Exploitation des dépendances entre labels pour la classification de textes multi-labels par le biais de transformeurs

Haytame Fallah[*,***], Patrice Bellot[*], Elisabeth Murisasco[**], Emmanuel Bruno[**]

[*] Aix-Marseille Université, Université de Toulon, CNRS, LIS, Marseille, France
[**] Université de Toulon, Aix-Marseille Université, CNRS, LIS, Toulon, France
[***] Hyperbios, Aix-en-Provence, France

Résumé. Nous présentons une nouvelle approche pour améliorer et adapter les transformeurs pour la classification multi-labels de textes. Les dépendances entre les labels sont un facteur important dans le contexte multi-labels. Les stratégies que nous proposons tirent profit des co-occurrences entre labels. Notre première approche consiste à mettre à jour l'activation de chaque label par une somme pondérée de toutes les activations par les probabilités d'occurrence. La deuxième méthode proposée consiste à inclure les activations de tous les labels dans la prédiction, en utilisant une approche similaire au mécanisme de 'self-attention'. Les jeux de données multi-labels les plus connus ont tendance à avoir une faible cardinalité, nous proposons un nouveau jeu de données, appelé 'arXiv-ACM', composé de résumés scientifiques d'arXiv, étiquetés avec leurs mots-clés ACM. Nous montrons que nos approches contribuent à un gain de performance, établissant un nouvel état de l'art pour les jeux de données étudiés.

1 Introduction

La classification multi-labels peut être considérée comme une généralisation de la classification multi-classes. Dans la classification multi-labels de texte *(CMLT)*, l'objectif est d'associer un ou plusieurs labels au texte d'entrée. C'est une tâche importante pour différentes applications telles que la réponse aux questions (Wu et al., 2019; Sahu et al., 2019) où les questions peuvent contenir plus d'un sujet, ou la reconnaissance d'entités (Remolona et al., 2017; de Souza et al., 2020) où les entités peuvent avoir plusieurs catégories sémantiques. Dans ce contexte, le nombre de labels à prédire est plutôt limité, ce qui diffère de la CMLT extrême (Liu et al., 2017; Yu et al., 2019; Shen et al., 2020). Plusieurs méthodes ont été proposées pour la CMLT et peuvent être divisées en deux familles : les méthodes de transformation de problèmes et les méthodes d'ensemble. Les méthodes de transformation de problèmes (Tsoumakas et al., 2010; Luaces et al., 2012) visent à "transformer" le jeu de données pour changer le problème en une classification multi-classes à label unique. Dans les méthodes d'ensemble (Tsoumakas et Vlahavas, 2007; Saini et Ghosh, 2017), plusieurs classifieurs sont formés pour prédire la présence d'un label, puis combinés pour capturer tous les labels présents. En transformant le

problème, ces méthodes ne sont pas de véritables approches multi-labels, et ne tiennent donc pas compte des corrélations entre labels.

Dans la configuration multi-labels, des caractéristiques discriminantes doivent être trouvées pour identifier chaque label dans le texte donné, mais dans certains cas, des dépendances peuvent exister entre les labels. Ces deux facteurs opposés font de la CMLT une tâche difficile. Les cooccurrences et les dépendances entre les labels sont des caractéristiques importantes qui peuvent conduire à une amélioration des performances de classification. Par exemple, un article sur l'informatique est très susceptible de contenir des sujets mathématiques, et peut être lié à des sujets de physique, mais il est très peu probable qu'on y trouve de l'ingénierie électrique.

En fonction du niveau de corrélation des labels utilisé par le modèle, les méthodes de CMLT peuvent être divisées en trois catégories différentes :

— premier ordre : les dépendances entre les labels ne sont pas prises en compte ;
— second ordre : les dépendances par paire entre les labels sont prises en compte ;
— ordre élevé : l'influence de tous les autres labels sur chaque label est imposée.

Avec l'émergence des transformeurs basés sur l'attention (Vaswani et al., 2017) et leur capacité en utilisant uniquement le mécanisme d'attention, à mieux extraire les représentations sémantiques d'un texte, une adaptation de ces modèles pour la CMLT reste à explorer.

Nous proposons dans cet article deux approches d'exploitation de la corrélation des labels qui tirent parti des cooccurrences de labels, d'une manière simple mais efficace, pour la CMLT. Nous nous concentrons principalement sur le réseau à propagation avant (Feed-Forward Neural Network-FFNN) qui est généralement ajouté au modèle Transformeurs pour réaliser une tâche spécifique du TAL. La dernière couche de ce FFNN contient les activations pour chaque label à prédire. Dans la première méthode, nous utilisons la matrice de probabilité de cooccurrence par paire pour mettre à jour les activations de la dernière couche comme suit : l'activation d'un label sera la somme pondérée des activations de tous les labels multipliées par les probabilités de cooccurrence de ce label. Pour la deuxième approche, nous mettons à jour ces activations d'une manière similaire au mécanisme de 'self-attention' (Vaswani et al., 2017), où l'influence d'un label sur un autre n'est pas seulement basée sur la probabilité de co-occurrence mais aussi relative à tous les autres labels. Lorsque ces contraintes sont imposées à chaque neurone correspondant à un label, les dépendances entre labels seront apprises par le modèle tout au long des couches du transformeur. Cette approche peut être considérée comme une méthode d'ordre supérieur puisque l'activation de chaque label influence la prédiction d'un label spécifique.

La CMLT n'est pas présente dans les compétitions actuelles les plus populaires du TAL, le benchmark GLUE par exemple, ni dans les récentes conférences CLEF ou Semeval. En outre, peu de jeux de données de texte multi-labels sont utilisés parmi les articles traitant du problème multi-labels. AAPD (Yang et al., 2018), Reuters (Lewis et al., 2004) et PubMed (NCBI Resource Coordinators, 2016; Tsatsaronis et al., 2015) semblent être les jeux de données les plus utilisés. Mais la plupart d'entre eux souffrent d'une faible cardinalité (nombre moyen de labels par instance), avec un grand nombre d'instances n'ayant qu'un seul label, ce qui rend difficile l'expérimentation de nouvelles approches pour la CMLT. Nous introduisons ainsi un nouveau jeu de données multi-labels à cardinalité élevée, construit en associant les résumés d'articles scientifiques à leurs mots-clés ACM donnés par les auteurs.

Nous évaluons notre approche en utilisant une architecture basée sur BERT (Devlin et al., 2019) sur Reuters, une collection d'articles d'actualité, et d'articles scientifiques AAPD, et

notre nouveau jeu de données ‘arXiv-ACM’. Nous montrerons que nos approches conduisent à un gain en performance de prédiction sur tous ces jeux de données.

Nos deux principales contributions sont les suivantes :

— deux nouvelles méthodes qui permettent aux approches de classification basées sur les transformeurs d’apprendre les dépendances qui existent entre les labels en utilisant les informations de co-occurrence disponibles dans le jeu de données ;

— un nouveau jeu de données à cardinalité élevée construit à partir de arXiv.org, bien adapté à la classification multi-labels.

2 Travaux antérieurs

La classification multi-labels consiste à pouvoir associer chaque entrée X à plusieurs labels $\mathbf{Y}$, plutôt qu’à un seul. Les méthodes de classification multi-labels peuvent être classées en trois catégories : transformation du problème, adaptation et méthodes d’ensemble. Ces méthodes peuvent soit ignorer la corrélation des labels (premier ordre), soit prendre en compte les dépendances qui peuvent exister entre les labels (second ordre et ordre supérieur).

2.1 Stratégies de classification multi-labels

La transformation du problème consiste à "transformer" le jeu de données pour convertir le problème en une classification multi-classes à un seul label. Une de ces méthodes consiste à considérer toutes les combinaisons uniques possibles de labels, *label powerset* (Tsoumakas et al., 2010), et à former un classifieur multi-classes $M : X \rightarrow P(Y)$, où $P(Y)$ est l’ensemble des sous-ensembles distincts de labels. Outre le nombre élevé de labels possibles qui peut atteindre $2^{|Y|}$, le défi consiste à trouver suffisamment d’exemples pour chaque combinaison de labels. Il est important de noter qu’en transformant le problème en une classification multi-classes, les dépendances qui peuvent exister entre les différents labels ne sont plus considérées.

Un ensemble de classifieurs multi-classes peut être combiné pour créer un classifieur multi-labels. Pour une instance donnée, chaque classifieur prédit un seul label et toutes les sorties de ces classifieurs sont ensuite combinées par une méthode d’ensemble. L’algorithme $RAKEL$ (Tsoumakas et Vlahavas, 2007) est une autre variante de cette méthode. L’utilisation de classifieurs multiples impose de fortes contraintes d’utilisation mémoire, ainsi que la nécessité d’optimiser plusieurs modèles qui augmentent linéairement avec le nombre de labels.

Une adaptation des algorithmes d’apprentissage profond, sans transformation préalable des données, pourrait être une méthode meilleure et plus efficiente pour la CMLT.

2.2 Dépendances des labels

Il a été démontré que la capture explicite de la dépendance entre les labels améliore les performances de la classification multi-labels (Zhang et Zhou, 2007), de nombreuses méthodes ont été proposées pour modéliser cette dépendance.

La structure hiérarchique qui peut exister entre les labels a été utilisée pour tenter de mieux explorer les relations entre les labels (Yang et al., 2016; Alaydie et al., 2012). Les graphes et réseaux conditionnels (Zhang et Zhang, 2010; Guo et Gu, 2011) utilisent les dépendances hiérarchiques sémantiques et les cooccurrences de labels, ou un mélange des deux (Wu et al.,

2018), pour construire des réseaux de dépendance combinés avec le modèle principal. Cependant, les labels d'un jeu de données ne sont pas toujours de nature hiérarchique. De plus, ces méthodes négligent les dépendances latérales entre les labels au profit des relations verticales.

MAGNET (Pal et al., 2020), un réseau de graphes qui utilise les plongements de BERT, met en œuvre le mécanisme d'attention pour capturer la structure de dépendance entre les labels. Cette méthode parvient à obtenir de bonnes performances en F1 pour AAPD et Reuters (cf. section 5.1) tout en utilisant une variante de LSTM pour les représentations textuelles.

Kurata et al. (2016) implémentent une couche cachée dédiée connectée à la couche de classification de sortie. Les poids entre ces deux couches sont initialisés en utilisant des modèles de cooccurrences. Cette méthode n'est pas agnostique aux modèles, et impose des valeurs pour les poids ce qui peut entraver le processus d'apprentissage.

Liu et al. (2022) encodent les labels en embeddings et les introduisent dans le mécanisme d'attention avec la séquence de texte. La dépendance entre les labels peut être considérée comme prise en compte via le mécanisme d'attention, mais cette approche n'aboutit à des améliorations de performances que dans le cas où les labels sont constitués de mots pleins et non pas d'abréviations ou de codes (par ex. 'cs.it' pour le jeu de données AAPD).

3 Exploitation des dépendances à l'aide de transformeurs

Nous détaillons dans cette section comment l'architecture basée sur les transformeurs est adaptée à la CMLT, et comment nous utilisons les informations de cooccurrence pour permettre aux transformeurs (Devlin et al., 2019) d'apprendre des dépendances de label d'ordre élevé.

3.1 Classification multi-label avec BERT

BERT introduit un token de classification **[CLS]** contenant un *état caché* de la phrase, mis à jour dans chaque couche du modèle tout au long du processus d'entraînement.

Un feed-forward neural network (FFNN) de L couches denses est ajouté à la dernière couche du modèle. Ceci est fait pour affiner le transformeur pré-entraîné pour la tâche désirée (classification de texte dans notre cas). Le token **[CLS]** constitue l'entrée de ce FFNN.

Pour la CMLT, les valeurs des activations de la couche de sortie peuvent être utilisées pour déterminer la présence d'un label. Nous utilisons la fonction d'activation *Sigmoïde* σ pour lier chaque activation à une probabilité de la présence du label correspondant. Nous utilisons l'entropie croisée binaire comme fonction de perte, qui est la mieux adaptée à ce cas.

3.2 Apprentissage de dépendance des labels

Notre objectif est de mettre à jour les activations de la dernière couche du FFNN, où l'activation de chaque label est influencée par les activations de tous les labels, en utilisant uniquement les probabilités d'occurrence par paire. Nous utilisons les informations de cooccurrence extraites du jeu de données d'apprentissage en construisant une matrice de cooccurrence $C^{n\times n}$, avec n étant le nombre de labels cibles, comme suit :

$$C_{i|j} = \frac{\text{Instances où le label } j \text{ et } i \text{ sont présents ensemble}}{\text{Instances où le label } i \text{ est présent}} \tag{1}$$

$C_{i|j}$ est la probabilité de la présence du label *j* étant donné la présence du label *i*.

Approche de mise à jour simple (Simple Update-SU) Nous utilisons la matrice de cooccurrence C pendant l'apprentissage pour mettre à jour les activations de la dernière couche de classification du FFNN $A^{[L]}$, en multipliant simplement ces activations par la matrice de cooccurrence. Nous appliquons ensuite la fonction d'activation *Sigmoïde* σ sur le vecteur résultant.

Chaque activation correspondant à un label *i* est mise à jour par les valeurs des activations des autres labels, pondérées par les probabilités de cooccurrence. Le modèle apprend alors ces dépendances de label au fur et à mesure que l'information est propagée dans les couches du modèle pendant le processus d'apprentissage.

Méthode de mise à jour par self-attention (Self-Attention Update-SAU). Cette approche s'inspire du mécanisme du 'self-attention' de l'architecture transformeur. Avec ce mécanisme, la représentation d'un token (son plongement) est mise à jour par tous les plongements de la séquence d'entrée. Dans Vaswani et al. (2017), le self-attention est définie comme suit :

$$Attention(Q, K, V) = Softmax(\frac{QK^T}{\sqrt{dk}})V \tag{2}$$

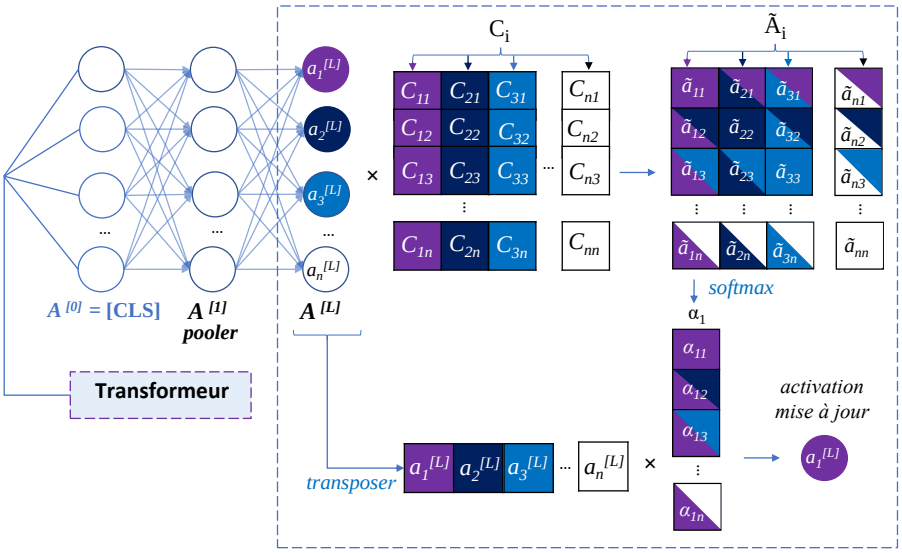

FIG. 1 – La mise à jour des activations utilise une approche similaire au mécanisme du self-attention, où l'influence d'un label sur un autre est relative à tous les autres labels.

Q, K, V étant les matrices de requêtes, de clés et de valeurs, et dk étant la dimension des embeddings. Nous cherchons à mettre à jour les activations de la dernière couche du FFNN en pondérant les activations des labels et les valeurs de cooccurrence de manière similaire au self-attention. La Requête et la Valeur sont dans ce cas les activations de la dernière couche, et la Clé correspond à la matrice des probabilités de cooccurrence C :

$$A^{[L]} = \sigma(Softmax(\frac{A^{[L]}C}{\sqrt{n}})A^{[L]}) \tag{3}$$

$$a_i^{[L]} = \sigma(Softmax(\frac{A^{[L]}C_i}{\sqrt{n}})A^{[L]}) \quad (4)$$

L'objectif de cette approche est d'incorporer l'information de dépendance contenue dans les probabilités de cooccurrence d'une manière plus significative. Plutôt que de simplement pondérer les activations par ces seules probabilités, l'utilisation de la fonction $softmax$ pour obtenir la "contribution réelle" de chaque label par rapport à tous les labels peut être un moyen plus efficace d'encoder les dépendances des labels. Comme l'illustre la figure 1, pour chaque label i, toutes les valeurs d'activation sont multipliées par le vecteur de poids C_i de la matrice de cooccurrence correspondant à ce label, la fonction $softmax$ est ensuite appliquée aux activations pondérées résultantes $\tilde{A}_i$ pour obtenir des scores "d'attention" relatifs pour chaque label α_i. La somme pondérée des activations originales A^L et du vecteur des scores d'attention α_i calculés pour un label spécifique sera la valeur actualisée de l'activation de ce label. La fonction *sigmoïde* est toujours utilisée afin d'obtenir une probabilité de prédiction valide.

4 arXiv-ACM Dataset

Les jeux de données couramment utilisés, détaillés dans la section 5.1, présentent de nombreuses limitations. La plus contraignante est le nombre d'instances par label. En général, les instances avec un seul label sont plus fréquentes que les labels multiples. Cela peut introduire un biais de classification, où les modèles sont plus susceptibles d'apprendre à prédire un seul label, en général le plus fréquent. Pour remédier à ces limitations, nous introduisons dans cet article un nouveau jeu de données multi-labels, que nous appelons 'arXiv-ACM', avec une cardinalité élevée, une taille raisonnable et une meilleure distribution des instances par nombre de labels. Ce jeu de données est composé de résumés d'articles en informatique publiés entre 1998 et 2021 extraits via l'API arXiv [1]. Ces résumés ont ensuite été appariés avec les mots-clés ACM [2] fournis par les auteurs des articles. Seuls les mots-clés de deuxième niveau ont été pris en compte, le premier niveau étant large et les niveaux suivants trop spécifiques. Nous avons ensuite filtré les labels qui comptaient moins de 20 occurrences pour obtenir au final 64 labels. Le tableau 1 et la figure 2 présentent certaines caractéristiques de ce jeu de données par rapport à AAPD et Reuters, les autres jeux de données traités dans cet article.

	#Train	#Valid	#Test	labels	W	Card
arXiv-ACM	9600	2157	2160	64	152	2,33
AAPD	53840	1000	1000	54	163,16	1,4
Reuters-21578	6770	1000	3019	90	127,76	1,24

TAB. 1 – Jeux de données utilisés, W est le nombre moyen de mots par résumé, Card est le nombre moyen de labels par instance.

1. https ://arxiv.org/help/api/
2. https ://www.acm.org/publications/computing-classification-system/1998/ccs98

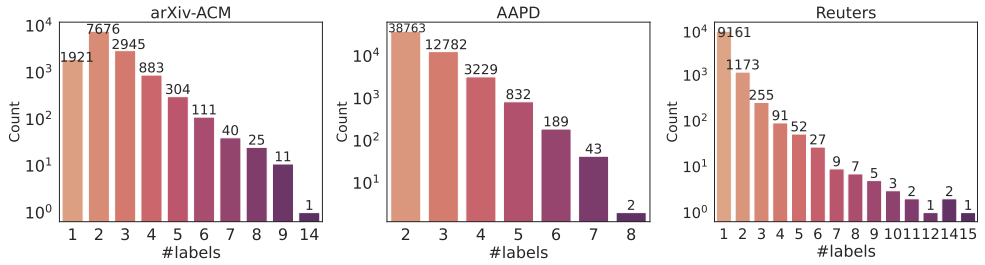

FIG. 2 – Nombre d'instances basé sur le nombre de labels pour tous les jeux de données. Pour AAPD, le nombre de labels commence à 2, car certaines paires de labels apparaissent toujours ensemble. Dans Reuters et AAPD, les instances avec un seul label sont beaucoup plus présentes que celles avec des labels multiples. En construisant arXiv-ACM nous visons à résoudre ce problème qui est présent dans plusieurs jeux de données multi-labels.

5 Expériences et résultats

5.1 Jeux de données

Nous fournissons des détails sur les jeux de données utilisés [3] :

— **arXiv-ACM** : Le nouveau jeu de données présenté dans cet article, où les résumés peuvent avoir un ou plusieurs mots-clés ACM ;

— **Reuters-21578** [4] est une collection d'articles du fil d'actualité Reuters de l'année 1987. Il s'agit d'un jeu de données qui a souvent été utilisé pour évaluer les modèles de CMLT. Un article peut appartenir à un ou plusieurs des 90 domaines du jeu de données ;

— **AAPD** (ou arXiv Academic Paper Dataset) est, de manière similaire à notre proposition de jeu de données 'arXiv-ACM', une collection de "résumés" de plusieurs publications scientifiques. Un article peut avoir une ou plusieurs classifications parmi 54 labels. Nous utilisons la même distribution d'entraînement (53840), de validation (1000) et de test (1000) que Yang et al. (2018).

5.2 Méthode d'évaluation

Pour l'évaluation des méthodes proposées, i.e. BERT+SU (Simple Update) et BERT+SAU (Self-Attention Update), nous utilisons l'implémentation (Wolf et al., 2020) de HugginFace de la version *uncased-base* de **BERT**, avec 12 couches transformeurs comportant chacune 12 têtes d'attention, un vecteur de plongements de 768 dimensions, et une longueur de séquence de 512 tokens. Nous utilisons l'entropie croisée binaire (BCE) pour la version base de BERT ainsi que pour nos approches, avec un taux d'apprentissage de 2×10^{-5}. AdamW (Loshchilov

3. Tous les jeux de données, ainsi que le code d'implémentation, peuvent être téléchargés sur GitHub : `https://anonymous.4open.science/r/EGC2023-350/`

4. `https://archive.ics.uci.edu/ml/datasets/reuters-21578+text+categorization+collection`

Modèles	arXiv-ACM			Reuters			AAPD		
	Pr.	R	F1	Pr.	R	F1	Pr.	R	F1
Baselines									
GradientBoost	57,99	29,87	39,43	88,06	80,56	84,14	79,73	46,8	58,98
SVM	**70,15**	39,79	50,78	**94,19**	79,62	80,64	**80,85**	59,98	68,86
MAGNET	57,31	53,24	55,2	91,2	88,6	89,9	72,88	66,79	69,7
CB-NTR	60,11	55,74	57,84	91,37	90,34	90,85	75,44	72,85	74,12
CNLE	56,85	52,37	54,52	90,9	88,7	89,8	74,71	69,11	71,8
BERT	60,04	55,58	57,72	91,22	90,33	90,77	76,33	71,95	74,07
Nos approches d'apprentissage par dépendance									
BERT+ SU	59,57	**57,73**	**58,63**	91,59	90,3	**90,94**	75,63	**73,12**	**74,35**
BERT+ SAU	60,33	56,02	58,10	91,35	**90,38**	90,86	75,84	72,66	74,22

TAB. 2 – Scores pour le jeu de données de test provenant de arXiv-ACM, Reuters et AAPD, les meilleurs scores sont en bleu gras. *Orig.* fait référence au résultat original tiré des articles correspondants.

et Hutter, 2022) est l'optimiseur utilisé avec $betas = (0,9; 0,999)$, et une dégradation des pondérations (weight decay) de $0,01$.

Nous comparons nos approches avec des approches non-neuronales, à savoir GradientBoosting et SVM, en utilisant TF-IDF (sans nombre maximum de caractéristiques) comme entrées, dans une approche One-vs-Rest. Le GradientBoosting est utilisé avec la régression logistique comme fonction d'erreur, un taux d'apprentissage de 0,1 et 100 estimateurs, tandis que nous utilisons le SVM avec un noyau de type RBF et une régularisation de 1,0. Nous comparons également nos approches à d'autres approches neuronales :

— **MAGNET** (Pal et al., 2020) : un réseau de graphes implémentant le mécanisme d'attention pour capturer les dépendances entre les labels ;
— **CB-NTR** (Huang et al., 2021) utilisant des fonctions de perte adaptées pour l'équilibrage des classes dans le cas des jeux de données déséquilibrés ;
— **CNLE** (Liu et al., 2022) qui introduit les plongements des labels dans le mécanisme d'attention avec le texte à classifier.

5.3 Résultats

Le tableau 2 montre les scores de micro-précision, de micro-rappel et de micro-F1 pour tous les jeux de données. Ces scores sont calculés sur une moyenne de 5 exécutions sans modification des hyperparamètres des modèles.

L'utilisation des informations de dépendance contenues dans la matrice de cooccurrence entraîne une augmentation du score micro-F1 pour tous les jeux de données.

Le SVM peut être considéré comme l'approche non neuronale la plus performante, mais n'est pas à la hauteur des autres méthodes testées, même si le SVM et GradientBoost obtiennent les scores de micro-précision les plus élevés pour tous les jeux de données. Cette précision élevée se fait au prix d'un rappel plus faible, ce qui réduit le score micro-F1.

Pour le jeu de données **Reuters**, la version de base de BERT parvient déjà à obtenir de bonnes performances. La méthode CB-NTR réalise un petit gain en précision et en rappel (0,15 et 0,01 respectivement), contribuant à un gain en micro-F1 avec 90,85 par rapport à la méthode de base qui utilise la perte d'entropie croisée binaire (90,77). Les approches d'apprentissage par dépendance pour ce jeu de données obtiennent une augmentation plus importante de la micro-précision de 0,37 et 0,13 pour les méthodes **SA** et **SAU** respectivement, avec des gains marginaux dans le micro-rappel (0,05 pour la méthode SAU), mais suffisants pour surpasser toutes les autres méthodes avec un score micro-F1 de 90,94 et 90,86 respectivement, une augmentation maximale de 0,17 par rapport à la version de base de BERT. Un résultat obtenu sur une moyenne de 5 exécutions avec un écart-type $\sigma = 0,013$.

En raison de la nature du vocabulaire utilisé dans les résumés scientifiques, **AAPD** est un jeu de données plus complexe que Reuters. Dans les articles scientifiques, plusieurs domaines et disciplines peuvent être impliqués. Avec un vocabulaire plus spécifique et des mots plus précis (moins d'homonymes et de synonymes), les tâches de modélisation du langage et de classification sont plus difficiles pour AAPD. L'apprentissage par dépendance permet cette fois d'obtenir une augmentation du rappel de 1,17 point pour la méthode **S**imple **U**pdate et de 0,71 pour la méthode **S**elf-**A**ttention inspired **U**pdate. La méthode **SU** réalise un gain plus important par rapport à CB-NTR (avec un gain de 0,9 en rappel) et permet à notre approche de surpasser une fois de plus toutes les autres méthodes avec un score micro-F1 de 74,35.

La nature du jeu de données **arXiv-ACM** est similaire à celle d'AAPD mais se confirme être un jeu de données plus difficile, notamment parce que la proportion d'instances avec plus d'un label est plus élevée. La version de base de BERT parvient à obtenir un score micro-F1 de 57,72. CB-NTR contribue à un léger gain en précision et en rappel, avec une augmentation de 0,12 du score micro-F1. La méthode **SU** présente le gain le plus élevé en rappel dans tous les jeux de données, avec une augmentation de 2,15. Cette augmentation notable contribue au meilleur score micro-F1 pour ce jeu de données (58,63, $\sigma = 0,021$), avec la plus forte augmentation par rapport à la version BERT de base.

Ce gain de performance obtenu par les approches d'apprentissage des dépendances que nous proposons peut s'expliquer par le fait que la prédiction d'un label est influencée par la prédiction de tous les autres labels à l'aide des cooccurrences. Dans certains cas, cette information permet de prédire des labels qui n'auraient pas été prédits autrement (augmentation du rappel). D'autre part, les dépendances des labels peuvent diminuer le nombre de faux positifs en réduisant le biais que le modèle peut avoir pour les labels fréquents dans le jeu de données, ce qui contribue à un gain en précision. La méthode de mise à jour simple est étonnamment plus efficace que sa contrepartie basée sur le self-attention, nous suggérons que l'imposition de contraintes complexes sur les activations pourrait avoir des effets dégradants sur la performance globale et que les informations de dépendance de label sont beaucoup plus difficiles à propager dans les couches du modèle. Néanmoins, la méthode **SAU** parvient à obtenir une augmentation par rapport à la version de premier ordre de BERT et de légers gains de performance par rapport aux autres méthodes.

6 Conclusion

La classification multi-labels de textes est une tâche importante pour de nombreuses applications. Malheureusement, elle n'est pas incluse dans les benchmarks les plus populaires tels

que GLUE. Nous avons proposé dans cet article des moyens simples mais efficaces, utilisant les informations de cooccurrence des labels par paire, pour permettre aux modèles transformeurs d'apprendre les dépendances entre les labels. Ces méthodes de dépendance de labels d'ordre élevé sont agnostiques par rapport au modèle et ne sont pas limitées à la classification de textes, mais peuvent être utilisées pour toute autre tâche de classification multi-labels.

Nous avons testé et montré que les cooccurrences et les dépendances entre labels peuvent être utilisées pour obtenir un gain de performance tangible pour la classification multi-labels de textes. Et ce, pour tous les jeux de données, mais avec un gain plus perceptible pour les jeux de données équilibrés (arXiv-ACM ici). La méthode de mise à jour de l'activation basée sur l'auto-attention montre des résultats prometteurs mais n'est toujours pas aussi efficace que la méthode de mise à jour de l'activation simple. L'ajout d'une couche cachée supplémentaire juste avant la couche de sortie, ou la projection du vecteur d'activation dans des matrices distinctes de requête, de clé et de valeur (de manière similaire au mécanisme d'auto-attention dans les transformeurs) pourrait améliorer l'efficacité de cette méthode.

Nous avons introduit dans cet article un nouveau jeu de données multi-labels qui est plus approprié pour tester de nouvelles approches multi-labels. Il vise à répondre aux limitations des jeux de données couramment utilisés, en termes de distribution du nombre de labels et d'équilibre des classes. Nous avons rendu public ce jeu de données 'arXiv-ACM', ainsi que le code d'implémentation des approches proposées ici.

Références

Alaydie, N., C. K. Reddy, et F. Fotouhi (2012). Exploiting Label Dependency for Hierarchical Multi-label Classification. In P.-N. Tan, S. Chawla, C. K. Ho, et J. Bailey (Eds.), *Advances in Knowledge Discovery and Data Mining*, Lecture Notes in Computer Science, Berlin, Heidelberg, pp. 294–305. Springer.

de Souza, J. V. A., E. T. R. Schneider, J. O. Cezar, L. E. Silva, Y. B. Gumiel, E. Paraiso, D. Teodoro, et C. M. C. M. Barra (2020). A multilabel approach to portuguese clinical named entity recognition. *Journal of Health Informatics 12*, 366–372.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019). BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL*.

Guo, Y. et S. Gu (2011). Multi-label classification using conditional dependency networks. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence - Volume Volume Two*, IJCAI'11, Barcelona, Catalonia, Spain, pp. 1300–1305. AAAI Press.

Huang, Y., B. Giledereli, A. Köksal, A. Özgür, et E. Ozkirimli (2021). Balancing Methods for Multi-label Text Classification with Long-Tailed Class Distribution. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic, pp. 8153–8161. Association for Computational Linguistics.

Kurata, G., B. Xiang, et B. Zhou (2016). Improved Neural Network-based Multi-label Classification with Better Initialization Leveraging Label Co-occurrence. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, San Diego, California, pp. 521–526. Association for Computational Linguistics.

Lewis, D. D., Y. Yang, T. G. Rose, et F. Li (2004). RCV1 : A New Benchmark Collection for Text Categorization Research. *Journal of Machine Learning Research 5*(Apr), 361–397.

Liu, J., W.-C. Chang, Y. Wu, et Y. Yang (2017). Deep learning for extreme multi-label text classification. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, New York, NY, USA, pp. 115–124. Association for Computing Machinery.

Liu, M., L. Liu, J. Cao, et Q. Du (2022). Co-attention network with label embedding for text classification. *Neurocomputing 471*, 61–69.

Loshchilov, I. et F. Hutter (2022). Decoupled Weight Decay Regularization.

Luaces, O., J. Díez, J. Barranquero, J. J. del Coz, et A. Bahamonde (2012). Binary relevance efficacy for multilabel classification. *Progress in Artificial Intelligence 1*(4), 303–313.

NCBI Resource Coordinators (2016). Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research 44*(D1), D7–19.

Pal, A., M. Selvakumar, et M. Sankarasubbu (2020). Multi-Label Text Classification using Attention-based Graph Neural Network. In *ICAART*.

Remolona, M. F. M., M. F. Conway, S. Balasubramanian, L. Fan, Z. Feng, T. Gu, H. Kim, P. M. Nirantar, S. Panda, N. R. Ranabothu, N. Rastogi, et V. Venkatasubramanian (2017). Hybrid ontology-learning materials engineering system for pharmaceutical products : Multi-label entity recognition and concept detection. *Computers & Chemical Engineering 107*, 49–60.

Sahu, T. P., R. S. Thummalapudi, et N. K. Nagwani (2019). Automatic Question Tagging Using Multi-label Classification in Community Question Answering Sites. In *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/ 2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, pp. 63–68.

Saini, R. et S. Ghosh (2017). Ensemble classifiers in remote sensing : A review. In *2017 International Conference on Computing, Communication and Automation (ICCCA)*, pp. 1148–1152.

Shen, Y., H.-f. Yu, S. Sanghavi, et I. Dhillon (2020). Extreme Multi-label Classification from Aggregated Labels. *arXiv :2004.00198 [cs, stat]*.

Tsatsaronis, G., G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, Y. Almirantis, J. Pavlopoulos, N. Baskiotis, P. Gallinari, T. Artiéres, A.-C. N. Ngomo, N. Heino, E. Gaussier, L. Barrio-Alvers, M. Schroeder, I. Androutsopoulos, et G. Paliouras (2015). An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition. *BMC Bioinformatics 16*(1), 138.

Tsoumakas, G., I. Katakis, et I. Vlahavas (2010). Mining Multi-label Data. In O. Maimon et L. Rokach (Eds.), *Data Mining and Knowledge Discovery Handbook*, pp. 667–685. Boston, MA : Springer US.

Tsoumakas, G. et I. Vlahavas (2007). Random k -Labelsets : An Ensemble Method for Multilabel Classification. Volume 4701, pp. 406–417.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, et I. Polosukhin (2017). Attention is All you Need. In *Advances in Neural Information Processing*

Systems, Volume 30. Curran Associates, Inc.

Wolf, T., L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, et A. M. Rush (2020). HuggingFace's Transformers : State-of-the-art Natural Language Processing. Technical Report arXiv :1910.03771, arXiv.

Wu, B., F. Jia, W. Liu, B. Ghanem, et S. Lyu (2018). Multi-label Learning with Missing Labels Using Mixed Dependency Graphs. *International Journal of Computer Vision 126*(8), 875–896.

Wu, H., S. Zhang, J. Wang, M. Liu, et S. Li (2019). Multi-label Aspect Classification on Question-Answering Text with Contextualized Attention-Based Neural Network. In M. Sun, X. Huang, H. Ji, Z. Liu, et Y. Liu (Eds.), *Chinese Computational Linguistics*, Lecture Notes in Computer Science, Cham, pp. 479–491. Springer International Publishing.

Yang, P., X. Sun, W. Li, S. Ma, W. Wu, et H. Wang (2018). SGM : Sequence Generation Model for Multi-label Classification. In *Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, New Mexico, USA, pp. 3915–3926. Association for Computational Linguistics.

Yang, Z., D. Yang, C. Dyer, X. He, A. Smola, et E. Hovy (2016). Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, San Diego, California, pp. 1480–1489. Association for Computational Linguistics.

Yu, H.-F., K. Zhong, I. S. Dhillon, W.-C. Wang, et Y. Yang (2019). X-bert : extreme multi-label text classification using bidirectional encoder representations from transformers. In *NeurIPS 2019 Workshop on Science Meets Engineering of Deep Learning*.

Zhang, M.-L. et K. Zhang (2010). Multi-label learning by exploiting label dependency. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, New York, NY, USA, pp. 999–1008. Association for Computing Machinery.

Zhang, M.-L. et Z.-H. Zhou (2007). ML-KNN : A lazy learning approach to multi-label learning. *Pattern Recognit.*.

Summary

We introduce a new approach to improve and adapt transformers for multi-label text classification. Dependencies between labels are an important factor in the multi-label context. Our proposed strategies take advantage of co-occurrences between labels. Our first approach consists in updating the final activation of each label by a weighted sum of all activations by these occurrence probabilities. The second proposed method consists in including the activations of all labels in the prediction. This is done using an approach similar to the 'self-attention' mechanism. As the most known multi-label datasets tend to have a small cardinality, we propose a new dataset, called 'arXiv-ACM', comprised of scientific abstracts from arXiv, tagged with their ACM keywords. We show that our approaches contribute to a performance gain, establishing a new state of the art for the studied datasets.