

Sélection globale de segments pour la reconnaissance d’entités nommées

Urchade Zaratiana^{*†}, Niama El Khbir[†], Pierre Holat^{*†},
Nadi Tomeh[†], Thierry Charnois[†]

^{*} FI Group, [†] LIPN, CNRS UMR 7030, France

{zaratiana, elkhbir, holat, tomeh, charnois}@lipn.fr

RÉSUMÉ

La reconnaissance d’entités nommées est une tâche importante en traitement automatique du langage naturel avec des applications dans de nombreux domaines. Dans cet article, nous décrivons une nouvelle approche pour la reconnaissance d’entités nommées, dans laquelle nous produisons un ensemble de segmentations en maximisant un score global. Pendant l’entraînement, nous optimisons notre modèle en maximisant la probabilité de la segmentation correcte. Pendant l’inférence, nous utilisons la programmation dynamique pour sélectionner la meilleure segmentation avec une complexité linéaire. Nous prouvons que notre approche est supérieure aux modèles champs de Markov conditionnels et Semi-CRF pour la reconnaissance d’entités nommées.

ABSTRACT

Global Span Selection for Named Entity Recognition

Named Entity Recognition is an important task in Natural Language Processing with applications in many domains. In this paper, we describe a novel approach to named entity recognition, in which we output a set of spans (i.e., segmentations) by maximizing a global score. During training, we optimize our model by maximizing the probability of the gold segmentation. During inference, we use dynamic programming to select the best segmentation under a linear time complexity. We prove that our approach outperforms CRF and semi-CRF models for Named Entity Recognition.

MOTS-CLÉS : Reconnaissance d’entités nommées, segmentation, Champ aléatoire conditionnel.

KEYWORDS: Named entity recognition, segmentation, Conditional Random Fields.

1 Introduction

La reconnaissance d’entités nommées (REN) est une tâche cruciale du traitement du langage naturel dont le but est d’identifier et de classer les entités pertinentes dans les textes telles que les personnes, les organisations et les lieux. La reconnaissance de telles entités est avantageuse pour des applications telles que l’extraction de relations (El Khbir *et al.*, 2022) et la construction de taxonomie (Dauxais *et al.*, 2022). Il existe deux paradigmes principaux pour la reconnaissance d’entités : l’étiquetage de séquences (ES) (Huang *et al.*, 2015; Lample *et al.*, 2016; Akbik *et al.*, 2018) et les approches basées sur les segments (ABS) (Sohrab & Miwa, 2018; Yu *et al.*, 2020a; Li *et al.*, 2021). L’ES considère la reconnaissance d’entités comme une prédiction au niveau du jeton, en utilisant par exemple les schémas BIO (Ramshaw & Marcus, 1995) ou BILOU (Ratinov & Roth, 2009), tandis que les ABS

considèrent les segments (segments contigus de jetons) comme des unités de base au lieu des jetons et effectuent une classification au niveau du segment en attribuant une étiquette à chaque entité et une étiquette spéciale `null` aux segments sans entités (aussi segments non-entités).

L'ES est généralement réalisé en représentant les jetons à l'aide de modèles d'apprentissage profond, puis en utilisant un champ aléatoire conditionnel (Lafferty *et al.*, 2001) comme couche de sortie. La meilleure séquence d'étiquettes est calculée à l'aide de l'algorithme de Viterbi et l'apprentissage maximise typiquement la vraisemblance des séquences de référence. En revanche, les ABS énumèrent tous les segments candidats d'un texte d'entrée et calculent leur représentation avant de les fournir à une couche softmax pour la classification. L'un des avantages des ABS est qu'elles permettent une représentation plus riche des segments en comparaison à l'ES, puisque les caractéristiques au niveau des segments sont apprises de bout en bout.

Cependant, ces modèles basés sur les segments *unstructurés* prédisent l'étiquette de chaque segment indépendamment. Ils ont tendance à produire des entités qui se chevauchent, ce qui est interdit dans la reconnaissance d'entités plate et imbriquée. Les travaux antérieurs utilisaient un algorithme de décodage (Johnson, 1973; Yu *et al.*, 2020b; Li *et al.*, 2021) pour obtenir un ensemble d'entités non chevauchantes. Les entités ayant obtenu le meilleur score sont sélectionnées de manière itérative tant qu'elles ne chevauchent pas avec celles sélectionnées précédemment. Le décodage glouton est efficace mais tend à souffrir d'un biais de myopie. Le fait de choisir des segments sans tenir compte des décisions futures peut conduire à des ensembles d'entités sous-optimaux.

Une formulation alternative de la REN sous forme de segmentation et d'étiquetage joints avec des semi-champs de Markov conditionnels (Semi-CRF) a été proposée dans la littérature : (Sarawagi & Cohen, 2005; Kong *et al.*, 2016; Ye & Ling, 2018). Cette approche présente deux avantages : (a) elle utilise un modèle globalement normalisé pour calculer la probabilité de chaque segmentation étiquetée, au lieu d'évaluer chaque segment indépendamment ; et (2) elle garantit le non-chevauchement des entités de sortie en utilisant une variante de l'algorithme de Viterbi pour le décodage.

Néanmoins, les Semi-CRF sont moins performants en pratique, comme nous le montrons dans nos expériences. Nous supposons que l'évaluation de segmentations composées d'entités et de non-entités à la fois est la principale faiblesse. Tout d'abord, les segments sans entités peuvent être segmentés de plusieurs façons, toutes aussi valides les unes que les autres, mais une seule d'entre elles est appliquée par les Semi-CRF, à la fois pendant l'apprentissage et l'inférence. De plus, la majorité des segments ne comportant pas d'entité, une masse considérable de probabilité est gaspillée sur des segmentations inintéressantes.

Dans cet article, nous proposons une nouvelle formulation pour la REN basée sur les segments qui combine des idées provenant d'approches en deux étapes (filtrage et décodage) et de modèles basés sur des champs de Markov conditionnels (CRF) globalement normalisés. Notre approche commence par le filtrage de tous les segments sans entités à l'aide d'un classificateur de segments et la construction d'un graphe (*chevauchant*) des segments restants. Un modèle globalement normalisé est ensuite utilisé pour calculer la probabilité de chaque *ensemble indépendant maximal* (EIM) dans le graphe. Chacun de ces ensembles correspond à une sélection d'entités non chevauchantes. L'apprentissage et l'inférence peuvent être réalisés efficacement en utilisant la programmation dynamique, comme nous l'expliquons dans la section 2.2. De plus, nous entraînons le classificateur de segments et le modèle global de sélection d'entités de manière jointe en utilisant un objectif multi-tâches. Nous montrons que notre approche surpasse à la fois l'ES et les Semi-CRF sur toutes les tâches et surpasse les modèles sur les deux étapes du filtrage et du décodage glouton dans la plupart des cas.

2 La REN basée sur les segments en deux étapes

Les approches de la littérature basées sur les segments utilisent un classificateur de segments non structuré et localement normalisé pour filtrer les segments non-entités, suivi d'un décodage glouton pour sélectionner un ensemble d'entités non chevauchantes (Li *et al.*, 2021; Fu *et al.*, 2021). Nous décrivons ces deux étapes dans cette section.

2.1 La classification de segments

Cette étape consiste à énumérer tous les segments de la séquence d'entrée et à calculer leur représentations en utilisant des transformeurs pré-entraînés tels que BERT. Conformément aux travaux antérieurs (Lee *et al.*, 2017; Luan *et al.*, 2019), la représentation s_{ij} d'un segment (i, j) de longueur k est calculée en concaténant la représentation de ses jetons d'extrémité gauche et droite (h_i et h_j respectivement) avec une caractéristique apprise de largeur du segment f_k . Un perceptron multicouche à deux couches avec activation *ReLU* est appliqué aux caractéristiques pour obtenir la représentation finale du segment

$$s_{ij} = \text{MLP}([h_i; h_j; f_k]) \quad (1)$$

Ensuite, la représentation du segment est introduite dans une couche linéaire (ou un perceptron multicouche) pour la classification du segment. Une tâche de REN avec L types d'entités aurait $L + 1$ étiquettes puisque nous allouons une étiquette `null` pour les segments non-entités. Le score de l'étiquette y pour un segment (i, j) est calculée comme suit :

$$\phi(i, j, y) = w_y^T s_{ij} \quad (2)$$

où w_y est un vecteur de poids apprenable (nous omettons le terme de biais pour des raisons de lisibilité). Ces scores sont ensuite normalisés à l'aide de la fonction softmax.

Le modèle est entraîné pour minimiser la log-vraisemblance négative des segments de référence de l'ensemble d'entraînement \mathcal{T} :

$$\mathcal{L}_{\text{clf}} = - \sum_{(i,j,y) \in \mathcal{T}} \log \frac{\exp \phi(i, j, y)}{\sum_{y'} \exp \phi(i, j, y')} \quad (3)$$

Pendant l'inférence, chaque segment (i, j) se voit attribuer l'étiquette $y(i, j) = \underset{y}{\text{argmax}} \phi(i, j, y)$ avec le score $k(i, j) = \max_y \phi(i, j, y)$. Nous appelons \mathcal{C} l'ensemble des entités candidates qui est l'ensemble de tous les segments auxquels on a attribué une étiquette différente de `null`. Cet ensemble peut contenir des segments qui se chevauchent, ce qui n'est pas autorisé dans les tâches de reconnaissance plate, une étape de décodage est donc nécessaire.

2.2 Ensemble indépendant à poids maximal dans les graphes d'intervalles

Un *graphe de chevauchement* sur \mathcal{C} est le graphe G dont les nœuds sont les éléments de \mathcal{C} et qui contient une arête entre chaque paire d'entités chevauchantes. Ce graphe peut également être appelé un *graphe d'intervalles* puisque les segments peuvent être vues comme des intervalles de leurs positions de début et de fin. Un *ensemble indépendant (EI)* du graphe G est un ensemble de nœuds tel que deux nœuds de cet ensemble ne sont pas reliés par une arête. Un ensemble indépendant est dit *maximal* s'il n'est pas correctement contenu dans un autre ensemble indépendant. Chaque nœud (i, j)

du graphe se voit attribuer un nombre réel $r(i, j)$, le graphe G est dit être un graphe *pondéré*. Pour chaque sous-ensemble de nœuds $\mathcal{S} \subseteq \mathcal{C}$, $\sum_{(i,j) \in \mathcal{S}} r(i, j)$ est appelé le poids de \mathcal{S} . Un *Ensemble indépendant de poids maximal (EIPM)* est un ensemble indépendant tel que son poids est maximal sur tous les ensembles indépendants. Sous cette formulation, le problème du décodage revient à trouver un EIPM dans le graphe G :

$$\mathcal{S} = \operatorname{argmax}_{\mathcal{S} \in \Psi(\mathcal{C})} \sum_{(i,j) \in \mathcal{S}} r(i, j) \quad (4)$$

où $\Psi(\mathcal{C})$ est l'ensemble de tous les EIM de G .

Le décodage glouton Le décodage glouton construit une approximation de \mathcal{S} en ajoutant itérativement l'entité de \mathcal{C} ayant le meilleur score et qui ne se chevauche avec aucune entité précédemment sélectionnée. Cet algorithme est d'une complexité de $O(n \log n)$ avec $n = |\mathcal{C}|$.

Dans la section suivante, nous proposons une alternative exacte qui utilise un modèle globalement normalisé.

Le décodage exact La solution exacte de l'équation (4) peut être obtenue par la programmation dynamique en utilisant l'algorithme EIPM présenté par Gupta *et al.* (1982); Hsiao *et al.* (1992). Cet algorithme a une complexité temporelle linéaire en $O(n)$ avec n le nombre de nœuds dans le graphe, qui est supposé être trié par les extrémités d'intervalles, sinon, il peut être trié en un temps de $O(n \log n)$. En pratique, le nombre de nœuds n est bien inférieur à la longueur de la séquence d'entrée.

2.3 Un modèle EIPM globalement normalisé

Une façon d'estimer les poids $r(i, j)$ des nœuds du graphe est d'utiliser les scores produits par les classificateurs locaux : $r(i, j) = k(i, j)$. Dans cette section, nous proposons d'apprendre un modèle probabiliste dédié du EIM globalement normalisé et entraîné pour maximiser la probabilité du EIM de référence.

La probabilité d'un EIM est calculée par :

$$P(\mathcal{S}) = \mathcal{Z}^{-1} \exp \sum_{(i,j) \in \mathcal{S}} r(i, j) \quad (5)$$

Le score non normalisé d'un EIM est simplement égal à la somme des poids individuels des segments, chacune étant une projection linéaire de la représentation du segment :

$$r(i, j) = w^T s_{ij} \quad (6)$$

où w est un vecteur de paramètres à apprendre. La constante de normalisation est donnée par :

$$\mathcal{Z} = \sum_{\mathcal{S} \in \Psi(\mathcal{C})} \exp \sum_{(i,j) \in \mathcal{S}} r(i, j) \quad (7)$$

Alors que \mathcal{Z} , la fonction de partition, peut être ignorée pendant l'inférence, elle doit être calculée pendant l'apprentissage car nous utilisons la log probabilité négative du EIM or comme fonction de perte. La fonction de partition peut être calculée efficacement en utilisant une modification au programme dynamique de l'algorithme EIPM. Cependant, en pratique, nous énumérons simplement tous les EIM, ce qui est faisable puisque le nombre de segments restants est faible. L'énumération peut

Modèles	Conll-2003			OntoNotes 5.0			TDM			ACE Arabe		
	P	R	F	P	R	F	P	R	F	P	R	F
CRF	92.64	91.82	92.23	87.77	89.47	88.61	69.77	73.65	71.66	82.79	84.44	83.61
Semi-CRF	91.46	90.77	91.11	87.44	88.85	88.14	69.38	72.85	71.05	82.97	84.24	83.60
Standard	93.40	91.68	92.53	89.47	90.00	89.73	67.75	69.88	68.78	83.21	83.76	83.48
+ Glouton	93.82	91.40	92.60	90.43	89.04	89.73	75.12	67.82	71.26	83.73	83.56	83.64
+ Global	93.83	91.51	92.65	90.58	89.45	90.01	75.25	68.12	71.48	83.72	83.55	83.63
Global	94.84	90.72	92.73	89.05	89.77	89.41	63.30	72.75	67.53	83.54	83.65	83.60
+ Glouton	95.07	90.42	92.69	89.98	88.44	89.21	74.16	68.23	71.07	83.87	82.75	83.31
+ Global	95.11	90.52	92.76	90.18	88.85	89.51	75.55	70.34	72.84	84.14	83.35	83.74

TABLE 1 – **Résultats expérimentaux.** Nous rapportons la moyenne sur trois échantillons aléatoires.

être effectuée en un temps $O(n^2 + \beta)$ où n est le nombre de segments et β la somme des nombres de segments de tous les ensembles énumérés (Leung, 1984; Liang *et al.*, 1991).

Pendant l’apprentissage, nous modifions l’ensemble \mathcal{C} , c’est-à-dire la sortie du classificateur local, de sorte (1) qu’il contienne tous les segments de référence, et (2) qu’il ne contienne pas de segments ne se chevauchant pas avec les segments de référence. Ce faisant, nous nous assurons que les intervalles or forment un EIM dans le graphe de chevauchement sur \mathcal{C} . Enfin, nous utilisons une fonction de perte multitâche qui correspond à la somme de la perte du classificateur local (Eq. (3)) et de la perte du modèle global.

3 Expériences

3.1 Configuration

Baselines Nous comparons notre approche à un étiqueteur CRF, au modèle standard ABS et au modèle basé sur les segments avec Semi-CRF. Pour tous les modèles, nous utilisons des transformeurs pré-entraînés pour la représentation des jetons.

Jeux de données Nous évaluons notre modèle sur divers jeux de données de REN : TDM (Hou *et al.*, 2021), Conll-2003 (Tjong Kim Sang & De Meulder, 2003), et OntoNotes 5.0 (Weischedel *et al.*, 2013) pour les données anglaises, et ACE05 pour les données arabes (Walker *et al.*, 2006).

Mesures d’évaluation Nous évaluons les modèles en utilisant la correspondance exacte entre les entités prédites et les entités de référence. Nous rapportons la précision, le rappel et le F1.

Hyperparamètres Pour les jeux de données Conll-2003 et Ontonotes, nous utilisons `bert-base-cased`. (Devlin *et al.*, 2019) pour produire la représentation contextuelle, pour TDM nous utilisons SciBERT (Beltagy *et al.*, 2019) et pour ACE Arabe nous utilisons `bert-base-arabertv2` (Antoun *et al.*, 2020). Nous utilisons la taille de base, avec 12 couches de transformeurs, pour tous les modèles. Nous n’utilisons pas d’incorporations auxiliaires (par exemple l’*incorporation de caractères*) pour des raisons de simplicité. Tous les modèles sont entraînés avec l’optimiseur de Kingma & Ba (2017) avec un taux d’apprentissage de $2e-5$, une taille de lot de 10 et une époque maximale de 25. Nous conservons le meilleur point de contrôle sur l’ensemble de validation pour les tests. Nous avons entraîné tous les modèles sur un serveur équipé de GPU V100.

3.2 Résultats

Nous rapportons les résultats de nos expériences dans la Table 1 pour les quatre ensembles de données en utilisant les modèles CRF, Semi-CRF, Standard et Global basés sur les segments. Pour les modèles Standard et Global, nous rapportons les résultats obtenus en utilisant (cf. + lignes Global) ou en n'utilisant pas le décodage (cf. + lignes Glouton).

Résultats principaux D'après la Table 1, nos modèles globaux avec décodage global obtiennent les meilleurs résultats sur la plupart des ensembles de données (tous sauf sur OntoNotes). De plus, le Semi-CRF a le score le plus bas sur toutes les données, ce qui peut expliquer sa faible adoption au fil des ans par rapport au CRF standard.

Décodage global vs. décodage glouton Pour les deux approches basées sur les segments, nous constatons que le décodage améliore généralement la performance du score F1 et la précision tout en diminuant le rappel. Nous pouvons expliquer ce comportement par le fait que quand on utilise le décodage, les segments non fiables sont supprimés, ce qui augmente la précision. Cependant, certains faux négatifs peuvent également être supprimés, d'où la légère diminution du rappel. De plus, pour les modèles standards, le décodage glouton et le décodage global ont des performances similaires, tandis que pour les modèles entraînés globalement, le décodage global a toujours les meilleures performances, ce qui montre l'efficacité de notre approche. De plus, nous pouvons observer sur les jeux de données Conll-2003, ACE Arabe et OntoNotes 5.0 que le décodage glouton peut même diminuer la performance du modèle, ce qui serait un effet du biais myopique.

4 Travaux connexes

Approches pour la REN Traditionnellement, les tâches de REN sont conçues comme un ES (Lample *et al.*, 2016; Akbik *et al.*, 2018), c'est-à-dire une classification au niveau du jeton. Récemment, de nombreuses approches ont été proposées qui vont au-delà de la prédiction au niveau du token. Par exemple, certains travaux ont abordé la REN nommées comme une tâche de réponse aux questions (Li *et al.*, 2020) et d'autres utilisent des modèles de séquence-à-séquence (Yan *et al.*, 2021; Yang & Tu, 2022). Dans ce travail, nous nous sommes concentrés sur les ABS (Liu *et al.*, 2016; Sohrab & Miwa, 2018; Fu *et al.*, 2021; Zaratiana *et al.*, 2022b,c,a).

Décodage pour la REN La REN est une tâche pour laquelle un algorithme de décodage doit être appliqué afin de garantir que les sorties du modèle sont bien entraînées. Par exemple, les CRF (Lafferty *et al.*, 2001) ont été proposés pour l'ES et le Semi-CRF pour les ABS. En raison de la faible performance du Semi-CRF (Sarawagi & Cohen, 2005), les chercheurs ont proposé d'entraîner une méthode locale basée sur les segments et d'utiliser un décodage glouton pour garantir des entités non chevauchantes pour le décodage. Dans ce travail, nous proposons un décodage exact/global pour produire un ensemble de segments non chevauchants qui maximisent le score global afin d'éviter le biais myope de l'approche gloutonne.

5 Conclusion

Dans ce travail, nous proposons une nouvelle approche de reconnaissance d'entités nommées basée sur les segments. Notre modèle atténue le biais myope de l'approche standard et obtient de meilleurs résultats que les modèles structurés tels que les CRF ou les Semi-CRF. Des travaux futurs pourraient explorer l'interaction entre les segments pour améliorer davantage les performances.

Remerciements

Ces travaux ont bénéficié d'un accès aux moyens de calcul de l'IDRIS au travers de l'allocation de ressources 2022-AD011013096R1 attribuée par GENCI.

Références

- AKBIK A., BLYTHE D. & VOLLGRAF R. (2018). Contextual string embeddings for sequence labeling. In *Proceedings of the 27th International Conference on Computational Linguistics*, p. 1638–1649, Santa Fe, New Mexico, USA : Association for Computational Linguistics.
- ANTOUN W., BALY F. & HAJJ H. M. (2020). Arabert : Transformer-based model for arabic language understanding. *ArXiv*, **abs/2003.00104**.
- BELTAGY I., LO K. & COHAN A. (2019). Scibert : A pretrained language model for scientific text.
- BENAMARA F., HATOUT N., MULLER P. & OZDOWSKA S., Éds. (2007). *Actes de TALN 2007 (Traitement automatique des langues naturelles)*, Toulouse. ATALA, IRIT.
- DAUXAIS Y., ZARATIANA U., LANEUVILLE M., HERNANDEZ S. D., HOLAT P. & GROSMAN C. (2022). Towards automation of topic taxonomy construction. In T. BOUADI, E. FROMONT & E. HÜLLERMEIER, Éds., *Advances in Intelligent Data Analysis XX*, p. 26–38, Cham : Springer International Publishing.
- DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, p. 4171–4186, Minneapolis, Minnesota : Association for Computational Linguistics. DOI : [10.18653/v1/N19-1423](https://doi.org/10.18653/v1/N19-1423).
- DIAS G., Éd. (2015). *Actes de TALN 2015 (Traitement automatique des langues naturelles)*, Caen. ATALA, HULTECH.
- EL KHBIR N., TOMEH N. & CHARNOIS T. (2022). ArabIE : Joint entity, relation and event extraction for Arabic. In *Proceedings of the The Seventh Arabic Natural Language Processing Workshop (WANLP)*, p. 331–345, Abu Dhabi, United Arab Emirates (Hybrid) : Association for Computational Linguistics.
- FU J., HUANG X. & LIU P. (2021). SpanNER : Named entity re-/recognition as span prediction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 7183–7195, Online : Association for Computational Linguistics. DOI : [10.18653/v1/2021.acl-long.558](https://doi.org/10.18653/v1/2021.acl-long.558).
- GUPTA U. I., LEE D. T. & LEUNG J. Y.-T. (1982). Efficient algorithms for interval graphs and circular-arc graphs. *Networks*, **12**, 459–467.
- HOU Y., JOCHIM C., GLEIZE M., BONIN F. & GANGULY D. (2021). TDMSci : A specialized corpus for scientific literature entity tagging of tasks datasets and metrics. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics : Main Volume*, p. 707–714, Online : Association for Computational Linguistics. DOI : [10.18653/v1/2021.eacl-main.59](https://doi.org/10.18653/v1/2021.eacl-main.59).

HSIAO J. Y., TANG C. Y. & CHANG R. S. (1992). An efficient algorithm for finding a maximum weight 2-independent set on interval graphs. *Information Processing Letters*, **43**(5), 229–235. DOI : [https://doi.org/10.1016/0020-0190\(92\)90216-I](https://doi.org/10.1016/0020-0190(92)90216-I).

HUANG Z., XU W. & YU K. (2015). Bidirectional lstm-crf models for sequence tagging.

JOHNSON D. S. (1973). Approximation algorithms for combinatorial problems. *Proceedings of the fifth annual ACM symposium on Theory of computing*.

KINGMA D. P. & BA J. (2017). Adam : A method for stochastic optimization.

KONG L., DYER C. & SMITH N. A. (2016). Segmental recurrent neural networks. *CoRR*, **abs/1511.06018**.

LAFFERTY J. D., MCCALLUM A. & PEREIRA F. C. N. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning, ICML '01*, p. 282–289, San Francisco, CA, USA : Morgan Kaufmann Publishers Inc.

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

LAMPLE G., BALLESTEROS M., SUBRAMANIAN S., KAWAKAMI K. & DYER C. (2016). Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 260–270, San Diego, California : Association for Computational Linguistics. DOI : [10.18653/v1/N16-1030](https://doi.org/10.18653/v1/N16-1030).

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

LEE K., HE L., LEWIS M. & ZETTLEMOYER L. (2017). End-to-end neural coreference resolution. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, p. 188–197, Copenhagen, Denmark : Association for Computational Linguistics. DOI : [10.18653/v1/D17-1018](https://doi.org/10.18653/v1/D17-1018).

LEUNG J. Y.-T. (1984). Fast algorithms for generating all maximal independent sets of interval, circular-arc and chordal graphs. *J. Algorithms*, **5**, 22–35.

LI X., FENG J., MENG Y., HAN Q., WU F. & LI J. (2020). A unified mrc framework for named entity recognition.

LI Y., LEMAO LIU & SHI S. (2021). Empirical analysis of unlabeled entity problem in named entity recognition. In *International Conference on Learning Representations*.

LIANG Y., DHALL S. & LAKSHMIVARAHAN S. (1991). On the problem of finding all maximum weight independent sets in interval and circular-arc graphs. In [*Proceedings*] 1991 Symposium on Applied Computing, p. 465–470. DOI : [10.1109/SOAC.1991.143921](https://doi.org/10.1109/SOAC.1991.143921).

LIU Y., CHE W., GUO J., QIN B. & LIU T. (2016). Exploring segment representations for neural segmentation models. In *IJCAI*.

LUAN Y., WADDEN D., HE L., SHAH A., OSTENDORF M. & HAJISHIRZI H. (2019). A general framework for information extraction using dynamic span graphs.

RAMSHAW L. A. & MARCUS M. P. (1995). Text chunking using transformation-based learning. *ArXiv*, **cmp-lg/9505040**.

- RATINOV L. & ROTH D. (2009). Design challenges and misconceptions in named entity recognition. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)*, p. 147–155, Boulder, Colorado : Association for Computational Linguistics.
- SARAWAGI S. & COHEN W. W. (2005). Semi-markov conditional random fields for information extraction. In L. SAUL, Y. WEISS & L. BOTTOU, Éd.s., *Advances in Neural Information Processing Systems*, volume 17 : MIT Press.
- SERETAN V. & WEHRLI E. (2007). Collocation translation based on sentence alignment and parsing. In (Benamara *et al.*, 2007), p. 401–410.
- SOHRAB M. G. & MIWA M. (2018). Deep exhaustive model for nested named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, p. 2843–2849, Brussels, Belgium : Association for Computational Linguistics. DOI : [10.18653/v1/D18-1309](https://doi.org/10.18653/v1/D18-1309).
- 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*, p. 142–147.
- WALKER C., STRASSEL S., MEDERO J. & MAEDA K. (2006). Ace 2005 multilingual training corpus. DOI : [10.35111/MWXC-VH88](https://doi.org/10.35111/MWXC-VH88).
- WEISCHEDEL R., PALMER M., MARCUS M., HOVY E., PRADHAN S., RAMSHAW L., XUE N., TAYLOR A., KAUFMAN J., FRANCHINI M., EL-BACHOUTI M., BELVIN R. & HOUSTON A. (2013). OntoNotes Release 5.0. DOI : [11272.1/AB2/MKJJ2R](https://doi.org/11272.1/AB2/MKJJ2R).
- YAN H., GUI T., DAI J., GUO Q., ZHANG Z. & QIU X. (2021). A unified generative framework for various ner subtasks. In *ACL*.
- YANG S. & TU K. (2022). Bottom-up constituency parsing and nested named entity recognition with pointer networks. In *ACL*.
- YE Z. & LING Z.-H. (2018). Hybrid semi-Markov CRF for neural sequence labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2 : Short Papers)*, p. 235–240, Melbourne, Australia : Association for Computational Linguistics. DOI : [10.18653/v1/P18-2038](https://doi.org/10.18653/v1/P18-2038).
- YU J., BOHNET B. & POESIO M. (2020a). Named entity recognition as dependency parsing. In *ACL*.
- YU J., BOHNET B. & POESIO M. (2020b). Named entity recognition as dependency parsing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, p. 6470–6476, Online : Association for Computational Linguistics. DOI : [10.18653/v1/2020.acl-main.577](https://doi.org/10.18653/v1/2020.acl-main.577).
- ZARATIANA U., ELKHBIR N., HOLAT P., TOMEH N. & CHARNOIS T. (2022a). Global span selection for named entity recognition. In *Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS)*, p. 11–17, Abu Dhabi, United Arab Emirates (Hybrid) : Association for Computational Linguistics.
- ZARATIANA U., TOMEH N., HOLAT P. & CHARNOIS T. (2022b). GNNer : Reducing overlapping in span-based NER using graph neural networks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics : Student Research Workshop*, p. 97–103, Dublin, Ireland : Association for Computational Linguistics. DOI : [10.18653/v1/2022.acl-srw.9](https://doi.org/10.18653/v1/2022.acl-srw.9).
- ZARATIANA U., TOMEH N., HOLAT P. & CHARNOIS T. (2022c). Named entity recognition as structured span prediction. In *Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS)*, p. 1–10, Abu Dhabi, United Arab Emirates (Hybrid) : Association for Computational Linguistics.