,1	26,0
CASIE	17,6	16,3	**18,5**	**16,8**	–	–
PHEE	**45,2**	**42,7**	–	–	–	–
MAVEN	**59,0**	**25,3**	53,5	14,4	–	–
Moyenne	**36,5**	**27,6**	–	–	–	–

TABLE 4 – Performances (score F1) en DE zero-shot. **TI** : identification du déclencheur, **TC** : classification du déclencheur. **GLEE** : GLEE avec un encodeur DeBERTa-v3-large. **DiCoRe** : résultats avec Llama3-8B issus de (Parekh *et al.*, 2025). **DivED** : résultats de (Cai *et al.*, 2024).

Les résultats présentés à la figure 2 montrent que la suppression du score de confiance de classification a entraîné des baisses respectives de 2,1, 2,3 et 1,2 points de F1 pour TI, TC et macro TC. De manière similaire, l'élimination du score de confiance d'annotation a conduit à des diminutions respectives de 2,8, 1,4, et 0,8 points de F1.

Lorsque les deux scores de confiance sont supprimés, la dégradation est la plus marquée, avec des baisses de 3,2 points en TI, 2,5 points en TC et 1,5 point en macro TC. Ces résultats mettent en évidence les bénéfices complémentaires de l'utilisation des scores de confiance aux deux étapes. Le filtrage fondé sur la confiance est essentiel pour maintenir la qualité des annotations, en permettant au modèle de sélectionner des phrases plus informatives et des assignations d'étiquettes plus fiables.

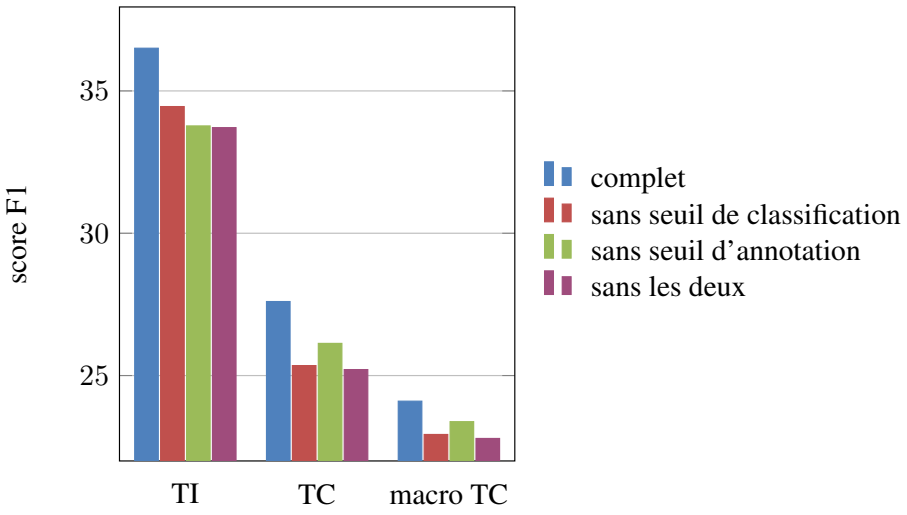

FIGURE 2 – Performance moyenne sur 9 benchmarks des différentes versions de notre jeu de données.

Modèle	ACE05			M2E2			PHEE			CASIE		
	AI	**AC**	**macAC**	**AI**	**AC**	**macAC**	**AI**	**AC**	**macAC**	**AI**	**AC**	**macAC**
GuidelineEE*	20.6	17.4	–	–	–	–	–	–	-	**31.5**	**30.8**	–
DecomposeEE*	23.6	22.9	–	–	–	–	–	–	–	30.9	27.6	–
AEC*	26.7	25.2	–	–	–	–	–	–	–	28.8	26.7	–
GLEE	**34.1**	**25.2**	24.5	28.8	26.0	27.3	48.1	28.6	20.7	23.6	18.7	13.6

* Les résultats issus de (Guo *et al.*, 2025) sont à prendre avec précaution car nous avons identifié des erreurs concernant certains d'entre eux dans l'article les présentant, publié à AAAI 2026, erreurs confirmées par les auteurs dans une communication privée.

TABLE 5 – Extraction d'arguments d'événements zero-shot (score F1). **AI** : Identification d'arguments (segment correct), **AC** : Classification d'argument (segment et catégorie corrects). **GuidelineEE**, **DecomposeEE** et **AEC** : résultats extraits de (Guo *et al.*, 2025) pour un modèle GPT3.5-turbo.

4.2 Extraction d'arguments d'événements zero-shot

Le modèle d'extraction d'arguments GLEE prend en charge l'inférence zero-shot après un pré-entraînement sur le corpus Omnivent. Le modèle identifie les arguments candidats à partir de tous les segments de texte possibles, permettant une couverture complète sans nécessiter d'annotations de segments de référence. Les performances du modèle sont présentées dans le tableau 5. À titre de comparaison, nous incluons les performances de GuidelineEE (Srivastava *et al.*, 2025), DecomposeEE (Shiri *et al.*, 2024) et AEC (Guo *et al.*, 2025). GLEE obtient de meilleures performances que tous ses concurrents aussi bien en identification d'argument qu'en classification pour ACE05 malgré le fait que ceux-ci utilisent GPT3.5-turbo. Il a cependant plus de mal avec CASIE, un jeu de données très spécifique au domaine de la cybersécurité. Il convient de noter que le modèle parvient à extraire des arguments dans quatre domaines distincts, démontrant ainsi une forte capacité de généralisation et de transfert entre domaines, sans ajustement spécifique à la tâche.

5 Conclusion et perspectives

Dans ce travail, nous proposons une méthode d'annotation automatique qui exploite les LLM pour annoter des données réelles, en encadrant leur production par des scores de confiance afin de n'en retenir que les résultats les plus fiables. Ainsi, en filtrant rigoureusement les annotations générées, nous avons pu constituer Omnivent, un corpus d'événements en anglais couvrant 38 859 types et 9 981 rôles d'arguments, sans recourir à l'expertise humaine. Pré-entraîné sur ce corpus, GLEE, notre transformeur bidirectionnel, tire parti de la sémantique des étiquettes pour aborder la détection d'événements et l'extraction d'arguments sans exemples supervisés. Les expériences menées sur neuf jeux de données issus de sept domaines distincts suggèrent que, malgré une taille environ 25 fois inférieure, GLEE parvient à devancer les meilleures approches à base de LLM, indiquant qu'une annotation automatique soigneusement contrôlée peut constituer une alternative crédible à l'annotation manuelle [2]. Ces résultats ouvrent plusieurs pistes de recherche. D'une part, notre cadre d'annotation ne se limite pas par principe aux événements et pourrait s'étendre à d'autres tâches d'extraction d'information. D'autre part, l'adoption d'architectures à contexte étendu et la constitution de versions multilingues d'Omnivent permettraient d'élargir le champ d'application de notre approche, sachant que l'extension de notre travail à une nouvelle langue ne nécessite qu'un corpus non annoté dans la langue visée et un LLM ayant des capacités de génération dans cette même langue.

2. Les éléments constitutifs de ce travail, corpus et code, sont librement accessibles à des fins de reproductibilité.

Remerciements

Ce travail est financièrement soutenu par France 2030, opéré par l'Agence Nationale de la Recherche (ANR), au travers du projet ANR SHARP ANR-23-PEIA-0008 et du programme Cluster IA ANR-23-IACL-0003 – DATAIA CLUSTER. Il a également été rendu possible par l'utilisation du cluster de calcul FactoryIA, financé par le conseil régional d'Île-de-France.

Références

BOROVIKOVA M., FERRÉ A., BOSSY R., ROCHE M. & NÉDELLEC C. (2024). Semantically-Informed Domain Adaptation for Named Entity Recognition. In A. APPICE, H. AZZAG, M.-S. HACID, A. HADJALI & Z. RAS, Éds., *Foundations of Intelligent Systems*, p. 55–64, Cham : Springer Nature Switzerland. DOI : 10.1007/978-3-031-62700-2_6.

CAI Z., KUNG P.-N., SUVARNA A., MA M., BANSAL H., CHANG B., BRANTINGHAM P. J., WANG W. & PENG N. (2024). Improving Event Definition Following For Zero-Shot Event Detection. In L.-W. KU, A. MARTINS & V. SRIKUMAR, Éds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 2842–2863, Bangkok, Thailand : Association for Computational Linguistics. DOI : 10.18653/v1/2024.acl-long.157.

CHEN R., QIN C., JIANG W. & CHOI D. (2024). Is a Large Language Model a Good Annotator for Event Extraction ? In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, p. 17772–17780.

DENG S., ZHANG N., KANG J., ZHANG Y., ZHANG W. & CHEN H. (2020). Meta-Learning with Dynamic-Memory-Based Prototypical Network for Few-Shot Event Detection. In *The Thirteenth ACM International Conference on Web Search and Data Mining (WSDM)*.

DODDINGTON G. R., MITCHELL A., PRZYBOCKI M. A., RAMSHAW L. A., STRASSEL S. M. & WEISCHEDEL R. M. (2004). The Automatic Content Extraction (ACE) Program - Tasks, Data, and Evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC)*.

DUBEY A., JAUHRI A., PANDEY A., KADIAN A., AL-DAHLE A., LETMAN A., MATHUR A., SCHELTEN A., YANG A., FAN A. *et al.* (2024). The Llama 3 Herd of Models. *CoRR*.

GAO J., ZHAO H., YU C. & XU R. (2023). Exploring the Feasibility of ChatGPT for Event Extraction. *arXiv preprint arXiv:2303.03836*.

GUO Q., WANG S., ZHANG J., ZHANG B., KANG Z., TIAN L. & YAN K. (2025). Extracting Events Like Code : A Multi-Agent Programming Framework for Zero-Shot Event Extraction. *arXiv preprint arXiv:2511.13118*.

HE P., GAO J. & CHEN W. (2023). DeBERTav3 : Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

HUANG K.-H., HSU I.-H., PAREKH T., XIE Z., ZHANG Z., NATARAJAN P., CHANG K.-W., PENG N. & JI H. (2024). TextEE : Benchmark, Reevaluation, Reflections, and Future Challenges in Event Extraction. In *ACL (Findings)*.

KIM J.-D., WANG Y., TAKAGI T. & YONEZAWA A. (2011). Overview of Genia Event Task in BioNLP Shared Task 2011. In J. TSUJII, J.-D. KIM & S. PYYSALO, Éds., *Proceedings of BioNLP*

Shared Task 2011 Workshop, p. 7–15, Portland, Oregon, USA : Association for Computational Linguistics.

LI M., ZAREIAN A., ZENG Q., WHITEHEAD S., LU D., JI H. & CHANG S.-F. (2020). Cross-media Structured Common Space for Multimedia Event Extraction. In D. JURAFSKY, J. CHAI, N. SCHLUTER & J. TETREAULT, Éds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 2557–2568, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-main.230.

LI S., ZHAN Q., CONGER K., PALMER M., JI H. & HAN J. (2023). GLEN : General-Purpose Event Detection for Thousands of Types. In H. BOUAMOR, J. PINO & K. BALI, Éds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, p. 2823–2838, Singapore : Association for Computational Linguistics. DOI : 10.18653/v1/2023.emnlp-main.170.

LIN S., HILTON J. & EVANS O. (2022). Teaching Models to Express Their Uncertainty in Words. *Transactions on Machine Learning Research*.

LIN Y., JI H., HUANG F. & WU L. (2020). A Joint Neural Model for Information Extraction with Global Features. In D. JURAFSKY, J. CHAI, N. SCHLUTER & J. TETREAULT, Éds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 7999–8009, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.acl-main.713.

LYU Q., ZHANG H., SULEM E. & ROTH D. (2021). Zero-shot Event Extraction via Transfer Learning : Challenges and Insights. In C. ZONG, F. XIA, W. LI & R. NAVIGLI, Éds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2 : Short Papers)*, p. 322–332, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.acl-short.42.

MA J., BALLESTEROS M., DOSS S., ANUBHAI R., MALLYA S., AL-ONAIZAN Y. & ROTH D. (2022). Label Semantics for Few Shot Named Entity Recognition. In S. MURESAN, P. NAKOV & A. VILLAVICENCIO, Éds., *Findings of the Association for Computational Linguistics : ACL 2022*, p. 1956–1971, Dublin, Ireland : Association for Computational Linguistics. DOI : 10.18653/v1/2022.findings-acl.155.

MA M. D., WANG X., KUNG P.-N., BRANTINGHAM P. J., PENG N. & WANG W. (2024). STAR : Boosting Low-Resource Information Extraction by Structure-to-Text Data Generation with Large Language Models. *Proceedings of the AAAI Conference on Artificial Intelligence*, **38**(17), 18751–18759. DOI : 10.1609/aaai.v38i17.29839.

MUELLER A., KRONE J., ROMEO S., MANSOUR S., MANSIMOV E., ZHANG Y. & ROTH D. (2022). Label Semantic Aware Pre-training for Few-shot Text Classification. In S. MURESAN, P. NAKOV & A. VILLAVICENCIO, Éds., *60th Annual Meeting of the Association for Computational Linguistics (ACL 2022)*, p. 8318–8334, Dublin, Ireland : Association for Computational Linguistics. DOI : 10.18653/v1/2022.acl-long.570.

PAREKH T., MAC A., YU J., DONG Y., SHAHRIAR S., LIU B., YANG E., HUANG K.-H., WANG W., PENG N. *et al.* (2024). Event Detection from Social Media for Epidemic Prediction. In *NAACL-HLT*.

PAREKH T., MEHTA K., MEHRABI N., CHANG K.-W. & PENG N. (2025). DiCoRe : Enhancing zero-shot event detection via divergent-convergent LLM reasoning. In C. CHRISTODOULOPOULOS, T. CHAKRABORTY, C. ROSE & V. PENG, Éds., *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, p. 20560–20582, Suzhou, China : Association for Computational Linguistics. DOI : 10.18653/v1/2025.emnlp-main.1038.

POURAN BEN VEYSEH A., EBRAHIMI J., DERNONCOURT F. & NGUYEN T. (2022). MEE : A Novel Multilingual Event Extraction Dataset. In Y. GOLDBERG, Z. KOZAREVA & Y. ZHANG,

Éds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, p. 9603–9613, Abu Dhabi, United Arab Emirates : Association for Computational Linguistics. DOI : 10.18653/v1/2022.emnlp-main.652.

PYYSALO S., OHTA T., MIWA M., CHO H.-C., TSUJII J. & ANANIADOU S. (2012). Event extraction across multiple levels of biological organization. *Bioinformatics*, **28**(18), i575–i581. DOI : 10.1093/bioinformatics/bts407.

SATYAPANICH T., FERRARO F. & FININ T. (2020). CASIE : Extracting Cybersecurity Event Information from Text. *Proceedings of the AAAI Conference on Artificial Intelligence*, **34**(05), 8749–8757. DOI : 10.1609/aaai.v34i05.6401.

SHIRI F., MOGHIMIFAR F., HAFFARI R., LI Y.-F., NGUYEN V. & YOO J. (2024). Decompose, Enrich, and Extract ! Schema-aware Event Extraction using LLMs. In *27th International Conference on Information Fusion (FUSION)*, p. 1–8. DOI : 10.23919/FUSION59988.2024.10706385.

SRIVASTAVA S., PATI S. & YAO Z. (2025). Instruction-Tuning LLMs for Event Extraction with Annotation Guidelines. In W. CHE, J. NABENDE, E. SHUTOVA & M. T. PILEHVAR, Éds., *Findings of the Association for Computational Linguistics : ACL 2025*, p. 13055–13071, Vienna, Austria : Association for Computational Linguistics. DOI : 10.18653/v1/2025.findings-acl.677.

SUN Z., LI J., PERGOLA G., WALLACE B., JOHN B., GREENE N., KIM J. & HE Y. (2022). PHEE : A Dataset for Pharmacovigilance Event Extraction from Text. In Y. GOLDBERG, Z. KOZAREVA & Y. ZHANG, Éds., *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, p. 5571–5587, Abu Dhabi, United Arab Emirates : Association for Computational Linguistics. DOI : 10.18653/v1/2022.emnlp-main.376.

TONG M., XU B., WANG S., HAN M., CAO Y., ZHU J., CHEN S., HOU L. & LI J. (2022). DocEE : A Large-Scale and Fine-grained Benchmark for Document-level Event Extraction. In M. CARPUAT, M.-C. DE MARNEFFE & I. V. MEZA RUIZ, Éds., *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 3970–3982, Seattle, United States : Association for Computational Linguistics. DOI : 10.18653/v1/2022.naacl-main.291.

VEYSEH A. P. B., EBRAHIMI J., DERNONCOURT F. & NGUYEN T. (2022). MEE : A Novel Multilingual Event Extraction Dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

WANG X., WANG Z., HAN X., JIANG W., HAN R., LIU Z., LI J., LI P., LIN Y. & ZHOU J. (2020). MAVEN : A Massive General Domain Event Detection Dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

WEI X., CUI X., CHENG N., WANG X., ZHANG X., HUANG S., XIE P., XU J., CHEN Y., ZHANG M., JIANG Y. & HAN W. (2024). ChatIE : Zero-Shot Information Extraction via Chatting with ChatGPT. *arXiv preprint arXiv:2302.10205*.

XIONG M., HU Z., LU X., LI Y., FU J., HE J. & HOOI B. (2024). Can LLMs Express Their Uncertainty ? An Empirical Evaluation of Confidence Elicitation in LLMs. In *The Twelfth International Conference on Learning Representations*.

ZARATIANA U., TOMEH N., HOLAT P. & CHARNOIS T. (2024). GLiNER: Generalist Model for Named Entity Recognition using Bidirectional Transformer. In K. DUH, H. GOMEZ & S. BETHARD, Éds., *2024 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2024)*, p. 5364–5376, Mexico City, Mexico : Association for Computational Linguistics. DOI : 10.18653/v1/2024.naacl-long.300.

ZHOU W., ZHANG S., GU Y., CHEN M. & POON H. (2024). UniversalNER : Targeted Distillation from Large Language Models for Open Named Entity Recognition. In *The Twelfth International Conference on Learning Representations*.

A Annexes

A.1 Instructions utilisées pour l'annotation du corpus Omnivent

Les instructions utilisées pour la classification et l'annotation des phrases par un LLM (Llama-70B-Instruct) sont données à la figure 3.

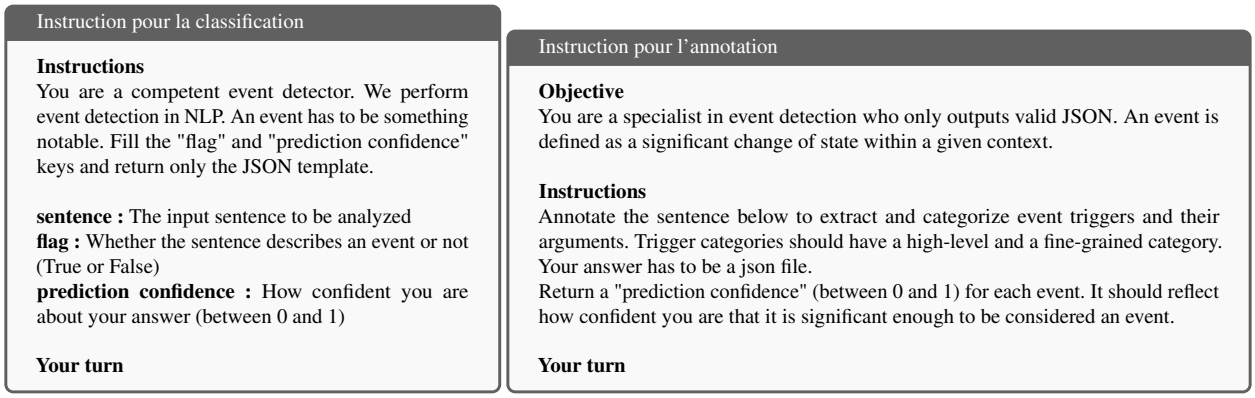

FIGURE 3 – Instructions pour la classification (gauche) et pour l'annotation (droite).

A.2 Impact de la taille du jeu de données de pré-entraînement

La capacité à annoter des corpus non structurés ouvre la voie à la construction de corpus de plus en plus vastes. Pour comprendre comment la taille du jeu de données de pré-entraînement influence les performances, nous avons mené une analyse détaillée en pré-entraînant GLEE sur des sous-ensembles d'Omnivent de tailles variées, puis en évaluant le modèle sur neuf benchmarks. La figure 4 présente la performance moyenne sur l'ensemble des tâches en fonction de la taille du corpus de pré-entraînement. Nous observons que les performances s'améliorent fortement avec de petites quantités de données, notamment en dessous de 10 000 exemples, puis se stabilisent, indiquant des gains marginaux à grande échelle. On peut noter que l'écart entre les scores d'identification (TI et AI) et les scores de classification (TC et AC) est plus marqué dans les contextes à faibles ressources. Cela suggère que, bien que le modèle puisse reconnaître la présence de déclencheurs et d'arguments, il a du mal à leur attribuer correctement des étiquettes lorsqu'il est entraîné sur un nombre limité de données. Ces résultats soulignent l'importance de la quantité de données aux premières étapes de l'entraînement, tout en suggérant qu'au-delà d'un certain seuil, une augmentation supplémentaire a un effet limité.

A.3 Analyse de la distribution des types d'événements dans Omnivent

Les figures 5 et 6 présentent respectivement la distribution du nombre d'occurrences par type d'événement et la distribution du nombre d'événements par nombre d'occurrences dans Omnivent.

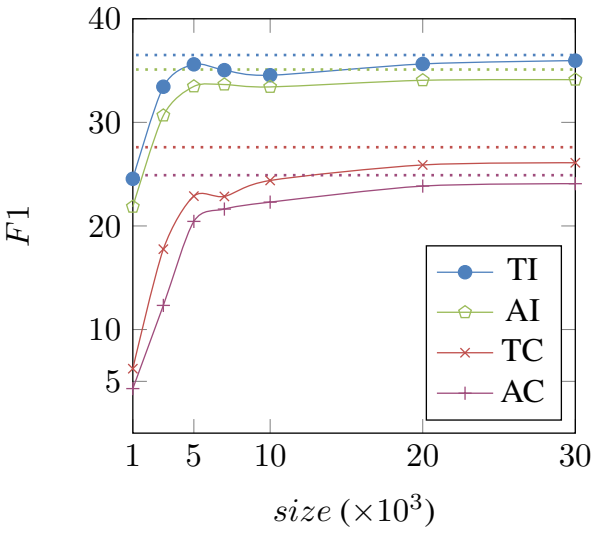

FIGURE 4 – Impact de la taille du jeu de données sur les performances zero-shot pour 9 benchmarks.

La distribution obtenue reflète la distribution naturelle des événements avec relativement peu de types d'événements très fréquents, tels que Communication:Statement ou encore Conflict:Attack, et beaucoup d'autres plus anecdotiques qui constituent la queue de la distribution. De plus, contrairement à ACE05, où le type d'événements le plus populaire, Conflict:Attack, représente 28,8 % des cas, dans Omnivent, le type le plus fréquent, Communication:Statement ne représente que 2,4 % des cas. De même, les 10 types d'événements les plus fréquents représentent 79,4 % des données dans ACE05 contre 5,9 % des données dans Omnivent. Cette propriété d'Omnivent vient de sa fonction, à savoir le pré-entraînement de modèle, tâche qui nécessite une diversité des données utilisées. Cette propriété est renforcée par notre choix de favoriser la diversité linguistique au sein du jeu de données en n'agrégeant pas les mots présentant un sens proche.

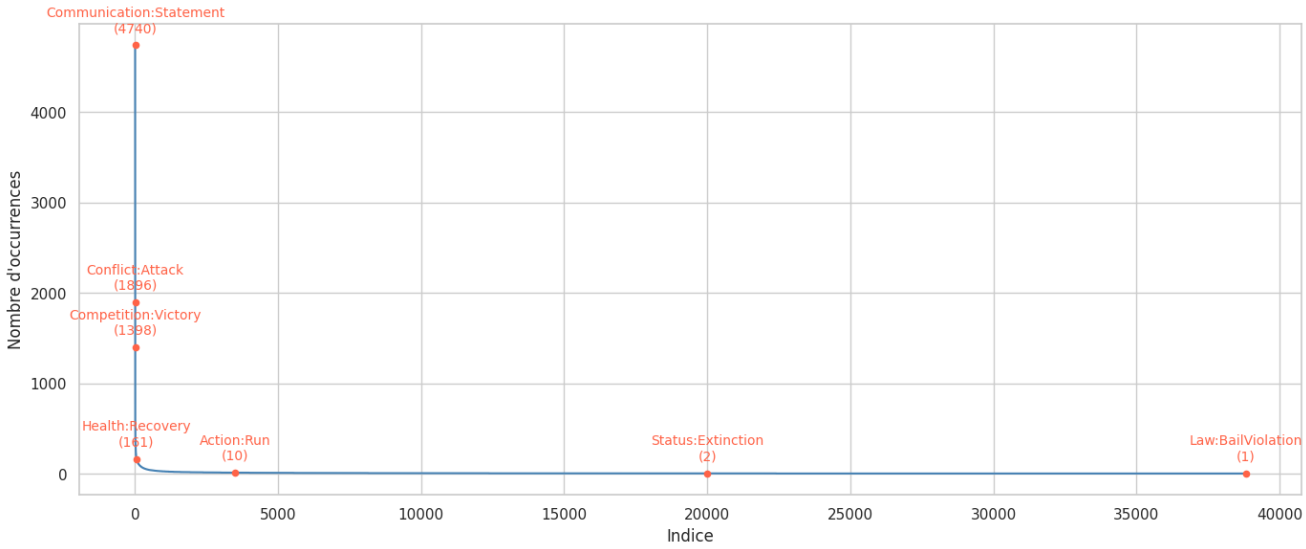

FIGURE 5 – Nombre d'occurrences pour chaque type d'événement. L'indice indique le classement du type d'événement dans le sens croissant.

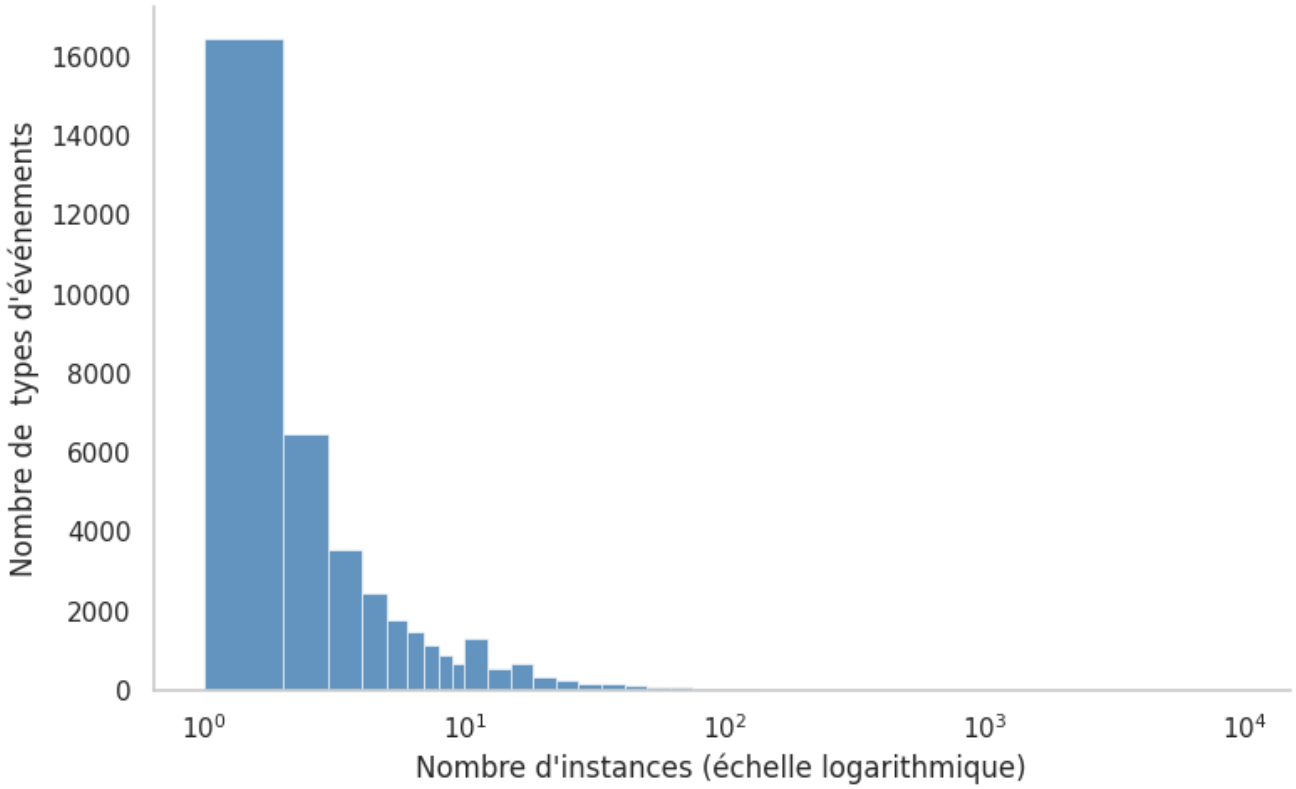

FIGURE 6 – Nombre de types d'événements pour chaque nombre d'occurrences.

A.4 Quelques exemples de phrases du corpus Omnivent et de leurs annotations

Dans cette section, nous proposons quelques exemples extraits du corpus Omnivent afin d'évaluer la qualité et la diversité des annotations aussi bien en termes de types d'événements que de rôles d'arguments. Les déclencheurs d'événements sont surlignés en rouge tandis que les arguments associés sont surlignés en bleu.

Example 1 :

Dylan Sprouse **dropped** 30 pounds and **learned** to speak Mandarin for his latest film

Type d'événement : Health:WeightLoss

— Déclencheur : dropped
— Personne : Dylan Sprouse
— Poids : 30 pounds

Type d'événement : Education:SkillAcquisition

— Déclencheur : learned
— Personne : Dylan Sprouse
— Langue : Mandarin

Example 2 :

Steve Von Deschwanden was **stopped** by police after he was allegedly seen speeding on John Street in the township of Madawaska Valley, west of Ottawa, on Dec

Type d'événement : Law:Arrest

— Déclencheur : stopped
— Suspect : Steve Von Deschwanden
— Autorité : police
— Localisation : John Street
— Localisation : Madawaska Valley, west of Ottawa
— Temps : Dec

Example 3 :

Humanitarian partners are also reporting that those arriving in Uganda continue to cite **violence** and **indiscriminate killings** of civilians, **sexual violence**, and **destruction** of livelihoods," the note said

Type d'événement : Conflict:Violence
— Déclencheur : violence
— Victimes : civilians

Type d'événement : Conflict:Killing
— Déclencheur : indiscriminate killings
— Victimes : civilians

Type d'événement : Conflict:Sexual Violence
— Déclencheur : sexual violence
— Victimes : civilians

Type d'événement : Conflict:Destruction
— Déclencheur : destruction
— Propriété : livelihoods

Example 4 :

PRHC's Pediatric Urgent Care Clinic (POP) is **offering** extended after-hours care throughout December

Type d'événement : Healthcare:Service
— Déclencheur : offering
— Unité de soin : PRHC's Pediatric Urgent Care Clinic (POP)
— Aménagement : extended after-hours care
— Temps : December

A.5 Interfaces d'évaluation de l'annotation du corpus Omnivent

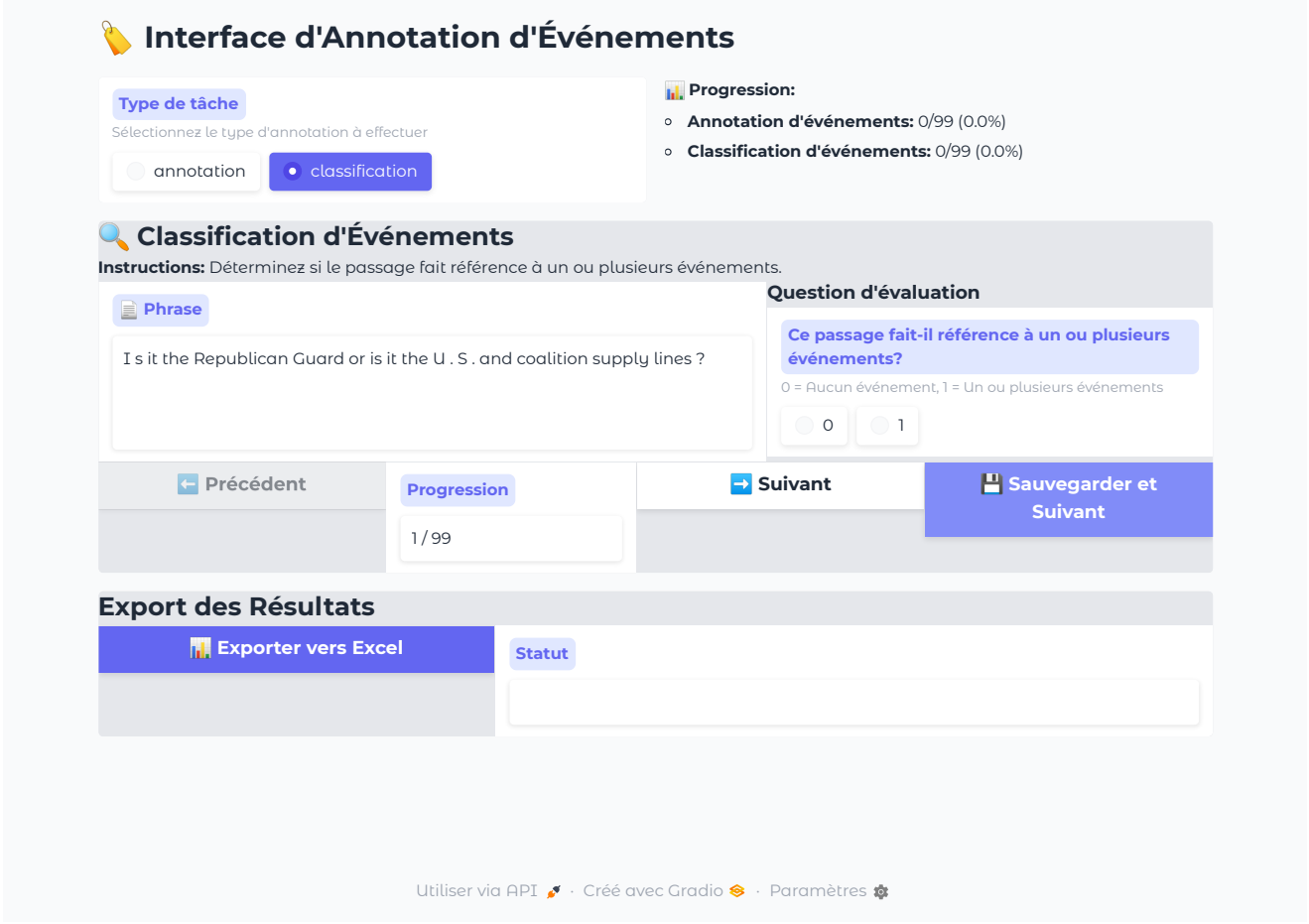

FIGURE 7 – Interface d'évaluation de la qualité des données pour l'expérience 1 : annotation de la présence d'événements.

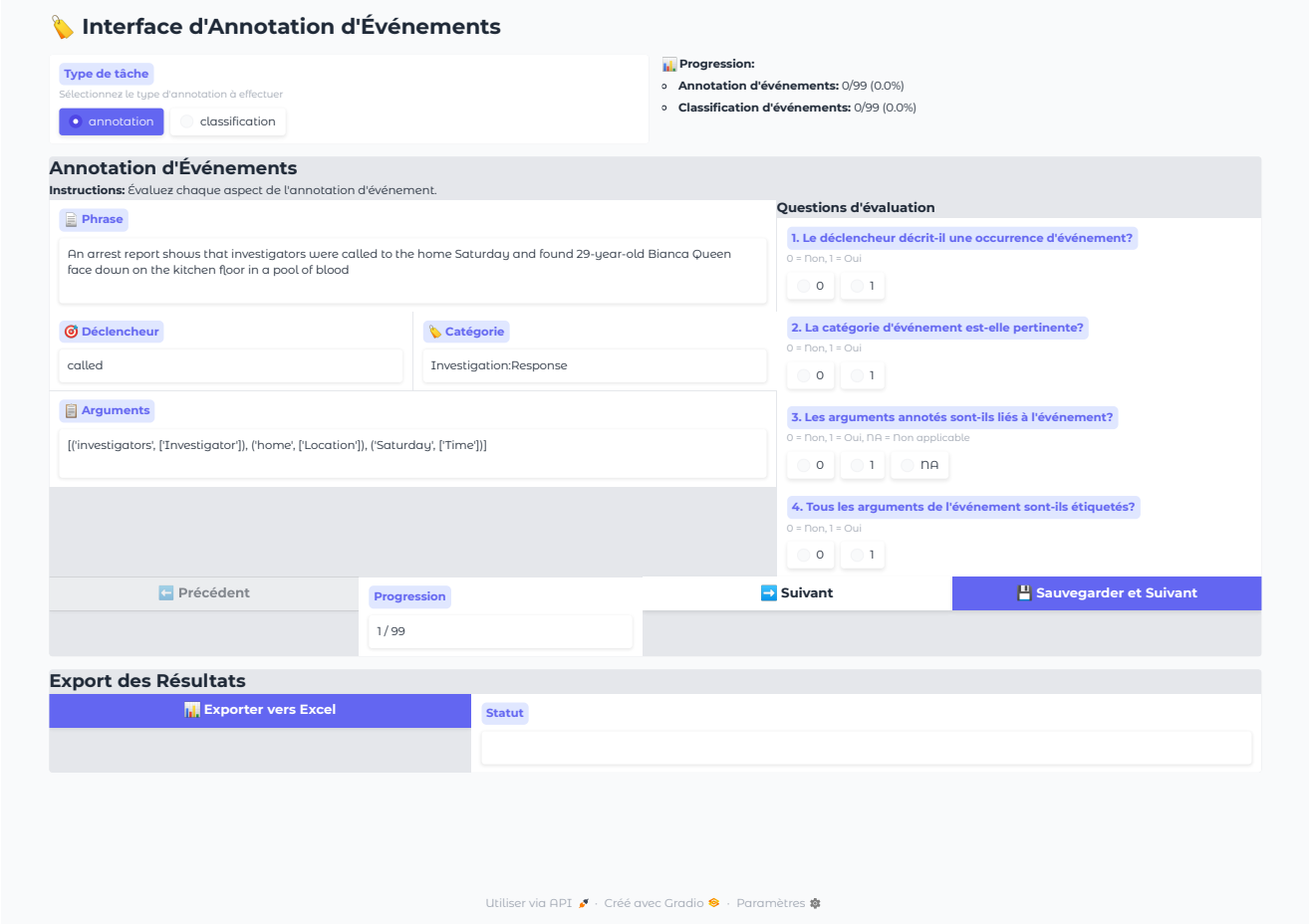

FIGURE 8 – Interface d'évaluation de la qualité des données pour l'expérience 2 : annotation détaillée d'un événement et de ses arguments.