# Annotation sémantique pour la géolocalisation d'entités spatiales dans des tweets

Gaëtan Caillaut, Cécile Gracianne, Samuel Auclair, Nathalie Abadie, Guillaume Touya

# Annotation sémantique pour la géolocalisation d'entités spatiales dans des tweets

G. Caillaut[1], C. Gracianne[1], S. Auclair[1], N. Abadie[2] et G. Touya[2]

[1] BRGM
[2] LASTIG, Univ Gustave Eiffel, IGN-ENSG

g.caillaut@brgm.fr, c.gracianne@brgm.fr, s.auclair@brgm.fr, nathalie-f.abadie@ign.fr, guillaume.touya@ign.fr

## Résumé

*Cet article présente une méthode d'annotation sémantique (Entity Linking) dédiée à la géolocalisation de messages postés sur les réseaux sociaux. Nous proposons une variante de l'architecture à double encodeur capable de détecter simultanément les mentions des entités présentes dans un texte et d'en calculer des représentations vectorielles ; là où les travaux récents menés dans ce domaine ne proposent que des systèmes capable de produire une représentation pour une unique entité annotée au préalable par un système tiers. Nous montrons également que, malgré la difficulté accrue de la tâche, ce système parvient à concurrencer, et même à surpasser, les systèmes à double encodeur.*

## Mots-clés

*Annotation sémantique, Reconnaissance d'entités nommées, Double Encodeur, Wikipédia, Géolocalisation*

## Abstract

*We present an Entity Linking method dedicated to the geolocation of social network posts. We propose a variant of the dual-encoder architecture which enables both the detection of entity mentions and the computation of vectorial representations for these entities ; while state-of-the-art approaches can only compute one representation at one time and rely on a second system to annotate entity mentions. We show that, despite the increased difficulty of the task, our system competes and even outperforms vanilla dual-encoders systems*

## Keywords

*Entity Linking, Mention Detection, Dual-Encoder, Wikipedia, Geotagging*

## 1 Introduction

Dans ce travail, nous nous intéressons à la géolocalisation automatique du contenu textuel de messages postés sur les réseaux sociaux lors de la survenue de catastrophes naturelles à cinétique rapide, telles que des séismes ou des inondations rapides. De nombreuses études ont démontré que les utilisateurs de réseaux sociaux ont tendance à relayer spontanément et massivement ce genre d'évènements [21, 7], faisant de leur contenu une source d'informations extrêmement précieuse pour localiser rapidement et précisément les zones sinistrées. Les messages émis dans ce contexte particulier contiennent bien souvent des références spatiales [10], telles que des noms de villes ou des cours d'eau, qu'il est important de pouvoir identifier pour permettre de localiser relativement précisément l'évènement. Dans ce contexte, l'information géographique s'avère déterminante, car elle permet à la fois de construire une connaissance situationnelle partagée entre les acteurs de la gestion de crise et d'identifier des problématiques particulières devant être prises en compte, telle que l'identification de personnes demandant à être secourues. Par conséquent, des informations de géolocalisation imprécises peuvent s'avérer inutiles, et des erreurs de localisation avoir des conséquences désastreuses. Enfin, un système de géolocalisation utilisé dans ce contexte doit être capable de fournir des résultats en temps réel afin de permettre une intervention rapide sur la zone touchée.

Afin de géolocaliser une information, il est tout d'abord nécessaire d'identifier au préalable les entités porteuses d'informations spatiales présentes dans le texte. C'est le rôle des méthodes de détection d'entité nommées, ou *Named Entity Recognition* (NER), qui sont entraînées pour détecter et classifier les différentes entités présentes dans un document. Les architectures `BERT` [4] actuelles se sont montrées particulièrement performantes pour la réalisation de cette tâche [25], et sont aujourd'hui largement utilisées. Dans un deuxième temps, les entités spatiales détectées doivent ensuite être positionnées géographiquement, ce qui s'avère être un problème plus complexe si on se fie à la précision des méthodes existantes [11, 29]. Différentes pistes de recherche sont proposées, comme l'inférence de la localisation de l'utilisateur à travers le contenu d'un message émis sur les réseaux sociaux ou de l'adresse IP d'un appareil utilisé pour consulter certaines pages Web spécifiques [5, 24], mais les solutions actuelles restent imprécises et leurs performances sont très dépendantes de la zone à localiser. HAN, COOK et BALDWIN [12] montrent, par exemple, que l'immense majorité des utilisateurs de Twitter habitent à proximité des villes, ce qui rend discutables les perfor-

mances de tels système en milieu rural.

L'émergence des bases de connaissances, telle que Wikidata [1] a permis de proposer des stratégies alternatives pouvant permettre d'améliorer les résultats des méthodes de géolocalisation. Ces bases de connaissances proposent, en effet, de structurer l'ensemble des connaissances humaines, qui comprennent également les entités géographiques. Elles présentent également l'avantage d'être facilement éditables et en constante évolution. Assez naturellement, les chercheurs se sont intéressés au problème d'annotation sémantique, ou *Entity Linking* (EL), qui consiste à lier les entités présentes dans un texte avec celles des bases de connaissances. Il est alors possible d'enrichir le texte étudié par des connaissances externes, issues de ces bases de connaissances. Les travaux à ce sujet s'appuient de façon prépondérante sur Wikipédia et Wikidata, ce qui s'explique par la quantité massive de données **annotées** qu'offre Wikipédia. En effet, les entités présentes dans les pages Wikipédia sont généralement associées à des liens hypertextes qui redirigent le lecteur vers une page les décrivant. Ces liens sont, en principe, positionnés sur les mentions représentant une entité cible, ce qui permet de les considérer comme des annotations en soi. Il devient alors possible de répondre au problème d'EL en (1) entraînant un système à détecter les entités présentes dans une page Wikipédia (tâche *Mention Detection* ou MD), puis (2) en prédisant la page/entité cible. Bien que les approches NER puissent être utilisées pour répondre à la première étape, il est généralement superflu d'attribuer une étiquette aux entités détectés (puisque celle-ci peut être déduite de l'entité prédite), c'est pourquoi on parle de détection, et non pas de reconnaissance, d'entités.

L'objectif de ce travail de recherche est d'exploiter ces avancées récentes du Traitement Automatique de la Langue (TAL) en exploitant une approche d'EL pour résoudre le problème de géolocalisation d'évènements. À cet effet, nous nous appuyons sur les travaux de BOTHA, SHAN et GILLICK [2] et proposons deux modifications visant à alléger certaine contraintes afin d'aboutir à un système autonome capable de, **conjointement** :

1. détecter les $n$ mentions d'entités dans un document, au lieu de se reposer sur un système tiers et potentiellement coûteux en ressources ;
2. lier **toutes** ces entités à Wikidata, au lieu d'effectuer $n$ fois ce processus.

Ce nouveau système est alors à la fois plus léger, n'ayant plus recours à un modèle tiers de détection de mentions, et plus efficace, puisque toutes les $n$ mentions sont liées en une seule étapes, contre $n$ auparavant.

# 2 Travaux connexes

Depuis les avancées récentes en TAL, les méthodes de détection d'entités nommées reposent principalement sur des approches d'apprentissage profond [14]. Les architectures neuronales modernes produisent des représentations contextuelles pour les mots, et plus précisément pour les *tokens* [2], reçus en entrée. Ces représentations capturent des informations syntactiques et sémantiques qui peuvent être exploitées pour classer et identifier les tokens représentant des entités spatiales. Aujourd'hui, ces approches s'appuient généralement sur l'architecture `BERT` [4], elle-même reposant sur l'architecture Transformer [26].

Ces modèles sont tout d'abord entraînés pour apprendre à extraire des informations (syntactiques et sémantiques) génériques à partir de corpus massifs et sont redistribués par la suite. Les chercheurs sont ainsi en mesure de s'appuyer directement sur ces modèles pré-entrainés sans devoir répéter la coûteuse phase de pré-entraînement. Par exemple, `CamemBERT` [16] et `Cedille` [20] sont des modèles pré-entraînés sur la langue française, qui contiennent une grande quantité de connaissances spécifiques à cette langue. Ces modèles pré-entrainés peuvent ensuite être spécialisés sur des tâches spécifiques, comme de la traduction automatique, de l'analyse de sentiment etc. En particulier, ces modèles peuvent être entraînés à associer les entités spatiales à des coordonnées géographiques, ou encore aux identifiants de gazetiers tels que GeoNames [3] ou la BD-TOPO ® [4]. D'autre part, ces approches, dites contextuelles, sont capables de tenir compte du contexte dans lequel est utilisé un token, ce qui leur confère un potentiel élevé pour désambiguïser des entités. En effet, dans le cas des entités spatiales, un même toponyme peut faire référence à plusieurs lieux. Par exemple, il y a cinq villes en France qui s'appellent « Chaumont », et il est nécessaire de s'appuyer sur le contexte dans lequel le toponyme apparaît pour en déduire la localité effectivement recherchée.

De nombreux travaux proposent de poser le problème de géolocalisation comme un problème de classification [18]. Pour ce faire, la surface de la Terre (ou de la zone d'intérêt) est discrétisée en $n$ cellules. L'objectif du système de géolocalisation est alors d'affecter les entités spatiales aux bonnes classes/cellules. Toutefois, ces approches ne permettent pas de prédire une localisation exacte, mais une région dans laquelle l'entité spatiale est censée se situer. PAULE, SUN et MOSHFEGHI [23] montrent que ces approchent peuvent se révéler relativement précises dans un cadre restreint. Ils proposent un système capable de localiser un incident en lien avec le trafic routier et parviennent à prédire une localisation avec une marge d'erreur, en moyenne, inférieure à 10 km. Toutefois, leurs modèles ne sont entraînés que sur une région de taille limitée (Chicago ou New York) et se concentrent principalement sur les grands axes routiers, réduisant davantage la zone à couvrir. Sans ces restrictions, les marges d'erreurs sont de l'ordre de plusieurs centaines, voire milliers, de kilomètres [11, 29].

Les approches reposant sur des gazetiers proposent d'associer chaque entité spatiale à un lieu enregistré dans l'annuaire. Autrement dit, cela revient à répondre au problème EL en alignant les entités spatiales présentes dans un do-

1. `https://www.wikidata.org`

2. Les approches modernes découpent les mots en sous-mot, un mot peut alors être constitué de plusieurs tokens.

3. `http://www.geonames.org/`

4. `https://geoservices.ign.fr/documentation/donnees/vecteur/bdtopo`

cument à celles enregistrées dans le gazetier. Ces derniers contiennent non seulement des informations spatiales précises concernant la forme (point, surface, ligne) et la localisation des entités spatiale, mais elles contiennent également différentes propriétés (population, altitude, zones inondables, ...) dont certaines peuvent s'avérer particulièrement pertinentes selon le contexte. Par exemple, connaître le niveau de dénivelé d'un terrain peut permettre de mieux évaluer les conséquences d'une inondation. Ces atouts ont naturellement poussés certains chercheurs à proposer des solutions basées sur l'annotation sémantique pour répondre au problème de géolocalisation de messages postés sur internet [13, 6, 1].

Traditionnellement, ces approches reposaient sur des comparaisons de chaînes de caractères, ce qui, d'une part, ne permet pas de désambiguïser les homonymes tels que « Chaumont », mais rend également difficile la prise en compte des éventuelles fautes d'orthographe ou de frappe, qui sont particulièrement présentes sur les réseaux sociaux. Les approches récentes tentent donc de s'extraire de ces contraintes en s'appuyant sur des méthodes d'apprentissage profond [28, 9, 2].

# 3 Architecture proposée

## 3.1 Système à double encodeur

GILLICK et al. [9] proposent une architecture à double encodeur pour répondre au problème EL. Cette architecture comprend un premier encodeur, le *Mention Encoder*, dont le rôle est de produire une représentation pour une mention d'une entité, étant donnée le texte de la mention et son contexte. Le second encodeur, l'*Entity Encoder*, est quant à lui chargé de produire une représentation pour l'entité cible à partir d'une description de celle-ci (on utilise généralement le premier paragraphe de la page Wikipédia correspondante). Les deux encodeurs sont entraînés de manière à produire des représentations similaires. Une fois le système entraîné, les représentations des entités candidates peuvent être pré-calculées par l'*Entity Encoder*, qui peut alors être retiré du système. Les mentions d'entités peuvent alors être liées aux entités en comparant les sorties du *Mention Encoder* aux représentation ainsi pré-calculées.

BOTHA, SHAN et GILLICK [2] s'appuient sur ces travaux et remplacent les deux encodeurs par des modèles BERT pré-entrainés. Les résultats publiés par les auteurs semblent indiquer que cette architecture est particulièrement performante pour répondre au problème d'EL. Nous estimons cependant qu'elle peut être améliorée sur plusieurs aspects, en éliminant certaines contraintes imposées par les auteurs. En effet, le système proposé permet de calculer la représentation d'une **unique** entité présente dans un texte **préalablement annoté**. D'une part, une phrase contient bien souvent plusieurs entités ; d'autre part, il est nécessaire d'appliquer un système de détection d'entités au préalable, afin d'annoter le texte en conséquence. Les performances en EL de cette approche se trouvent alors limitées par les performances de la méthode de détection d'entités employée. Par ailleurs, selon la quantité d'information à traiter et la complexité du système, ajouter une étape dans le processus de géolocalisation peut significativement altérer le temps de réponse du système. Or, ce genre de risque est inacceptable dans un contexte où le traitement des informations en temps réel est crucial. De plus, si le texte en entrée contient plusieurs entités, le processus EL devra être appliqué autant de fois que nécessaire puisque la méthode proposée par les auteurs ne permet d'aligner les entités qu'une par une. Si un document contient $n$ entités, alors il est nécessaire de le dupliquer $n$ fois et d'annoter chaque entité individuellement, ce qui risque encore une fois d'impacter négativement le temps de réponse du système.

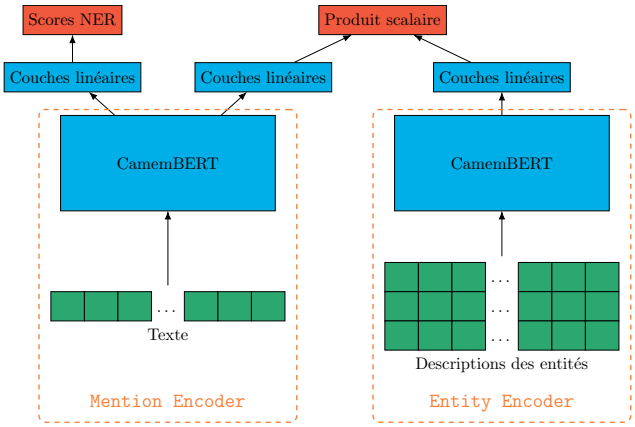


FIGURE 1 – Architecture MD + EL à double encodeur. L'*Entity Encoder* reçoit une description par entité présente dans le texte en entrée du *Mention Encoder*. Ici, on suppose que le texte contient 3 entités. Les sorties des encodeurs sont fournies telles quelles aux couches linéaires spécialisées (calculs des scores NER et similarités).

C'est pourquoi nous proposons de nous appuyer sur les travaux de BOTHA, SHAN et GILLICK [2] pour proposer une architecture capable de détecter conjointement **toutes** les entités présentes dans le texte et d'en calculer leurs représentations en une unique étape.

## 3.2 Système MD + EL à double encodeur

Comme évoqué précédemment, nous trouvons regrettable d'être contraint d'enchaîner plusieurs systèmes (MD puis EL), là où un unique système pourrait se révéler suffisant. Simplifier la chaîne de traitement est susceptible d'être bénéfique sur plusieurs aspects, parmi lesquels, la réduction du temps de traitement d'une même information, l'allègement des besoins en espace de stockage et en puissance de calcul, ainsi que la minimisation des biais et des erreurs introduits à chaque étape du processus. MARTINS, MARINHO et MARTINS [17] ont ainsi montré qu'entraîner conjointement un système sur ces deux tâches pouvait aboutir à une amélioration des performances globales du modèle. Nous proposons donc de modifier l'architecture utilisée par BOTHA, SHAN et GILLICK [2] sur deux points : (1) nous intégrons la détection des entités au sein du même système, et (2) nous proposons de calculer des représentations pour toutes les entités présentes, en une seule étape.

**Détection des entités** Nous avons ajouté un module de détection des entités à partir des représentations en sortie du *Mention Encoder*. Ce module s'apparente à un système classique de détection d'entités nommées. Il est constitué de deux couches linéaires, la première réduit la taille des représentations en sortie du *Mention Encoder*, la seconde attribue un score à chaque classe possible. Dans le cadre de cette étude, nous ne considérons qu'une répartition des entités en trois classes : `B` (début d'une entité), `I` (continuation d'une entité) et `O` (absence d'entité). Il est bien entendu envisageable d'étendre le nombre de classes afin de se rapprocher d'un système NER, mais nous ne nous intéressons, dans le cadre de ces travaux, qu'à la détection des entités (tâche MD) et non pas à leur classification (tâche NER).

**EL sur toutes les entités** Dans l'architecture à double encodeur originale, le *Mention Encoder* utilise le token spécial `[CLS]` en guise de représentation de l'entité annotée dans le document en entrée. De ce fait, le système est incapable de calculer des représentations pour plusieurs entités à la fois. Nous proposons d'ajouter un module recevant les représentations en sortie du *Mention Encoder* et calculant de nouvelle représentations pour chaque token. Ce système sera alors entraîné de manière à produire des représentations similaires à celles de l'*Entity Encoder*. Bien que l'architecture de ce dernier reste inchangée, il reçoit dorénavant autant de descriptions que d'entités présentes dans le document en entrée du *Mention Encoder*.
Ce système pourrait être allégé en ne calculant des représentations que pour les tokens détectés comme entité par le module de détection d'entités décrit dans le paragraphe précédent. Ces considérations ne sont pas prises en compte à cette étape du projet, mais constituent une de ses perspectives de développement futur.

# 4 Résultats expérimentaux

## 4.1 Jeux de données

CAILLAUT et al. [3] proposent un jeu de données extrait du Wikipédia français et dédié à l'entraînement de systèmes EL. Celui-ci ne couvre pas l'intégralité du corpus Wikipédia, mais, à l'instar de MERITY et al. [19], seulement les *bons articles*[5] et *articles de qualité*[6]. Ce jeu de données est constitué de **6023 documents** et **304 826 entités** distinctes apparaissant dans **1 619 961 mentions**. Dans la suite, `frwiki EL` sera utilisé pour faire référence à ce jeu de données
Notre objectif final étant de détecter et géolocaliser les entités spatiales mentionnées dans des tweets rédigés en français, s'appuyer uniquement sur un corpus Wikipédia risque d'affecter les performances du système final, puisque les styles d'écritures employés sur Twitter sont très différents de ceux employés dans Wikipédia. Pour atténuer ce problème, nous avons utilisé le jeu de données construit à l'occasion du challenge de détection d'entités nommées dans des tweets, organisé lors de la conférence CAp 2017 [15], que nous avons augmenté avec le jeu de données Wikiner [22]. Plus de détails sur ces jeux de données sont présentés dans les Tableaux 1 et 2. Ces deux jeux de données n'étant annotés qu'en entités nommées, il ne peuvent être utilisés que pour entraîner un système à détecter les mentions d'entités. Il n'existe pas, à notre connaissance, de corpus français issu de réseaux sociaux dédié à la tâche EL.
Ainsi, le jeu de données `CAp + Wikiner` nous permet d'estimer les capacités de chaque approche à détecter les mentions présentes dans des tweets, ce qui est particulièrement important puisque notre objectif est d'exploiter les informations contenues dans des messages provenant de réseaux sociaux. Le jeu de données `frwiki EL` nous permet, quant à lui, d'évaluer les performances EL des différents modèles étudiés.
La seconde portion de notre système sera donc nécessairement entraînée exclusivement à partir de données provenant de Wikipédia.

| Jeu de données | Documents | Mentions |
|---|---|---|
| CAp 2017 | 6688 | 6660 |
| Wikiner FR | 266 348 | 505 856 |
| CAp + Wikiner | 273 033 | 512 418 |
| frwiki EL | 6023 | 1 619 961 |

TABLE 1 – Jeux de données pour la détection d'entités nommées. La colonne *Documents* indique le nombre de tweets (`CAp 2017`), de phrases (`Wikiner`) ou d'articles (`frwiki EL`) présents dans le jeu de données.

| Jeu de données | B | I | O |
|---|---|---|---|
| `CAp 2017` | 6649 | 2768 | 129 339 |
| `Wikiner FR` | 506 448 | 324 824 | 6 174 131 |
| `CAp + Wikiner` | 513 097 | 327 592 | 6 303 470 |
| `frwiki EL` | 1 619 961 | 1 293 998 | 34 137 467 |

TABLE 2 – Distribution des classes.

## 4.2 Systèmes évalués et entraînement

Nous avons réalisé trois expériences. La première vise à évaluer la capacité des architecture BERT à détecter les mentions d'entités dans un texte (tâche *Mention Detection*, ou MD). L'objectif de la seconde est d'évaluer les performances sur la tâche EL du système à double encodeur sur le jeu de données, en français, proposé par CAILLAUT et al. [3]. Enfin, la dernière expérience cherche à évaluer les performances du système à double encodeur modifié, que nous proposons, sur la double tâche MD + EL. Tous nos modèles ont été initialisés avec les poids de `CamemBERT` [16].
Nous avons entraîné quatre systèmes :

**Système $\text{MD}_{\text{tweets}}$** `CamemBERT` spécialisé sur le jeu de données `CAp + Wikiner` pour la détection d'entités.

5. https://fr.wikipedia.org/wiki/Cat%C3%A9gorie:Bon_article
6. https://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Articles_de_qualit%C3%A9/Justification_de_leur_promotion

**Système $\text{MD}_{\text{frwiki}}$** `CamemBERT` spécialisé sur le jeu de données `frwiki EL` pour la détection d'entités.

**Système EL** Double encodeur, sans modification, entraîné sur `frwiki EL` pour l'*Entity Linking*.

**Système MD + EL** Double encodeur, avec nos modifications, entraîné sur `frwiki EL` pour la détection d'entités et l'*Entity Linking*.

Dans le cas des systèmes à double encodeur (systèmes `EL` et `MD + EL`), les expérimentations ont été effectuées en initialisant les encodeurs avec les 4 premières couches de `CamemBERT`, selon les recommandations de BOTHA, SHAN et GILLICK [2], ainsi qu'avec les 12 couches du modèle. Seules les performances des meilleurs modèles sont reportées.

Les deux jeux de données ont été partitionnés en ensembles d'entraînement, validation et test :

— 80 % entraînement, 10 % validation et 10 % test pour `CAp + Wikiner`

— 90 % entraînement, 5 % validation et 5 % test pour `frwiki EL`

Le système `EL` est entraîné en optimisant le critère ci-dessous, où $\mathcal{L}_{EL}$ est l'*in-batch cross-entropy* [8] calculée sur les représentations des entités en sortie des deux encodeurs et $\mathcal{L}_{HN}$ la *binary cross-entropy* calculée sur les *hard-negatives* [9] minés après chaque *epoch*.

$$\mathcal{L}_{EL} + \mathcal{L}_{HN}$$

Le système `MD + EL` est entraîné en optimisant les mêmes critères, auxquels s'ajoute la *cross-entropy* $\mathcal{L}_{MD}$ calculée sur les sorties du module de détection de mentions d'entités.

$$\mathcal{L}_{EL} + \mathcal{L}_{HN} + \mathcal{L}_{MD}$$

Les systèmes `MD` sont, quant à eux, entraînés en optimisant uniquement le critère $\mathcal{L}_{MD}$.

Le pas d'apprentissage a été fixé à $3{\times}10^{-5}$ et décroît linéairement au fil de l'entraînement. Le système `EL` à été optimisé à l'aide de l'algorithme AdamW paramétré avec les valeurs par défaut de la bibliothèque PyTorch.

## 4.3 Résultats

**Détection des entités** La tâche MD se distingue de la tâche NER par son objectif qui est de détecter les mentions d'entités sans chercher à les catégoriser. Ainsi, dans un problème NER, les entités représentant des personnes doivent être distinguées des entités représentant des organisations ou des dates. La tâche MD, quant à elle, consiste à simplement détecter les zones de texte mentionnant une entité, sans chercher à lui assigner une étiquette.

Nous avons encodé les jeux de données en entrée au format `BIO` et nous avons entraîné nos systèmes à prédire la bonne étiquette pour chaque token. Les performances de nos différents systèmes sont données en Tableau 4. Les systèmes sont évalués sur l'ensemble de test du jeu de données `CAp + Wikiner` puisque ce jeu de données inclut des données provenant de Twitter. Nous supposons alors que ce jeu de données est susceptible de se rapprocher des données réelles auxquelles sera confronté le système.

Sans surprise, le modèle $\text{MD}_{\text{tweets}}$, entraîné sur le jeu `CAp + Wikiner`, écrase très largement les deux autres modèles, qui n'ont pas été entraînés sur Twitter. Il est en revanche plus surprenant de constater que le système `MD + EL` surpasse légèrement le système $\text{MD}_{\text{frwiki}}$. Ce résultat tend à valider les conclusions de MARTINS, MARINHO et MARTINS [17] et suggère que les systèmes de reconnaissance d'entités nommées tirent profit à être entraînés conjointement sur la tâche EL.

Étant donnée la forte prépondérance de l'étiquette `O`, à savoir les tokens qui ne sont associés à aucune entité, nous craignons que les modèles soient incités naturellement à prédire l'absence d'entités. Comme le montrent les matrices de confusion en Tableau 3, les modèles entraînés sur `frwiki EL` ont effectivement tendance à favoriser l'étiquette `O`.

| | $\text{MD}_{\text{tweets}}$ | | | $\text{MD}_{\text{frwiki}}$ | | | MD + EL | | |
|---|---|---|---|---|---|---|---|---|---|
| | B | I | O | B | I | O | B | I | O |
| B | 50 091 | 912 | 574 | 33 411 | 5298 | 12 868 | 34 235 | 4119 | 13 223 |
| I | 493 | 98 664 | 714 | 2038 | 76 835 | 20 998 | 2083 | 75 160 | 22 628 |
| O | 953 | 1271 | 720 252 | 64 717 | 96 482 | 561 267 | 64 272 | 74 153 | 584 041 |

TABLE 3 – Matrices de confusion des différents systèmes sur le jeu de test `CAp + Wikiner`.

| Classes | $\text{MD}_{\text{tweets}}$ | $\text{MD}_{\text{frwiki}}$ | MD + EL |
|---|---|---|---|
| B | 0,97 | 0,65 | 0,66 |
| I | 0,99 | 0,77 | 0,75 |
| O | 1,00 | 0,78 | 0,81 |
| Global | 0,99 | 0,77 | 0,79 |

TABLE 4 – Taux de bonnes classifications par classe des différents systèmes sur le jeu de test `CAp + Wikiner`. La ligne « Global » correspond au micro-FScore toutes classes confondues.

Enfin, les deux systèmes souffrent cruellement de ne pas avoir été entraînés sur des documents provenant de Twitter. En effet, les deux modèles $\text{MD}_{\text{tweets}}$ et $\text{MD}_{\text{frwiki}}$ partagent rigoureusement la même architecture, seules les données d'entraînement diffèrent. Cela nous incite à penser que les performances des deux systèmes pourraient être largement améliorées en intégrant le corpus `CAp + Wikiner` lors de l'apprentissage, ce qui sera fait à l'avenir. En particulier, et si l'on considère qu'une mention est détectée si et seulement si l'étiquette `B` est prédite, les deux système entraînés sur `frwiki EL` ne détectent que les deux tiers des mentions, là où $\text{MD}_{\text{tweets}}$ détecte presque l'intégralité des mentions annotées, comme indiqué en Tableau 4.

**Entity Linking** Nous avons évalué les systèmes `EL` et `MD + EL` sur le jeu de test du corpus `frwiki EL`[3]. Les performances de ces deux modèles, données en Rappel@$k$, sont présentées en Tableau 5. Le Rappel@$k$, ou R@$k$, indique la proportion de prédictions pertinentes présentes

dans les $k$ premiers résultats [7]. Dans le cadre de la tâche EL, il n'existe qu'une entité pertinente pour chaque mention dans le texte. Une valeur R@100 de 0,42 indique que, dans 42 % des cas, l'entité à prédire apparaît dans les 100 premiers candidats.

Pour chaque mention $m$, l'ensemble des entités candidates correspond aux $k$ entités dont les représentations (calculées par l'*Entity Encoder*) sont les plus proches [8] de la représentation de $m$ produite par le *Mention Encoder*. Les représentations pour les entités sont calculées au préalable, en fournissant leurs descriptions textuelles à l'*Entity Encoder*, une fois celui-ci entraîné. L'*Entity Encoder* n'est alors pas utilisé pour lier une mention à son entité. Il reste toutefois indispensable pour calculer les représentations d'éventuelles nouvelles entités.

| Système | R@1 | R@5 | R@10 | R@100 |
|---|---|---|---|---|
| `EL` | 0,31 | 0,56 | 0,67 | 0,89 |
| `MD + EL` | 0,84 | 0,91 | 0,93 | 0,97 |

TABLE 5 – Performances sur la tâche *Entity Linking* (EL) sur le jeu de données `frwiki EL`.

Ici encore, entraîner conjointement un modèle sur les tâches MD et EL semble être très bénéfique. Toutefois, les résultats obtenus par les deux systèmes ne sont pas comparables directement, puisque les données d'évaluation sont structurées différemment. Le système `EL` reçoit un document dans lequel une unique mention est annotée entre des balises `[E]` et `[/E]`. De ce fait, l'objectif du système est de produire une **unique** représentation pour cette entité. Ainsi, dans 17 % des cas, l'entité à prédire apparaît dans les 100 candidats les plus probables.

Le système `MD + EL` doit, quant à lui, détecter **toutes** les mentions présentes dans un document dépourvu d'annotations et calculer leurs représentations. Cependant, les performances données en Tableau 5 sont calculées uniquement sur les entités détectées par le modèle. Or, comme montré en Tableau 3, le modèle `MD + EL` a tendance à ignorer un tiers des mentions. Par conséquent, ce système est alors évalué en réalité sur seulement deux tiers des mentions présentes dans le jeu de test, là où le système `EL` est évalué sur l'intégralité des mentions. Il est donc difficile, de comparer objectivement ces deux systèmes. Le système `MD + EL` présente toutefois l'intérêt d'être autonome, là où le système `EL` requiert, en amont, de détecter les entités dans le texte. Le système `MD + EL` que nous proposons peut alors être préférable dans des situations où les ressources sont limitées, alors qu'un enchaînement de système peut produire des résultats de meilleure qualité, au prix d'un coût plus élevé en termes de puissance de calcul et/ou d'espace de stockage.

7. Ici, le R@$k$ est équivalent à la *top $k$ accuracy* puisqu'il ne peut y avoir qu'une entité pertinente pour chaque mention.

8. Ici, nous mesurons la similarité à l'aide du produit scalaire.

# 5 Conclusion

L'objectif de ces travaux est de proposer une solution de géolocalisation d'informations associées à des catastrophes naturelles, à partir du flux provenant des réseaux sociaux, en particulier de la plateforme Twitter. Pour ce faire, nous avons choisi une approche *Entity Linking* afin d'associer les mentions d'entités spatiales présentes dans un tweet aux entités présentes dans une base de connaissances telle que Wikidata. Cette approche possède plusieurs avantages, le premier étant qu'il est possible d'exploiter les liens internes à Wikipédia comme des signaux de supervision et d'entraîner un modèle en conséquence. D'autre part, les entités spatiales présentes dans Wikidata contiennent non seulement des informations de localisation, mais également d'autres propriétés permettant de mieux appréhender, comprendre et anticiper le phénomène.

Nous proposons une évolution de l'architecture dédiée à l'*Entity Linking* proposée par BOTHA, SHAN et GILLICK [2] afin de lever certaines limitations, à savoir :

— la nécessité de reposer sur un système de détection d'entités en amont ;
— le traitement individuel de chaque entité.

Ainsi, l'architecture que nous proposons permet de, conjointement, détecter les entités présentes dans un document et d'en calculer des représentations en une passe, là où le système original (1) requiert d'annoter les entités au préalable et (2) ne peut produire qu'une représentation à la fois. Ces modifications ont été motivées par la nécessité, en cas de crise, de fournir une réponse rapide aux autorités et aux services d'urgence. C'est pourquoi, nous cherchons à limiter le nombre de traitements nécessaires pour produire une réponse. De plus, nous avons pu confirmer le fait qu'il semble y avoir un intérêt à mêler NER et EL au sein d'un même modèle, comme observé par MARTINS, MARINHO et MARTINS [17].

Bien que les performances de notre système nous semble satisfaisantes, nous envisageons d'intégrer un mécanisme de filtrage des entités candidates, en introduisant, par exemple, un sous-système tel que celui proposé par WU et al. [27]. De plus, les résultats présentés dans cet article ont été obtenu sur une petite fraction du corpus Wikipédia. Il convient de valider ces résultats sur un jeu de données de plus grande taille et, surtout, sur un jeu de données provenant de réseaux sociaux. Nos travaux futurs viseront donc à collecter et annoter des messages postés sur Twitter lors de catastrophes naturelles.

# Remerciements

Les travaux présentés dans cet articles ont été effectués dans le cadre du projet RéSoCIO (ANR-20-CE39-001) cofinancé par l'Agence Nationale de la Recherche (ANR).

# 6 Bibliographie

# Références

[1] Marco AVVENUTI et al. « GSP (Geo-Semantic-Parsing) : geoparsing and geotagging with machine

learning on top of linked data ». In : *European Semantic Web Conference*. Springer. 2018, p. 17-32.

[2] Jan A BOTHA, Zifei SHAN et Daniel GILLICK. « Entity linking in 100 languages ». In : *arXiv preprint arXiv :2011.02690* (2020).

[3] Gaëtan CAILLAUT et al. « Automated construction of a French Entity Linking dataset to geolocate social network posts in the context of natural disasters ». In : *Proceedings of the 19th ISCRAM Conference, Tarbes, France*. 2022.

[4] Jacob DEVLIN et al. « Bert : Pre-training of deep bidirectional transformers for language understanding ». In : *arXiv preprint arXiv :1810.04805* (2018).

[5] Mohammad EBRAHIMI et al. « A unified neural network model for geolocating twitter users ». In : *Proceedings of the 22nd Conference on Computational Natural Language Learning*. 2018, p. 42-53.

[6] Yuan FANG et Ming-Wei CHANG. « Entity linking on microblogs with spatial and temporal signals ». In : *Transactions of the Association for Computational Linguistics* 2 (2014), p. 259-272.

[7] Rosemary FAYJALOUN et al. « Integrating strong-motion recordings and twitter data for a rapid shakemap of macroseismic intensity ». In : *International Journal of Disaster Risk Reduction* 52 (2021), p. 101927.

[8] Daniel GILLICK, Alessandro PRESTA et Gaurav Singh TOMAR. « End-to-end retrieval in continuous space ». In : *arXiv preprint arXiv :1811.08008* (2018).

[9] Daniel GILLICK et al. « Learning dense representations for entity retrieval ». In : *arXiv preprint arXiv :1909.10506* (2019).

[10] Rob GRACE. « Toponym usage in social media in emergencies ». In : *International Journal of Disaster Risk Reduction* 52 (2021), p. 101923.

[11] Milan GRITTA, Mohammad PILEHVAR et Nigel COLLIER. « Which melbourne ? augmenting geocoding with maps ». In : *56th Annual Meeting of the Association for Computational Linguistics*. Melbourne, Australia, 2018, p. 1285-1296.

[12] Bo HAN, Paul COOK et Timothy BALDWIN. « Text-based twitter user geolocation prediction ». In : *Journal of Artificial Intelligence Research* 49 (2014), p. 451-500.

[13] Bahareh HARANDIZADEH et Sameer SINGH. « Tweeki : Linking named entities on Twitter to a knowledge graph ». In : *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*. 2020.

[14] Jing LI et al. « A Survey on Deep Learning for Named Entity Recognition ». In : *IEEE Transactions on Knowledge and Data Engineering* 34.1 (2020), p. 50-70.

[15] Cédric LOPEZ et al. « Cap 2017 challenge : Twitter named entity recognition ». In : *arXiv preprint arXiv :1707.07568* (2017).

[16] Louis MARTIN et al. « CamemBERT : a Tasty French Language Model ». In : *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online : Association for Computational Linguistics, juill. 2020, p. 7203-7219. URL : https : / / www . aclweb . org / anthology/2020.acl-main.645.

[17] Pedro Henrique MARTINS, Zita MARINHO et André FT MARTINS. « Joint learning of named entity recognition and entity linking ». In : *arXiv preprint arXiv :1907.08243* (2019).

[18] Fernando MELO et Bruno MARTINS. « Automated geocoding of textual documents : A survey of current approaches ». In : *Transactions in GIS* 21.1 (2017), p. 3-38.

[19] Stephen MERITY et al. « Pointer sentinel mixture models ». In : *arXiv preprint arXiv :1609.07843* (2016).

[20] Martin MÜLLER et Florian LAURENT. « Cedille : A large autoregressive French language model ». In : *arXiv preprint arXiv :2202.03371* (2022).

[21] Seema NAGAR, Aaditeshwar SETH et Anupam JOSHI. « Characterization of social media response to natural disasters ». In : *Proceedings of the 21st international conference on world wide web*. 2012, p. 671-674.

[22] Joel NOTHMAN et al. « Learning multilingual named entity recognition from Wikipedia ». In : *Artificial Intelligence* 194 (2013), p. 151-175.

[23] Jorge David Gonzalez PAULE, Yeran SUN et Yashar MOSHFEGHI. « On fine-grained geolocalisation of tweets and real-time traffic incident detection ». In : *Information Processing & Management* 56.3 (2019), p. 1119-1132.

[24] Robert J STEED et al. « Crowdsourcing triggers rapid, reliable earthquake locations ». In : *Science advances* 5.4 (2019), eaau9824.

[25] Pedro Javier Ortiz SUÁREZ et al. « Establishing a new state-of-the-art for French named entity recognition ». In : *arXiv preprint arXiv :2005.13236* (2020).

[26] Ashish VASWANI et al. « Attention is All you Need ». In : *NIPS*. 2017, p. 5998-6008.

[27] Ledell WU et al. « Scalable zero-shot entity linking with dense entity retrieval ». In : *arXiv preprint arXiv :1911.03814* (2019).

[28] Canwen XU et al. « DLocRL : A deep learning pipeline for fine-grained location recognition and linking in tweets ». In : *The World Wide Web Conference*. 2019, p. 3391-3397.

[29] Zheren YAN et al. « The Integration of Linguistic and Geospatial Features Using Global Context Embedding for Automated Text Geocoding ». In : *ISPRS International Journal of Geo-Information* 10.9 (2021), p. 572.