Détection d'événements à partir de peu d'exemples par seuillage dynamique

Aboubacar Tuo, Romaric Besançon, Olivier Ferret, Julien Tourille

Actes de CORIA-TALN 2023
Paris, France, 5 au 9 juin 2023

Détection d'événements à partir de peu d'exemples par seuillage dynamique

Aboubacar Tuo Romaric Besançon Olivier Ferret Julien Tourille
Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
{aboubacar.tuo,romaric.besancon,olivier.ferret,julien.tourille}@cea.fr

RÉSUMÉ

Les études récentes abordent la détection d'événements à partir de peu de données comme une tâche d'annotation de séquences en utilisant des réseaux prototypiques. Dans ce contexte, elles classifient chaque mot d'une phrase donnée en fonction de leurs similarités avec des prototypes construits pour chaque type d'événement et pour la classe nulle « non-événement ». Cependant, le prototype de la classe nulle agrège par définition un ensemble de mots sémantiquement hétérogènes, ce qui nuit à la discrimination entre les mots déclencheurs et non déclencheurs. Dans cet article, nous abordons ce problème en traitant la détection des mots non-déclencheurs comme un problème de détection d'exemples « hors-domaine » et proposons une méthode pour fixer dynamiquement un seuil de similarité pour cette détection.

ABSTRACT

Few Shot Event Detection with Dynamic Thresholding

Recent studies in few-shot event trigger detection from text address the task as a word sequence annotation task using prototypical networks. In this context, the classification of a word is based on the similarity of its representation to the prototypes built for each event type and for the "non-event" class. However, the "non-event" prototype aggregates by definition a set of semantically heterogeneous words, which hurts the discrimination between trigger and non-trigger words. We address this issue by handling the detection of non-trigger words as an out-of-domain detection problem and propose a method for dynamically setting a similarity threshold for this detection.

MOTS-CLÉS : Détection d'événements à partir de peu d'exemples, Méta-apprentissage.

KEYWORDS: Few-shot Event Detection, Meta-learning.

1 Introduction

La détection d'événements est une tâche de l'extraction d'information qui vise à extraire des instances de types d'événements donnés à partir de textes (Nguyen & Grishman, 2015a,b). Cette extraction consiste à identifier des déclencheurs d'événements, qui sont des groupes de mots indiquant explicitement la présence d'un événement dans une phrase. Par exemple, dans la phrase « *John D. Idol will [**take over**] as Chief Executive.* », un événement « Start-Position » est déclenché par le déclencheur « *take over* ». Les approches d'apprentissage supervisé pour la détection d'événements ont été largement étudiées ces dernières années, notamment les méthodes fondées sur des traits lexico-syntaxiques (Li *et al.*, 2013; Liao & Grishman, 2011), les réseaux neuronaux convolutifs (Nguyen & Grishman, 2015a), les réseaux neuronaux récurrents (Nguyen *et al.*, 2016) et les modèles fondés sur les graphes (Liu *et al.*, 2018; Nguyen & Grishman, 2018; Yan *et al.*, 2019). Cependant, toutes

Actes de la 30e Conférence sur le Traitement Automatique des Langues Naturelles (TALN), volume 4 : articles déjà soumis ou acceptés en conférence internationale, pages 159–168.

ces approches reposent sur des ensembles de données annotées conséquents pour l'entraînement, généralement difficiles à obtenir.

La détection d'événements à partir de peu d'exemples (*Few-Shot Event Detection*, FSED) a donc suscité un grand intérêt ces dernières années avec l'émergence de méthodes d'apprentissage à partir de peu de données, notamment via le méta-apprentissage (Snell *et al.*, 2017; Vinyals *et al.*, 2016; Sung *et al.*, 2018; Geng *et al.*, 2019), et le développement de modèles de langue pré-entraînés capables de transférer leurs connaissances linguistiques à de nouvelles tâches. Elle a été mise en œuvre sous plusieurs formes : *identification d'événement*, qui détermine si un mot dans une phrase est un déclencheur selon un type d'événement (Bronstein *et al.*, 2015; Chen *et al.*, 2021), *classification d'événement*, dont l'objectif est de choisir le type d'événement associé à un déclencheur déjà identifié dans une phrase (Shen *et al.*, 2021; Deng *et al.*, 2020; Lai & Nguyen, 2019; Lai *et al.*, 2020, 2021), et *la détection*, qui réalise ces deux étapes conjointement (Cong *et al.*, 2021; Tuo *et al.*, 2022).

Ces efforts de recherche ont fait de la FSED une tâche d'annotation de séquences, qui se transforme en un problème de classification de mots traité à l'aide de réseaux prototypiques (Snell *et al.*, 2017), qui sont particulièrement adaptés à l'apprentissage à partir de peu d'exemples. Dans ce contexte, un prototype est construit pour chaque type d'événement et la classe « non-événement » (aussi appelée classe nulle) (Yang & Katiyar, 2020; Cong *et al.*, 2021; Tuo *et al.*, 2022). Cependant, l'hétérogénéité intrinsèque du prototype « non-événement » rend difficile la discrimination entre les mots déclencheurs et non déclencheurs fondée sur la similarité avec les prototypes.

Pour résoudre ce problème, nous formulons la FSED comme un problème de détection hors domaine (Schölkopf *et al.*, 2001), en considérant les mots de la classe nulle comme des exemples hors-domaine et en apprenant un seuil de similarité dynamique afin que ces exemples ne soient associés à aucune classe d'événement. En résumé, notre contribution est triple : (1) nous proposons une nouvelle façon de traiter la classe nulle dans la FSED ; (2) nous définissons un nouveau modèle pour la FSED en utilisant des réseaux prototypiques et une fonction de coût contrastive ; (3) nous calculons un seuil de décision dynamique en utilisant la fonction de répartition empirique (*ECDF*). Ces contributions se traduisent par des gains significatifs, évalués sur plusieurs jeux de données [1].

2 Approche

2.1 Formulation du problème

Nous formulons la FSED comme un apprentissage épisodique à N-ways et k-shots (Vinyals *et al.*, 2016) avec des réseaux prototypiques. La tâche de détection des déclencheurs est prise comme un problème d'annotation de séquence au niveau des mots, en s'appuyant sur le format BIO (*Beginning-Inside-Outside*) comme dans Cong *et al.* (2021); Tuo *et al.* (2022) [2]. À chaque épisode, nous considérons un sous-ensemble de phrases annotées appelé *support set*. Il contient N types d'événements et k exemples annotés par type (k étant petit, par exemple de 1 à 10). Un second ensemble, appelé *query set*, est utilisé pour faire des prédictions fondées sur les exemples annotés du *support set*. Chaque phrase peut contenir un ou plusieurs déclencheurs, associés chacun à un type d'événements. L'identification de ce type et de la position du déclencheur est effectuée en attribuant une étiquette à chaque mot, ce qui correspond à une classification multi-classe au niveau des mots, avec autant de classes

1. Cet article reprend par ailleurs le travail présenté dans (Tuo *et al.*, 2023).
2. Les déclencheurs événementiels peuvent être des multi-mots, en pratique en très faible nombre dans les corpus d'évaluation, mais n'admettent pas d'insertion, ce qui rend le format BIO suffisant.

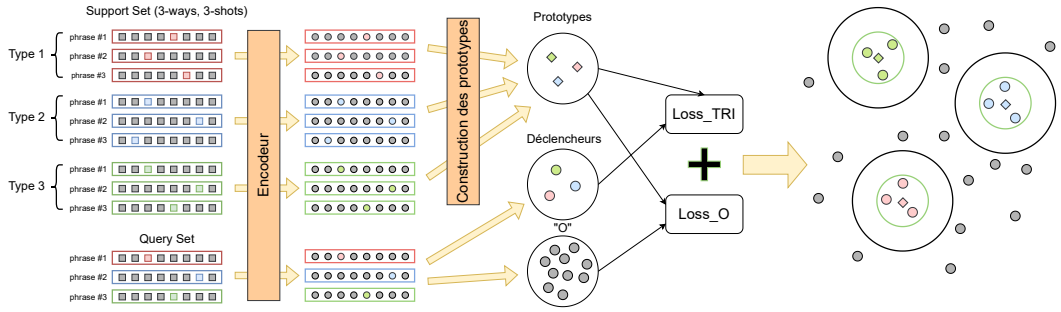


FIGURE 1 – Vue d'ensemble du modèle

que de types d'événements plus une classe nulle (étiquette « O ») pour les mots non-déclencheurs d'événements.

Nous construisons un prototype pour chaque classe à partir des exemples du *support set* en prenant la moyenne des représentations des k déclencheurs de cette classe. Ensuite, nous classons chaque mot du *query set* en fonction de sa similarité avec ces prototypes. Pendant l'apprentissage, ces similarités sont utilisées pour mettre à jour les poids du modèle via une fonction de coût. Cependant, cette formulation implique d'avoir un prototype pour la classe « O » qui est construit en pratique en rassemblant des mots qui ne sont pas sémantiquement homogènes. Nous proposons de traiter cette classe comme une classe « hors-domaine ». Inspirés par les efforts de recherche sur la classification d'exemple hors-domaine avec peu d'exemples (Tan *et al.*, 2019; Nimah *et al.*, 2021), nous évitons de construire le prototype « O » et proposons une approche fondée sur un seuillage dynamique adapté à chaque phrase en utilisant l'ECDF des similarités entre les mots et les prototypes.

2.2 Modèle

La figure 1 présente une vue d'ensemble de notre modèle dont les composants sont détaillés ci-après.

Encodeur Ce composant prend une phrase en entrée et produit une représentation contextuelle pour chaque mot. Pour une phrase $x = w_1, \ldots, w_L$, de longueur L, l'encodeur fournit $\bar{\mathbf{e}} = \mathbf{e_1}, \ldots, \mathbf{e_L}$, où $\mathbf{e_i}$ est la représentation du mot w_i.

Module prototypique Ce composant construit un prototype pour chaque type d'événements en faisant la moyenne des représentations des mots déclencheurs du *support set* et classe les mots du *query set* en fonction de leurs similarités avec ces prototypes. Contrairement à Tuo *et al.* (2022) et Cong *et al.* (2021), nous ne construisons pas de prototype pour la classe nulle. Nous nous fondons sur un seuil de similarité pour décider si un mot est déclencheur ou non.

Entraînement La fonction de coût habituellement utilisée dans les réseaux prototypiques est l'entropie croisée (*cross-entropy*). Nous proposons un apprentissage contrastif plus adapté à l'apprentissage de métrique. Contrairement à l'entropie croisée, dont l'objectif est d'apprendre à prédire une étiquette ou des valeurs à partir d'une entrée, les fonctions contrastives prédisent la similarité relative entre les entrées. Une telle fonction est plus appropriée dans notre cas puisque nous cherchons justement à rendre les déclencheurs plus proches de leurs prototypes que les mots « O » . Pour une classe y donnée, la fonction de coût comporte deux termes :

- **Loss-TRI** : rapproche le déclencheur $\mathbf{e_{tr}}$ de son prototype $\mathbf{c^y}$ et l'éloigne des autres prototypes $c^{j \neq y}$:

$$\mathcal{L}_{TRI}(\bar{\mathbf{e}}, y) = \sum_{j \neq y} \max(0, \mathcal{M}_0 + s(\mathbf{e_{tr}}, \mathbf{c^j}) - s(\mathbf{e_{tr}}, \mathbf{c^y})) \quad (1)$$

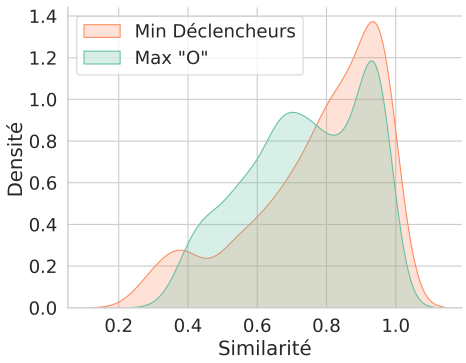
- **Loss-O** : éloigne les mots « O » $\mathbf{e_i}$ $(i \neq tr)$ de tous les prototypes c^j.

$$\mathcal{L}_O(\bar{\mathbf{e}}, y) = \max_{i \neq tr}(0, \max_j(s\left(\mathbf{e_i}, \mathbf{c^j}\right) - \mathcal{M}_1)) \quad (2)$$

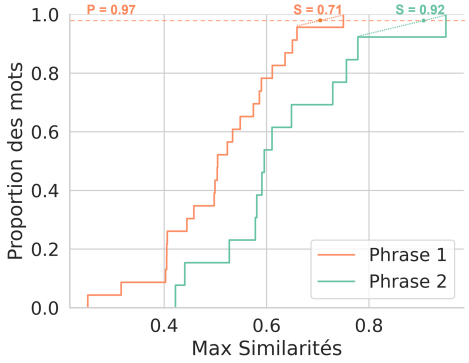
Dans ces fonctions de coût, la fonction $s(.)$ fait référence à la similarité entre la représentation d'un déclencheur et celle du prototype d'une classe tandis que $\mathcal{M}_0$ et $\mathcal{M}_1$ sont des hyperparamètres correspondant aux marges. Le modèle est entraîné en minimisant une fonction de coût correspondant à la somme de ces deux termes.

Classification et traitement de la classe « O » L'approche standard avec les réseaux prototypiques est de classer chaque mot en fonction de sa similarité avec les prototypes. Dans notre modèle, en l'absence de prototype pour la classe nulle, nous devons nous fier à un seuil en dessous duquel le mot est considéré comme un non-déclencheur. Typiquement, dans des travaux tels que Tan *et al.* (2019) et Nimah *et al.* (2021), un seuil global est défini en utilisant la distribution des valeurs de similarité sur un ensemble de validation. Cependant, dans notre cas, nous avons observé empiriquement que les distributions des valeurs de similarité entre un déclencheur et les prototypes varient trop d'une phrase à l'autre (cf. Figure 2a), ce qui rend impraticable l'utilisation d'un seuil global.

Pour résoudre ce problème, nous proposons de rechercher la probabilité correspondant au seuil optimal en utilisant la fonction de répartition sur les valeurs maximales de similarité. Ceci nous permet d'obtenir un seuil dynamique spécifique à la phrase considérée. Plus précisément, étant donné que les similarités des déclencheurs sont plus élevées que celles des mots « O » , nous supposons que, pour une phrase donnée, les similarités des déclencheurs ne seront présentes qu'au-dessus d'un certain quantile (assez élevé) dans la distribution des similarités. Nous supposons également que ce quantile est assez stable, même s'il ne correspond pas à la même valeur de similarité d'une phrase à l'autre. En pratique, pour une phrase donnée du *query set*, nous sélectionnons la phrase la plus similaire dans le *support set*. Puis, nous faisons varier le seuil entre les similarités minimum et maximum et adoptons celui maximisant la f1-mesure sur la phrase sélectionnée. Ensuite, nous déterminons la probabilité correspondant à ce seuil en utilisant la fonction de répartition sur les valeurs de similarité. Enfin, nous déterminons le seuil optimal pour la phrase du *query set* à partir de sa fonction de répartition et de la probabilité déterminée précédemment. Toutefois, comme les probabilités directement calculées à partir de la fonction de répartition dépendent du nombre de mots dans les phrases, nous interpolons linéairement la fonction de répartition sur un plus grand nombre de points avant d'estimer les probabilités, ce qui nous permet de donner artificiellement à toutes les phrases la même longueur (nous utilisons 512 points). Dans l'exemple de la Figure 2b, la « phrase 1 » (du *support set*) a son seuil optimal à $0{,}71$ correspondant à une probabilité de $0{,}97$. Nous reportons ensuite cette probabilité sur la fonction de répartition de la « phrase 2 » (du *query set*) pour obtenir son seuil optimal, égal à $0{,}92$.



(a) Similarités : min déclencheurs VS max « O »



(b) Fonctions de répartition de deux phrases

FIGURE 2 – La figure 2a montre que les mots déclencheurs et les mots « O » ne sont pas séparables avec un seuil global parce que la distribution des valeurs minimales des similarités des déclencheurs (Min déclencheurs) et la distribution des valeurs maximales des similarités des « O » (Max « O ») se chevauchent de façon importante. La Figure 2b montre, à partir de l'ECDF de deux exemples de phrases, que la similarité optimale varie d'une phrase à l'autre.

3 Expériences

3.1 Méthode d'évaluation et hyperparamètres

Nous expérimentons sur les ensembles de données ACE 2005 (Walker *et al.*, 2006), MAVEN (Wang *et al.*, 2020) et FewEvent (Deng *et al.*, 2020). Nous utilisons les découpages de Chen *et al.* (2021) pour ACE 2005 et MAVEN et celui de Cong *et al.* (2021) pour FewEvent. Dans tous les cas, les ensembles de test et d'apprentissage contiennent des classes distinctes, de sorte que lors de l'évaluation, le modèle doit faire face à de nouvelles classes qu'il n'a jamais vues auparavant.

Nous adoptons l'évaluation épisodique N ways, k shots, qui consiste à construire des épisodes avec N classes et k exemples annotés par classe. Dans l'évaluation épisodique standard (Vinyals *et al.*, 2016), les ensembles de test sont échantillonnés de façon à ce que toutes les classes soient distribuées uniformément, ce qui ne correspond pas à la distribution des mentions d'événements dans les données réelles. Ainsi, les scores de performance rapportés par cette méthode ne reflètent pas la distribution réelle des données. Nous adoptons la configuration plus réaliste de Yang & Katiyar (2020), qui construit le *support set* avec $N * K$ exemples et évalue le modèle sur tous les autres exemples.

Pour les expériences, nous avons utilisé le modèle pré-entraîné BERT-base comme encodeur et adopté la stratégie *Weighted* proposée par Tuo *et al.* (2022) pour obtenir des représentations contextuelles des mots. Nous adoptons une longueur maximale de séquence de 128 tokens, un taux d'apprentissage de 1e-5 et 30 000 épisodes N-ways, k-shots pour entraîner le modèle. Les hyper-paramètres $\mathcal{M}_0 = 1$ et $\mathcal{M}_1 = 0{,}4$ ont été obtenus sur l'ensemble de validation, pris entre $0{,}2$ et 1 (avec un pas de $0{,}2$).

3.2 Résultats et analyses

Nous comparons notre approche, **OUTFIT** (i.e. OUT oF trIgger deTection), à quatre autres modèles de l'état de l'art dans la configuration 5-ways 5-shots. **PA-CRF** (Cong *et al.*, 2021) et Tuo *et al.* (2022) sont des modèles de l'état de l'art qui calculent un prototype pour la classe « O » . **PA-CRF** estime les probabilités de transition entre les étiquettes BIO avec l'utilisation de CRF (*Conditional Random Fields)* (Lafferty *et al.*, 2001) tandis que Tuo *et al.* (2022) propose une meilleure exploitation

	Modèle	ACE 2005	MAVEN	FewEvent
5-ways, 5-shots	PROTO	49,2 ± 1,2	51,6 ± 1,4	53,6 ± 0,7
	PA-CRF (Cong *et al.*, 2021)	64,0 ± 0,6	65,2 ± 0,3	65,3 ± 2,0
	(Tuo *et al.*, 2022)	66,4 ± 1,8	67,1 ± 1,5	67,4 ± 1,1
	HCL-TAT† (Zhang *et al.*, 2022)	–	–	66,9 ± 0,7
	OUTFIT (ours)	**74,0*** ± **1,1**	76,9 ± 1,1	**79,6*** ± **4,2**
	− PoS tags	72,2 ± 2,2	**77,5*** ± **0,8**	77,9 ± 3,9
	− contrastive	66,5 ± 5,7	63,1 ± 12,6	75,9 ± 5,4
	− weighted	59,2 ± 3,6	50,0 ± 2,3	70,9 ± 2,7
	Seuil oracle	82,5 ± 1,9	87,2 ± 1,1	84,1 ± 0,5
1w,5s	FS-Causal† (Chen *et al.*, 2021)	76,9 ± 1,4	55,0 ± 0,4	–
	OUTFIT	80,9 ± 2,9	81,1 ± 1,1	79,1 ± 2,1

TABLE 1 – Performance de détection d'événement sur trois jeux de données, en moyenne et écart-type de la micro f1-mesure sur 5 essais. † indique les résultats issus de l'article original. * indique que la différence entre le meilleur modèle (**en gras**) et le deuxième (souligné) est statistiquement significative, en utilisant le test de significativité de (Dror *et al.*, 2019).

des couches du modèle BERT afin d'obtenir plus d'information pour l'apprentissage. **HCL-TAT** (Zhang *et al.*, 2022) est également un modèle sans prototype pour la classe nulle utilisant un seuil de décision égal à la moyenne des similarités pendant un épisode. Nous comparons ces méthodes à un modèle prototypique de base qui construit un prototype pour la classe nulle, utilise l'entropie croisée comme fonction de coût et un encodeur BERT-base (**PROTO**). **FS-Causal** (Chen *et al.*, 2021) est un modèle ajoutant une prise en compte explicite des relations de causalité entre les déclencheurs et leur contexte pour résoudre le problème dit de la malédiction des déclencheurs (*trigger curse*). En effet, un surapprentissage des déclencheurs peut nuire à la détection des déclencheurs rares dans l'ensemble de données. La prise en compte du contexte seul (sans les déclencheurs) permet de résoudre ce problème. Comme leurs résultats rapportés ne sont évalués que classe par classe, cela correspond à une configuration 1-way 5-shots. Nous avons également ajouté les résultats correspondant au seuil optimal trouvé directement sur les instances du *query set* (**Seuil oracle**), ce qui donne une indication du meilleur résultat pouvant être obtenu avec notre approche.

Dans les expériences préliminaires, nous avons remarqué que la précision ($\approx 65\%$) était relativement faible par rapport au rappel ($\approx 80\%$), ce qui indique que le modèle identifiait trop de mots comme déclencheurs. Pour augmenter la précision, nous avons filtré les prédictions en fonction de leurs catégories morphosyntaxiques (*PoS tags*), en ne conservant que les étiquettes les plus couramment associées aux déclencheurs d'événements dans l'ensemble d'apprentissage (verbe, adverbe et nom) [3].

Notre méthode établit une nouvelle performance de l'état de l'art avec une augmentation moyenne de 10 points de la f1-mesure pour les trois jeux de données considérés (cf. Tableau 1). Les analyses suggèrent que l'encodeur *Weighted* et l'apprentissage contrastif, combinés à notre nouvelle formulation, jouent un rôle important dans la performance globale du modèle. Plus spécifiquement, nous pouvons noter que la fonction contrastive contribue fortement à diminuer la variance des résultats. Nous pensons également que cette fonction, combinée à notre stratégie de recherche de seuil, contribue à la forte différence de performance avec HCL-TAT alors que nos problématiques sont initialement proches. Comme nos expériences préliminaires l'ont suggéré, le filtrage des déclencheurs candidats en fonction de leurs catégories morphosyntaxiques permet d'augmenter les performances de quelques points pour deux jeux de données. Toutefois, la condition sans ce filtrage, qui est la plus

3. Les déclencheurs événementiels sont essentiellement nominaux et verbaux mais ils peuvent inclure des mots qui ne sont ni des noms, ni des verbes, le cas le plus fréquent en anglais étant celui des verbes à particule.

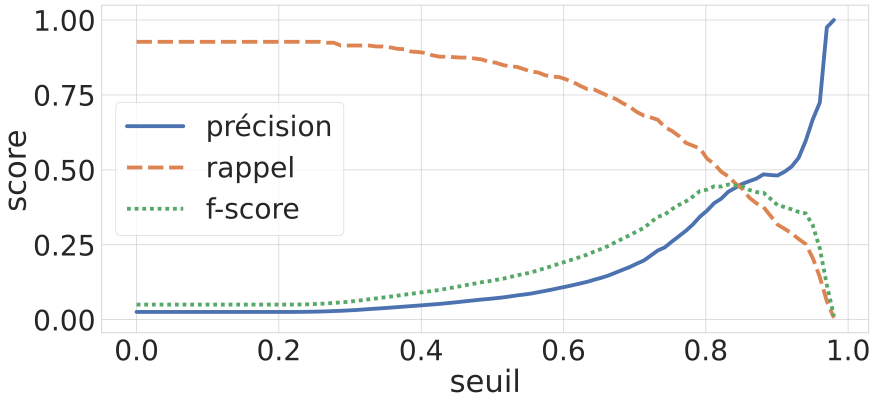


FIGURE 3 – Scores en utilisant un seuil global sur le jeu de données FewEvent

très directement comparable aux modèles de l'état de l'art, montre que celui-ci n'est pas le facteur principal des améliorations obtenues.

Dans la configuration 1-way 5-shots, notre modèle améliore également les performances par rapport à FS-Causal pour les deux jeux de données avec des résultats pour FS-Causal. Ce résultat montre d'abord que l'amélioration apportée par notre proposition n'est pas limitée à un unique cadre d'évaluation. Par ailleurs, considérer de nouveaux types d'événements un par un est la stratégie la plus générale pour l'adaptation à un nouveau domaine dans lequel le nombre de types d'événements n'est pas connu à l'avance. Cependant, l'écart important entre l'oracle et notre modèle suggère que notre approche pourrait être encore améliorée.

Enfin, la Figure 3 montre les scores pour un seuil global allant de 0 à 1 sur le jeu de données FewEvent, pour la condition 5-ways 5-shots. Nous remarquons que la meilleure f1-mesure pouvant être obtenue avec un seuil global est d'environ 0,45, ce qui justifie clairement l'intérêt d'adopter un seuil dynamique plutôt qu'un seuil global comme dans (Tan *et al.*, 2019) ou (Nimah *et al.*, 2021).

4 Conclusion et perspectives

Dans cet article, nous abordons la détection d'événement à partir de peu d'exemples comme une tâche de détection hors domaine en utilisant des réseaux prototypiques. Cette méthode évite de construire un prototype pour la classe nulle, qui est par nature hétérogène, et fournit un seuil dynamique pour décider si un mot est un déclencheur ou non. Les résultats expérimentaux suggèrent que cette nouvelle formulation offre une amélioration importante des performances par rapport aux autres méthodes de l'état de l'art. À notre connaissance, il s'agit du premier effort de recherche qui présente la FSED comme une tâche d'annotation de séquence tout en traitant la classe nulle comme un problème de détection hors domaine. Nous pensons que notre méthode pourrait être appliquée à d'autres tâches d'annotation de séquences et nous étudierons plus particulièrement son application à l'extraction d'arguments d'événements et à la reconnaissance d'entités nommées dans le cadre de travaux futurs.

Remerciements

Ces travaux ont été réalisés grâce au supercalculateur Factory-IA financé par le Conseil Régional d'Île-de-France.

Références

BRONSTEIN O., DAGAN I., LI Q., JI H. & FRANK A. (2015). Seed-Based Event Trigger Labeling : How far can event descriptions get us ? In *ACL-IJCNLP*, p. 372–376. DOI : 10.3115/v1/P15-2061.

CHEN J., LIN H., HAN X. & SUN L. (2021). Honey or Poison ? Solving the Trigger Curse in Few-shot Event Detection via Causal Intervention. In *Proceedings of EMNLP*, p. 8078–8088, Online and Punta Cana, Dominican Republic : Association for Computational Linguistics. DOI : 10.18653/v1/2021.emnlp-main.637.

CONG X., CUI S., YU B., LIU T., YUBIN W. & WANG B. (2021). Few-Shot Event Detection with Prototypical Amortized Conditional Random Field. In *Findings of ACL-IJCNLP*, p. 28–40, Online. DOI : 10.18653/v1/2021.findings-acl.3.

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 *WSDM*, p. 151–159, Houston, TX, USA. DOI : 10.1145/3336191.3371796.

DROR R., SHLOMOV S. & REICHART R. (2019). Deep Dominance - How to Properly Compare Deep Neural Models. In *ACL*, p. 2773–2785, Florence, Italy. DOI : 10.18653/v1/P19-1266.

GENG R., LI B., LI Y., ZHU X., JIAN P. & SUN J. (2019). Induction Networks for Few-Shot Text Classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, p. 3904–3913, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-1403.

LAFFERTY J. D., MCCALLUM A. & PEREIRA F. C. N. (2001). Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data. In *ICML*, p. 282–289, San Francisco, CA, USA.

LAI V., DERNONCOURT F. & NGUYEN T. H. (2021). Learning Prototype Representations Across Few-Shot Tasks for Event Detection. In *EMNLP*, p. 5270–5277.

LAI V. D. & NGUYEN T. (2019). Extending Event Detection to New Types with Learning from Keywords. In *W-NUT 2019*, p. 243–248, Hong Kong, China. DOI : 10.18653/v1/D19-5532.

LAI V. D., NGUYEN T. H. & DERNONCOURT F. (2020). Extensively Matching for Few-shot Learning Event Detection. In *Workshop NUSE*, p. 38–45, Online. DOI : 10.18653/v1/2020.nuse-1.5.

LI Q., JI H. & HUANG L. (2013). Joint Event Extraction via Structured Prediction with Global Features. In *ACL*, p. 73–82, Sofia, Bulgaria.

LIAO S. & GRISHMAN R. (2011). Acquiring topic features to improve event extraction : in pre-selected and balanced collections. In *RANLP* : Association for Computational Linguistics.

LIU X., LUO Z. & HUANG H. (2018). Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation. In *EMNLP*, p. 1247–1256. DOI : 10.18653/v1/D18-1156.

NGUYEN T. H., CHO K. & GRISHMAN R. (2016). Joint Event Extraction via Recurrent Neural Networks. In *NAACL-HLT*, p. 300–309, San Diego, California. DOI : 10.18653/v1/N16-1034.

NGUYEN T. H. & GRISHMAN R. (2015a). Event Detection and Domain Adaptation with Convolutional Neural Networks. In *ACL-IJCNLP*, p. 365–371, Beijing, China. DOI : 10.3115/v1/P15-2060.

NGUYEN T. H. & GRISHMAN R. (2015b). Event Detection and Domain Adaptation with Convolutional Neural Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 365–371, Beijing, China : Association for Computational Linguistics. DOI : 10.3115/v1/P15-2060.

NGUYEN T. H. & GRISHMAN R. (2018). Graph Convolutional Networks with Argument-Aware Pooling for Event Detection. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

NIMAH I., FANG M., MENKOVSKI V. & PECHENIZKIY M. (2021). ProtoInfoMax : Prototypical Networks with Mutual Information Maximization for Out-of-Domain Detection. In *Findings of the Association for Computational Linguistics : EMNLP*, p. 1606–1617 : Association for Computational Linguistics. DOI : 10.18653/v1/2021.findings-emnlp.138.

SCHÖLKOPF B., PLATT J. C., SHAWE-TAYLOR J., SMOLA A. J. & WILLIAMSON R. C. (2001). Estimating the Support of a High-Dimensional Distribution. *Neural Computation*, (7), 1443–1471. DOI : 10.1162/089976601750264965.

SHEN S., WU T., QI G., LI Y.-F., HAFFARI G. & BI S. (2021). Adaptive Knowledge-Enhanced Bayesian Meta-Learning for Few-shot Event Detection. In *Findings of ACL-IJCNLP*, p. 2417–2429, Online. DOI : 10.18653/v1/2021.findings-acl.214.

SNELL J., SWERSKY K. & ZEMEL R. (2017). Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, volume 30.

SUNG F., YANG Y., ZHANG L., XIANG T., TORR P. H. S. & HOSPEDALES T. M. (2018). Learning to compare : Relation network for few-shot learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, p. 1199–1208.

TAN M., YU Y., WANG H., WANG D., POTDAR S., CHANG S. & YU M. (2019). Out-of-Domain Detection for Low-Resource Text Classification Tasks. In *EMNLP-IJCNLP*, p. 3566–3572 : Association for Computational Linguistics. DOI : 10.18653/v1/D19-1364.

TUO A., BESANÇON R., FERRET O. & TOURILLE J. (2022). Better Exploiting BERT for Few-Shot Event Detection. In *NLDB*, p. 291–298, Berlin, Heidelberg : Springer-Verlag. DOI : 10.1007/978-3-031-08473-7_26.

TUO A., BESANÇON R., FERRET O. & TOURILLE J. (2023). Trigger or not trigger : Dynamic thresholding for few shot event detection. In J. KAMPS, L. GOEURIOT, F. CRESTANI, M. MAISTRO, H. JOHO, B. DAVIS, C. GURRIN, U. KRUSCHWITZ & A. CAPUTO, Éds., *45th European Conference on Information Retrieval (ECIR 2023) : Advances in Information Retrieval, short article session*, volume 13981 de *Lecture Notes in Computer Science*, p. 637–645, Dublin, Ireland : Springer Nature Switzerland.

VINYALS O., BLUNDELL C., LILLICRAP T., KAVUKCUOGLU K. & WIERSTRA D. (2016). Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, volume 29.

WALKER C., STRASSEL S. & JULIE MEDERO K. M. (2006). ACE 2005 Multilingual Training Corpus. DOI : 10.35111/mwxc-vh88.

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)*, p. 1652–1671, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.emnlp-main.129.

YAN H., JIN X., MENG X., GUO J. & CHENG X. (2019). Event detection with multi-order graph convolution and aggregated attention. In *EMNLP-IJCNLP*, p. 5766–5770.

YANG Y. & KATIYAR A. (2020). Simple and effective few-shot named entity recognition with structured nearest neighbor learning. In *EMNLP*, p. 6365–6375 : Association for Computational Linguistics. DOI : 10.18653/v1/2020.emnlp-main.516.

ZHANG R., WEI W., MAO X.-L., FANG R. & CHEN D. (2022). HCL-TAT : A Hybrid Contrastive Learning Method for Few-shot Event Detection with Task-Adaptive Threshold. In *Findings of the Association for Computational Linguistics : EMNLP*, p. 1808–1819 : Association for Computational Linguistics.