Actes de la 28e Conférence sur le Traitement Automatique des Langues Naturelles
Lille, France, 28 juin au 2 juillet 2021

Intérêt des modèles de caractères pour la détection d'événements

Emanuela Boros[1] Romaric Besançon[2] Olivier Ferret[2] Brigitte Grau[3]
(1) La Rochelle Université, L3i, F-17042 La Rochelle
(2) Université Paris-Saclay, CEA, List, F-91120, Palaiseau, France
(3) Université Paris-Saclay, CNRS, LIMSI, ENSIIE, F-91405, Orsay, France
emanuela.boros@univ-lr.fr, romaric.besancon,olivier.ferret@cea.fr,
brigitte.grau@limsi.fr

RÉSUMÉ

Cet article aborde la tâche de détection d'événements, visant à identifier et catégoriser les mentions d'événements dans les textes. Une des difficultés de cette tâche est le problème des mentions d'événements correspondant à des mots mal orthographiés, très spécifiques ou hors vocabulaire. Pour analyser l'impact de leur prise en compte par le biais de modèles de caractères, nous proposons d'intégrer des plongements de caractères, qui peuvent capturer des informations morphologiques et de forme sur les mots, à un modèle convolutif pour la détection d'événements. Plus précisément, nous évaluons deux stratégies pour réaliser une telle intégration et montrons qu'une approche de fusion tardive surpasse à la fois une approche de fusion précoce et des modèles intégrant des informations sur les caractères ou les sous-mots tels que ELMo ou BERT.

ABSTRACT

The interest of character-level models for event detection.

This paper tackles the task of event detection that aims at identifying and categorizing event mentions in texts. One of the difficulties of this task is the problem of event mentions corresponding to misspelled, custom, or out-of-vocabulary words. To analyze the impact of character-level features, we propose to integrate character embeddings, which can capture morphological and shape information about words, to a convolutional model for event detection. More precisely, we evaluate two strategies for performing such integration and show that a late fusion approach outperforms both an early fusion approach and models integrating character or subword information such as ELMo or BERT.

MOTS-CLÉS : Extraction d'information, événements, plongements lexicaux.

KEYWORDS: Information extraction, events, word embeddings.

1 Introduction

Dans cet article, nous nous concentrons plus particulièrement sur la détection d'événements, qui implique l'identification d'instances de types d'événements prédéfinis dans un texte. Ces instances, appelées mentions d'événements ou déclencheurs d'événements, prennent la forme de mots ou d'expressions polylexicales évoquant un type d'événements de façon plus ou moins spécifique. Les approches les plus efficaces pour réaliser cette tâche sont actuellement fondées sur des modèles neuronaux (Chen *et al.*, 2015; Nguyen & Grishman, 2015; Nguyen *et al.*, 2016a,b; Feng *et al.*,

	Tous les mots	Mentions d'événements
entraînement	14 021	931
test	3 553	219
mots inconnus dans le test	930 (26,2%)	66 (30,1%)
mots inconnus avec un mot similaire	825	54

TABLE 1 – Statistiques concernant le vocabulaire des parties entraînement et test du corpus ACE 2005. Mot inconnu : présent dans la partie entraînement mais pas dans la partie test

2016; Zhang *et al.*, 2019; Nguyen & Grishman, 2018) et ont permis en particulier de s'affranchir du problème du choix des traits linguistiques utilisés par les modèles d'apprentissage statistiques. Ces modèles reposent ainsi sur des plongements de mots qui les rendent en principe moins sensibles au problème des déclencheurs non rencontrés lors de l'entraînement puisque ces plongements intègrent une forme de similarité entre les mots.

Toutefois, cette capacité peut varier en fonction des raisons pour lesquelles un déclencheur n'a pas été vu lors de l'entraînement du modèle. Nous illustrons ces différents cas sur la partie anglaise du jeu de données ACE 2005, un corpus standard pour l'évaluation de la détection d'événements dont nous reprenons la subdivision classiquement faite pour cette tâche entre entraînement, validation et test (Ji *et al.*, 2008). Le déclencheur inédit peut ainsi être une variante morphologique d'un déclencheur déjà vu dans l'ensemble des données d'entraînement. Par exemple, *torturing* n'est pas présent dans les données d'entraînement ACE 2005 mais il s'agit d'une variante de *torture*, qui est considéré comme un déclencheur pour le même type d'événements, en l'occurrence *Life.Injury*. En outre, *torturing* est susceptible d'être présent au sein d'un modèle de langue général, auquel cas un modèle de détection d'événements neuronal reposant sur ledit modèle de langue est susceptible de détecter avec succès ce déclencheur.

La situation est différente lorsqu'un déclencheur est absent des données d'entraînement parce qu'il correspond à une version mal orthographiée d'un déclencheur de référence. En effet, dans un tel cas, le modèle de langue ne contient pas nécessairement la version altérée. Par exemple, *aquitted* fait partie du corpus de test ACE 2005 pour référer à un événement *Justice.Sentence* alors que seule *acquitted*, la forme correcte pour ce mot, est présente dans les données d'entraînement. Dans ce cas, il est peu probable que le mot inédit fasse partie du modèle de langue général et, par conséquent, il a peu de chances d'être détecté comme déclencheur d'un événement *Justice.Sentence*. Plus globalement, comme le montre le tableau 1, 30,1 % des déclencheurs du corpus de test ACE 2005 ne sont pas présents dans le corpus d'entraînement mais 88 % de ces déclencheurs absents sont proches (mesurés par un ratio de Levenshtein inférieur à 0,3) de mots du corpus d'entraînement. Le tableau 2 présente des exemples de telles paires de mots. On peut voir qu'en dehors des paires correspondant à des différences de casse (intifada/Intifada) ou relevant de la morphologie flexionnelle (opening/open), certaines paires correspondent à des cas plus complexes relevant de la morphologie dérivationnelle (creating/creation) ou même de relations sémantiques complexes (hacked/attacked) qui ne sont souvent pas capturées par les modèles de plongements de mots.

Différentes stratégies ont été proposées pour traiter le problème de la variabilité lexicale dans les modèles de langue neuronaux. Pour les plongements statiques de mots, fastText (Bojanowski *et al.*, 2017) s'appuie ainsi sur une représentation des mots fondée sur des n-grammes de caractères. Pour les modèles contextuels, ELMo (Peters *et al.*, 2018) exploite une représentation fondée sur les caractères

Type d'événements	Déclencheur inconnu/connu le plus proche
Start-Org	*creating/creation, opening/open, forging/forming, formed/form*
End-Org	*crumbled/crumbling, dismantling/dismantle, dissolved/dissolving*
Transport	*fleeing/flying, deployment/deployed, evacuating/evacuated*
Attack	*intifada/Intifada, smash/smashed, hacked/attacked, wiped/wipe*
End-Position	*retirement/retire, steps/step, previously/previous, formerly/former*

TABLE 2 – Exemples de déclencheurs événementiels de test proches de déclencheurs d'entraînement

construite grâce à un réseau de neurones convolutif (CNN) tandis que BERT (Devlin *et al.*, 2019) adopte une stratégie mixte fondée sur des sous-mots, appelés wordpieces (Luong & Manning, 2016; Kim *et al.*, 2016; Jozefowicz *et al.*, 2016), avec quelques limites sur sa capacité à gérer les entrées bruitées (Sun *et al.*, 2020).

Nos contributions dans cet article sont plus particulièrement axées sur l'intégration de modèles reposant sur le niveau des caractères dans les modèles de détection d'événements pour traiter la question des mots inconnus. Plus précisément, nous montrons qu'un modèle de détection d'événements exploitant une représentation fondée sur les caractères est complémentaire d'un modèle fondé sur les mots et que leur combinaison selon une approche de fusion tardive est plus performante qu'une stratégie de fusion précoce.

2 Modèles

Notre approche s'inscrit dans le droit fil de la plupart des modèles de détection supervisée d'événements en considérant cette tâche comme une forme de classification multiclasse de mots : étant donné une phrase et un ensemble de types d'événements possibles, l'objectif est de prédire pour chacun de ses mots s'il relève ou non d'un de ces types d'événements et le cas échéant, duquel. L'entrée du système est donc un mot cible dans le contexte d'une phrase et sa sortie, un type d'événements ou l'étiquette NONE pour les mots non déclencheurs. Pour étudier l'influence des traits fondés sur les caractères, nous nous appuyons sur le modèle CNN proposé par Nguyen & Grishman (2015). Ce modèle de base est utilisé dans les deux composantes de notre modèle global : le modèle fondé sur les mots, dit modèle CNN mot, et le modèle fondé sur les caractères, dit modèle CNN caractère. Ces deux composantes sont combinées en utilisant soit une approche de fusion précoce, soit une approche de fusion tardive, comme l'illustre la figure 1.

Dans le modèle CNN mot, le contexte d'un mot candidat en tant que mention événementielle est formé par les mots qui l'entourent dans la phrase. Pour tenir compte de la nécessité de gérer des entrées de même dimension, ce contexte prend la forme d'une fenêtre de taille fixe, centrée sur la mention candidate. De ce fait, les parties de phrases dépassant la limite de cette fenêtre sont tronquées tandis qu'un remplissage avec des valeurs nulles (*zero-padding*) est réalisé pour les phrases plus courtes. Au sein de cette fenêtre de contexte, chaque mot est représenté par un plongement de mot et une position relative par rapport à la mention candidate, elle aussi sous la forme d'un plongement. Les plongements de mots et de positions sont concaténés et passés au travers d'une couche de convolution. Plus précisément, un ensemble de filtres convolutifs de tailles différentes sont appliqués et une opération de *max pooling* est appliquée à l'échelle de la fenêtre pour obtenir une

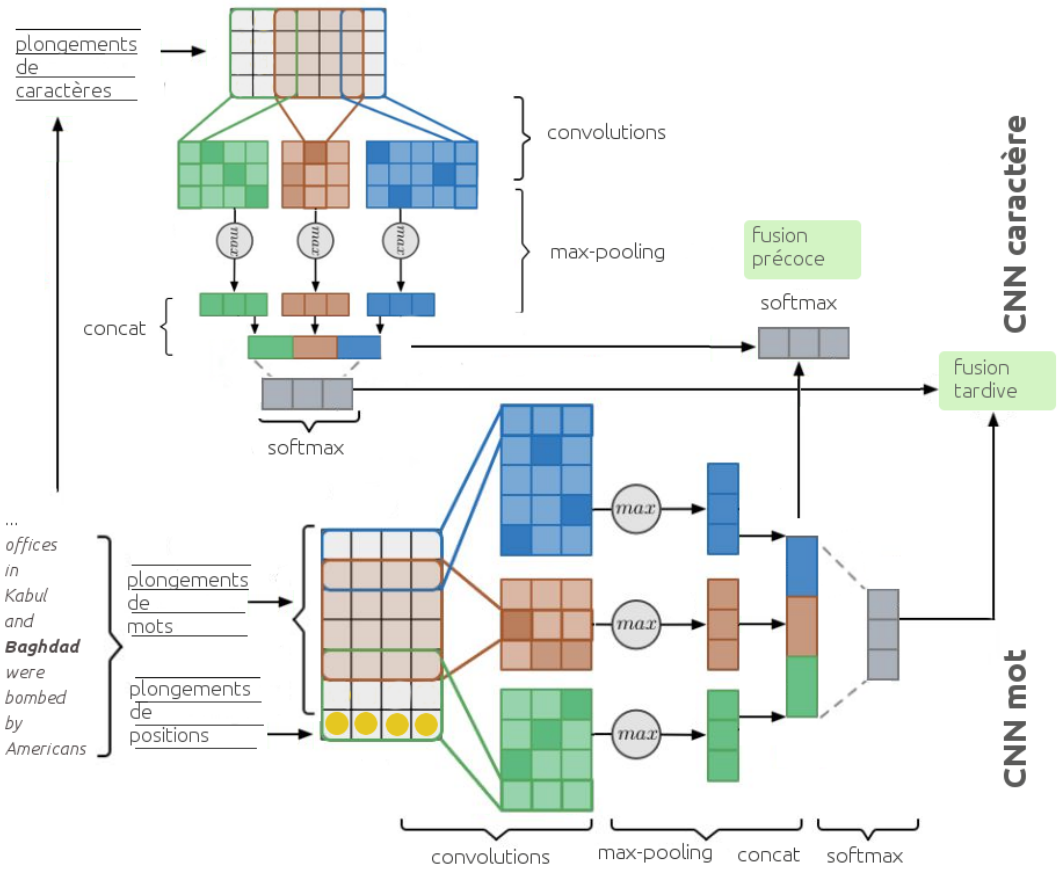

FIGURE 1 – Association d'un modèle fondé sur les mots et d'un modèle fondé sur les caractères

valeur par filtre. Le résultat de ces opérations se voit ensuite appliquer un *softmax* pour réaliser la classification en tant que telle. Le modèle CNN caractère est très proche du modèle CNN mot, avec deux différences principales : les mots sont remplacés par des caractères et il n'y a pas d'information de position associée à chaque caractère. Plus précisément, chaque mention candidate, identifiée sur la base des mots, se voit associer une fenêtre de contexte, comme dans le cas du CNN mot, mais cette fenêtre est dans ce cas déterminée sur la base d'un nombre fixe de caractères et les éléments de base de représentation sont constitués par des plongements de caractères. Les mêmes mécanismes de troncation et de remplissage permettant de considérer des phrases de taille variable et une fenêtre de contexte de taille fixe sont appliqués, mais ici à l'échelle du caractère.

Le premier type d'intégration de ces deux modèles est une fusion précoce, dans laquelle les deux représentations de la séquence d'entrée produites par les CNN de mots et de caractères sont concaténées avant la couche de classification. L'utilisation de ce type d'intégration permet un apprentissage conjoint des paramètres des deux modèles lors de la phase d'entraînement. L'intégration par fusion tardive repose quant à elle sur la combinaison par vote des décisions des deux modèles, qui sont entraînés séparément et apprennent donc des caractéristiques différentes des mentions candidates. La méthode de vote se définit comme suit : si une mention événementielle est détectée par un seul des deux modèles, nous conservons l'étiquette donnée par ce modèle ; sinon, si une mention est détectée par le CNN mot et le CNN caractère ensemble, nous conservons l'étiquette donnée par le CNN caractère. Cette stratégie est motivée par le fait que le modèle CNN mot possède une bonne couverture tandis que le modèle CNN caractère est davantage axé sur la précision.

3 Expérimentations, résultats et discussion

Cadre expérimental Nos expérimentations ont été réalisées sur le corpus ACE 2005. À des fins de comparabilité, nous utilisons le même découpage que les travaux antérieurs (Ji *et al.*, 2008; Liao & Grishman, 2010; Li *et al.*, 2013; Nguyen & Grishman, 2015; Nguyen *et al.*, 2016a), avec 529 documents (14 849 phrases) pour l'entraînement, 30 documents (863 phrases) pour le développement et 40 documents (672 phrases) pour le test. De même, nous considérons qu'une mention d'événement est correcte si son type d'événement, son sous-type et son empan correspondent à ceux d'une mention de référence. Nous utilisons les micro-mesures de précision, rappel et F1-mesure (F1) pour évaluer la performance globale.

Paramètres des modèles Pour le CNN mot, la taille de la fenêtre de contexte est de 31 mots. Les filtres de convolution ont pour leur part une dimension de 1, 2 et 3 mots et 300 filtres sont utilisés pour chaque dimension. Après chaque couche convolutive, initialisée selon un schéma orthogonal (Saxe *et al.*, 2014), une couche non linéaire *ReLU* est appliquée. Nous employons un abandon (*dropout*) de probabilité 0,5 après la couche initiale des plongements et de probabilité 0,3 après la concaténation du résultat des convolutions. La dimensionnalité des plongements de positions est de 50, à l'instar de (Nguyen & Grishman, 2015). Enfin, nous avons utilisé les plongements de mots préentraînés construits avec Word2vec sur le corpus Google News (Mikolov *et al.*, 2013).

Pour le CNN caractère, l'entrée est constituée de séquences de 1 024 caractères. Nous considérons tous les caractères sauf l'espace. La taille des filtres de convolution va de 2 à 10, avec 300 filtres par taille. La non-linéarité et l'initialisation de la couche convolutive sont les mêmes que pour le CNN mot. Les plongements de caractères comportent 300 dimensions et sont initialisés sur la base d'une distribution normale. Un abandon de 0,5 est réalisé après les plongements de caractères. Lors de l'entraînement conjoint dans le modèle de fusion précoce, les vecteurs de traits obtenus après les convolutions des deux modèles sont concaténés et comme pour le CNN mot, un abandon de 0,3 est appliqué avant la couche softmax.

Résultats et discussion Nous comparons notre modèle avec plusieurs modèles neuronaux proposés pour la même tâche n'utilisant pas de ressources externes : des modèles convolutifs (Nguyen & Grishman, 2015; Chen *et al.*, 2015; Nguyen *et al.*, 2016b; Nguyen & Grishman, 2018), des modèles récurrents (Nguyen *et al.*, 2016a; Zhao *et al.*, 2018), des modèles hybrides (Feng *et al.*, 2016), le modèle GAIL-ELMo (Zhang *et al.*, 2019) et un modèle fondé sur un mécanisme d'attention multilingue (Liu *et al.*, 2018). Nous ne considérons pas pour des raisons de comparabilité les modèles utilisant des ressources externes tels que (Bronstein *et al.*, 2015; Li *et al.*, 2019) ou (Yang *et al.*, 2019). Nous nous comparons également aux modèles plus récents fondés sur BERT tels que le modèle de (Wadden *et al.*, 2019) conjuguant BERT et un LSTM pour capturer un contexte intra et inter-phrastique et définir de façon plus dynamique les mentions candidates, le modèle BERT-QA (Du & Cardie, 2020), qui aborde la détection d'événements comme une tâche de question-réponse et le modèle DMBERT (Wang *et al.*, 2019), qui s'appuie sur l'apprentissage adverse pour mettre en œuvre une approche faiblement supervisée. Nous comparons également notre modèle avec 4 approches de base reposant sur BERT, en abordant la détection d'événements de manière similaire à la reconnaissance d'entités nommées dans (Devlin *et al.*, 2019) et avec les mêmes valeurs d'hyperparamètres.

La meilleure performance (F1 = 75,8 %) est obtenue en combinant les plongements de mots et de positions avec les plongements de caractères selon une stratégie de fusion tardive. Le tableau 3 montre également que l'ajout de plongements de caractères dans une stratégie de fusion tardive est plus performant que tous les modèles s'appuyant sur les mots, y compris les architectures complexes

Approches	**Précision**	**Rappel**	**F1**
Word CNN (Nguyen & Grishman, 2015)	71,8	66,4	69,0
Dynamic multi-pooling CNN (Chen *et al.*, 2015)	75,6	63,6	69,1
Joint RNN (Nguyen *et al.*, 2016a)	66,0	73,0	69,3
CNN with document context (Duan *et al.*, 2017)†	77,2	64,9	70,5
Non-Consecutive CNN (Nguyen *et al.*, 2016b)	na	na	71,3
Attention-based (Liu *et al.*, 2017)+	78,0	66,3	71,7
GAIL-ELMo (Zhang *et al.*, 2019)	74,8	69,4	72,0
Gated Cross-Lingual Attention (Liu *et al.*, 2018)	78,9	66,9	72,4
Graph CNN (Nguyen & Grishman, 2018)	77,9	68,8	73,1
Hybrid NN (Feng *et al.*, 2016)	84,6	64,9	73,4
DEEB-RNN3 (Zhao *et al.*, 2018)	72,3	75,8	74,0
BERT-base-uncased + LSTM (Wadden *et al.*, 2019)	na	na	68,9
BERT-base-uncased (Wadden *et al.*, 2019)	na	na	69,7
BERT-base-uncased (Du & Cardie, 2020)	67,2	73,2	70,0
BERT-QA (Du & Cardie, 2020)	71,1	73,7	72,4
DMBERT (Wang *et al.*, 2019)	77,6	71,8	74,6
DMBERT+Boot (Wang *et al.*, 2019)	77,9	72,5	75,1
BERT-base-uncased	71,7	68,5	70,0
BERT-base-cased	71,3	72,0	71,7
BERT-large-uncased	72,1	72,9	72,5
BERT-large-cased	69,3	**77,2**	73,1
CNN mot (équivalent à Word CNN)	71,4	65,9	68,5
CNN caractère	71,7	41,2	52,3
CNN mot + caractère - fusion précoce	**88,6**	61,9	72,9
CNN mot + caractère - fusion tardive	87,2	67,1	**75,8**

TABLE 3 – Évaluation de nos modèles et comparaison avec l'état de l'art pour la détection d'événements sur le test d'ACE 2005. †au-delà de la phrase, +avec les arguments de référence

s'appuyant sur les convolutions de graphe et les modèles exploitant BERT. Parmi ceux-ci, il est intéressant de noter que les modèles intégrant la casse (*cased*) sont plus performants que les modèles *uncased*, ce qui confirme l'importance de l'information portée par le niveau des caractères pour cette tâche, peut-être parce que la capitalisation est liée à la reconnaissance des entités nommées, qui sont généralement considérées comme importantes pour la détection des mentions d'événements. La similitude de nos résultats pour *BERT-base-uncased* avec ceux de (Du & Cardie, 2020) et (Wadden *et al.*, 2019) pour le même BERT accrédite par ailleurs la solidité de ce constat.

Cependant, nous pouvons constater que les plongements de caractères ne sont pas suffisants en eux-mêmes : en utilisant uniquement le CNN caractère, nous obtenons ainsi le plus petit rappel de toutes les approches considérées. Néanmoins, sa précision (71,7) est comparativement très élevée, ce qui confère une bonne fiabilité aux mentions qu'il détecte. Dans le cas de la fusion précoce, nous constatons que la précision est la plus élevée de tous les modèles comparés. Nous supposons que dans l'approche conjointe, l'influence des représentations fondées sur les caractères dépasse celle des plongements de mots et de positions et que la combinaison reproduit le déséquilibre entre la

Type d'événements	Nouvelles mentions trouvées	Mentions d'entraînement
End-Position	*steps*	*step*
Extradite	*extradited*	*extradition*
Attack	*wiped*	*wipe*
Start-Org	*creating*	*create*
Attack	*smash*	*smashed*
End-Position	*retirement*	*retire*

TABLE 4 – Nouvelles mentions trouvées grâce au modèle CNN mot+caractère (fusion tardive)

précision et le rappel observé pour le CNN caractère, le rappel étant le plus faible de tous les modèles à l'exception du CNN caractère. La fusion tardive permet un contrôle plus informé de la combinaison et, en donnant la priorité au CNN caractère pour déterminer le type des mentions identifiées par le CNN mot, la méthode tire profit de sa grande précision, permettant une augmentation de la précision de 71,7 à 87,2 tout en ayant un rappel élevé, passant de 65,9 pour le CNN mot à 67,1.

Finalement, nous avons mené une analyse plus qualitative en examinant les mentions d'événements nouvellement détectées par le modèle à fusion tardive comparativement au modèle à fusion précoce. Nous avons observé que parmi les 37 mentions concernées, certaines sont effectivement des variantes dérivationnelles ou flexionnelles de mots présents dans les données d'entraînement, comme illustré par le tableau 4. Ce constat semble confirmer que le modèle fondé sur les caractères peut capturer certaines informations sémantiques associées aux caractéristiques morphologiques des mots et parvenir ainsi à détecter de nouvelles mentions d'événements en relation avec des mentions d'entraînement. La présence dans le CNN caractère de filtres convolutifs d'une taille entre 2 et 10, c'est-à-dire couvrant une plage assez large de n-grammes de caractères, contribue très certainement à cette capacité.

4 Conclusion et perspectives

Dans cet article, nous avons étudié l'intégration de plongements de caractères dans un modèle neuronal de détection d'événements fondé un simple modèle CNN en testant des stratégies de fusion précoce ou tardive. Les meilleurs résultats sont obtenus en combinant les représentations fondées sur les mots avec celles fondées sur les caractères dans une stratégie de fusion tardive donnant la priorité au modèle de caractères pour décider du type d'événements. Cette méthode est plus performante que des approches plus complexes fondées sur les convolutions de graphe, les réseaux antagonistes ou les modèles BERT. Ces résultats montrent aussi qu'un modèle de caractères permet de surmonter certains problèmes concernant les mots nouveaux ou mal orthographiés dans les données de test.

Ce travail ouvre la voie à des études plus larges sur le problème de la robustesse des modèles de détection d'événements vis-à-vis des variations touchant les déclencheurs événementiels. De ce point de vue, il serait intéressant de tester si des modèles de langue de type Transformer s'appuyant sur les caractères (El Boukkouri *et al.*, 2020; Ma *et al.*, 2020), ou même s'affranchissant de la segmentation en mots (Clark *et al.*, 2021), pourraient s'avérer plus robustes qu'un modèle de type BERT.

Remerciements Ce travail a été partiellement soutenu par le programme européen Horizon 2020 au travers des projet NewsEyes (770299) et Embeddia (825153).

Références

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, **5**, 135–146.

BRONSTEIN O., DAGAN I., LI Q., JI H. & FRANK A. (2015). Seed-Based Event Trigger Labeling : How far can event descriptions get us ? In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 372–376.

CHEN Y., XU L., LIU K., ZENG D. & ZHAO J. (2015). Event extraction via dynamic multi-pooling convolutional neural networks. In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 167–176.

CLARK J. H., GARRETTE D., TURC I. & WIETING J. (2021). Canine : Pre-training an efficient tokenization-free encoder for language representation. *arXiv preprint arXiv :1602.02410*.

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT 2019)*, p. 4171–4186, Minneapolis, Minnesota : Association for Computational Linguistics.

DU X. & CARDIE C. (2020). Event Extraction by Answering (Almost) Natural Questions. In *2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 671–683, Online.

DUAN S., HE R. & ZHAO W. (2017). Exploiting Document Level Information to Improve Event Detection via Recurrent Neural Networks. In *Eighth International Joint Conference on Natural Language Processing (IJCNLP 2017)*, p. 352–361 : Asian Federation of Natural Language Processing.

EL BOUKKOURI H., FERRET O., LAVERGNE T., NOJI H., ZWEIGENBAUM P. & TSUJII J. (2020). CharacterBERT : Reconciling ELMo and BERT for word-level open-vocabulary representations from characters. In *28^{th} International Conference on Computational Linguistics (COLING 2020)*, p. 6903–6915, Barcelona, Spain (Online : International Committee on Computational Linguistics.

FENG X., HUANG L., TANG D., JI H., QIN B. & LIU T. (2016). A language-independent neural network for event detection. In *54th Annual Meeting of the Association for Computational Linguistics*, p. 66–71.

JI H., GRISHMAN R. *et al.* (2008). Refining Event Extraction through Cross-Document Inference. In *46th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, p. 254–262.

JOZEFOWICZ R., VINYALS O., SCHUSTER M., SHAZEER N. & WU Y. (2016). Exploring the limits of language modeling. *arXiv preprint arXiv :1602.02410*.

KIM Y., JERNITE Y., SONTAG D. & RUSH A. M. (2016). Character-Aware Neural Language Models. In *Thirtieth AAAI Conference on Artificial Intelligence*, p. 2741–2749.

LI Q., JI H. & HUANG L. (2013). Joint Event Extraction via Structured Prediction with Global Features. In *51st Annual Meeting of the Association for Computational Linguistics*, p. 73–82.

LI W., CHENG D., HE L., WANG Y. & JIN X. (2019). Joint event extraction based on hierarchical event schemas from FrameNet. *IEEE Access*, **7**, 25001–25015.

LIAO S. & GRISHMAN R. (2010). Using document level cross-event inference to improve event extraction. In *48th Annual Meeting of the Association for Computational Linguistics*, p. 789–797 : Association for Computational Linguistics.

LIU J., CHEN Y., LIU K. & ZHAO J. (2018). Event Detection via Gated Multilingual Attention Mechanism. In *Thirty-second AAAI Conference on Artificial Intelligence (AAAI-18)*.

LIU S., CHEN Y., LIU K. & ZHAO J. (2017). Exploiting Argument Information to Improve Event Detection via Supervised Attention Mechanisms. In *55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*, p. 1789–1798, Vancouver, Canada.

LUONG M.-T. & MANNING C. D. (2016). Achieving Open Vocabulary Neural Machine Translation with Hybrid Word-Character Models. In *54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, p. 1054–1063, Berlin, Germany.

MA W., CUI Y., SI C., LIU T., WANG S. & HU G. (2020). CharBERT : Character-aware pre-trained language model. In *28th International Conference on Computational Linguistics (COLING 2020)*, p. 39–50, Barcelona, Spain (Online) : International Committee on Computational Linguistics. DOI : 10.18653/v1/2020.coling-main.4.

MIKOLOV T., CHEN K., CORRADO G. & DEAN J. (2013). Efficient estimation of word representations in vector space. In *International Conference on Learning Representations (ICLR 2013), workshop track*.

NGUYEN T. H., CHO K. & GRISHMAN R. (2016a). Joint Event Extraction via Recurrent Neural Networks. In *2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 300–309.

NGUYEN T. H., FU L., CHO K. & GRISHMAN R. (2016b). A two-stage approach for extending event detection to new types via neural networks. *1st Workshop on Representation Learning for NLP*, p. 158.

NGUYEN T. H. & GRISHMAN R. (2015). Event Detection and Domain Adaptation with Convolutional Neural Networks. In *53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, p. 365–371.

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

PETERS M., NEUMANN M., IYYER M., GARDNER M., CLARK C., LEE K. & ZETTLEMOYER L. (2018). Deep Contextualized Word Representations. In *2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT 2018)*, p. 2227–2237, New Orleans, Louisiana : Association for Computational Linguistics.

SAXE A. M., MCCLELLAND J. L. & GANGULI S. (2014). Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. In *2nd International Conference on Learning Representations (ICLR 2014*.

SUN L., HASHIMOTO K., YIN W., ASAI A., LI J., YU P. & XIONG C. (2020). Adv-BERT : BERT is not robust on misspellings ! Generating nature adversarial samples on BERT. *arXiv preprint arXiv :2003.04985*.

WADDEN D., WENNBERG U., LUAN Y. & HAJISHIRZI H. (2019). Entity, relation, and event extraction with contextualized span representations. In *2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP 2019)*, p. 5784–5789, Hong Kong, China.

WANG X., HAN X., LIU Z., SUN M. & LI P. (2019). Adversarial training for weakly supervised event detection. In *2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies (NAACL-HLT 2019)*, p. 998–1008.

YANG S., FENG D., QIAO L., KAN Z. & LI D. (2019). Exploring Pre-trained Language Models for Event Extraction and Generation. In *57th Annual Meeting of the Association for Computational Linguistics*, p. 5284–5294.

ZHANG T., JI H. & SIL A. (2019). Joint entity and event extraction with generative adversarial imitation learning. *Data Intelligence*, **1**(2), 99–120.

ZHAO Y., JIN X., WANG Y. & CHENG X. (2018). Document embedding enhanced event detection with hierarchical and supervised attention. In *56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, p. 414–419 : Association for Computational Linguistics.