Benchmark pour la classification de commentaires toxiques sur le jeu de données Civil Comments

Corentin Duchêne*, Henri Jamet*, Pierre Guillaume*, Réda Dehak*

* EPITA Speaker and Language Recognition Group (ESLR),
Laboratoire de Recherche de l'EPITA (LRE), France
{prénom.nom}@epita.fr

Résumé. La détection des commentaires toxiques sur les réseaux sociaux est devenue essentielle pour la modération automatique des messages. Dans cet article, nous présentons une comparaison d'un large éventail de modèles sur un ensemble de données multi-labels de discours haineux. Nous prenons en compte dans notre comparaison le temps d'inférence, les performances et le biais en utilisant différentes métriques. Nous avons découvert que tous les modèles BERT ont des performances similaires, indépendamment de leur taille, des optimisations ou du langage utilisé pour le pré-entraînement. Les réseaux BiLSTM restent un bon compromis entre la performance et le temps d'inférence. Le modèle RoBERTa utilisant la fonction Focal Loss pour l'entraînement demeure le moins biaisé de tous. Comme prévu, le modèle DistilBERT a le temps d'inférence le plus faible des modèles BERT. Enfin, tous les modèles sont affectés par le biais d'association des identités à la toxicité. Les modèles BERT, RNN et XLNet y sont moins sensibles que les CNN et les Compact Convolutional Transformers.

1 Introduction

La détection des commentaires toxiques sur les médias sociaux s'est avérée essentielle pour la modération automatique du contenu. Selon le ministre français de l'éducation, 18% des élèves français ont été victimes de harcèlement sur les réseaux sociaux en 2021. Dans le même temps, le nombre de publications sur ces plateformes n'a cessé d'augmenter. En 12 ans, le nombre de tweets par jour a été multiplié par dix pour atteindre 500 millions aujourd'hui [1].

Cela montre que la détection rapide et ciblée des commentaires toxiques sur les réseaux sociaux est devenue un enjeu crucial pour assurer la cohésion de la société. Par conséquent, cela ne peut se faire qu'en automatisant la modération en ligne.

De nos jours, les types de modèles les plus performants en matière de classification de textes et représentant l'état de l'art sont des modèles basés sur des Transformers (Vaswani et al., 2017) tels que le modèle BERT (Devlin et al., 2019). Plusieurs modifications de ce modèle ont été proposées, Liu et al. (2019) ont fine-tuné un modèle BERT préentraîné pour identifier les discours offensants, catégorisant automatiquement les types de haine et identifiant la cible de ces commentaires.

1. https://www.internetlivestats.com/twitter-statistics/, Statistiques d'utilisation de Twitter - Internet Live Stats.

Dans cette étude, nous comparons les modèles les plus performants et les plus répandus en traitement du langage naturel, comme BERT, et en vision appliquée au texte, comme des ResNet et des Vision Transformers. À notre connaissance, nous n'avons pas trouvé dans l'état de l'art une comparaison aussi détaillée de tous ces modèles sur un large éventail de métriques en utilisant les mêmes conditions d'entraînement et les mêmes ensembles de données d'entraînement et de test. La plupart des benchmarks récents (Lee et al., 2018; Ibrohim et Budi, 2019) s'attachent à comparer différentes méthodes d'apprentissage automatique ou d'apprentissage profond en s'appuyant seulement sur des métrics comme le F1 score, la Precision ou le Recall. Cependant les travaux de Dixon et al. (2018) tendent à montrer que ces métriques ne sont pas suffisantes pour évaluer certains biais de classification. Borkan et al. (2019) proposent même leurs propres métriques pour évaluer ces biais. Comme jamais auparavant, la même méthodologie et le même ensemble de données sont utilisés tout au long de notre analyse pour se concentrer sur la performance, la mesure des biais et le temps d'inférence. Nous avons finetuné chacun de nos modèles pour obtenir les meilleures performances. Le résultat de ce travail devrait aider à déterminer quel modèle peut être utilisé dans la pratique.

Notre comparaison a été effectuée à l'aide des mêmes ensembles de données d'entraînement et de test extraits de Civil Comments 2019 [2]. Ce jeu de données est un jeu de données multi-labels avec des classes déséquilibrées fournies par Jigsaw/Conversation AI. Pour ce jeu de données, nous connaissons l'identité de la cible pour certains commentaires, ce qui nous permet d'évaluer les biais lors de la classification.

Le reste du document est organisé comme suit : La section 2 décrit le jeu de données et les modèles utilisés dans la comparaison. Le protocole d'expérimentation et l'analyse des résultats sont présentés dans les sections 3 et 4. Enfin, la section 5 conclut l'article.

2 Méthodologie

2.1 Jeu de données

En 2017, la plateforme d'hébergement de commentaires Civil Comments a fermé. Elle a rendu publics ses 1,8 million de commentaires pour soutenir la recherche visant à comprendre et à améliorer la détection de la haine dans les conversations en ligne. L'équipe de Jigsaw a soutenu cette initiative pour l'étiquetage de ces commentaires ; chaque commentaire a été montré à 10 annotateurs en leur demandant de "noter la toxicité du commentaire". Pour garantir l'exactitude des notes, certains commentaires ont été vus par plus de 100 annotateurs. Pour tous les commentaires, la valeur obtenue à la fin pour chaque classe est la fraction des annotations positives par rapport au nombre d'annotateurs. Tous les commentaires ont été classés en sept catégories : `toxicity`, `severe_toxicity`, `obscene`, `threat`, `insult`, `indentity_attack`, et `sexual_explicit`.

En outre, un sous-ensemble de 450 000 exemples a été étiqueté avec l'identité visée par chaque commentaire (Table 1) en utilisant une liste de questions, telles que "Quels genres sont référencés dans le commentaire ?" ou "Quelles ethnies sont référencées dans le commentaire ?". Encore une fois, le score obtenu pour chaque classe d'identité est la fraction d'annotateurs qui ont mentionné l'identité sur le nombre d'évaluateurs. Comme on peut le voir dans le Tableau 2,

2. `https://www.kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification/data` : Jigsaw unintended bias in toxicity classification Kaggle.

Catégorie	Identité
Gender	Male, Female, Transgender, Other gender
Sexual Orientation	Heterosexual, Homosexual, Bisexual, Other sexual orientation
Religion	Christian, Jewish, Muslim, Hindu, Buddhist, Atheist, Other religion
Race or ethnicity	Black, White, Latino, Other race or ethnicity
Disability	Physical disability, Intellectual or learning disability. Psychiatric disability or mental illness, Other disability

TAB. 1 – Liste des options de communautés présentées aux annotateurs.

Sous-groupe	Nombre	Fréquence
all comments	1 999 516	7.99%
male	48 870	15.05%
female	58 584	13.66%
transgender	2 759	21.13%
heterosexual	1 432	22.56%
homosexual	12 062	28.28%

TAB. 2 – Pourcentage de commentaires qualifiés de toxiques pour une sélection de communautés.

il y a un déséquilibre dans le pourcentage d'annotation de toxicité entre les différentes communautés.

2.2 Pré-traitement des données

Dans la base utilisée, pour chaque commentaire, les étiquettes représentent la probabilité d'appartenance ou pas aux différentes classes de toxicité et d'identité. Il faut rappeler que ces classes ne sont pas disjointes. . Pour déterminer si un commentaire est considéré comme positif ou négatif pour chaque classe, nous avons appliqué un seuil : si la probabilité est supérieure à 0,5, nous supposons que le commentaire est positif pour cette classe, sinon c'est négatif. A la fin, chaque commentaire peut être affecté à une des classes ou à un sous-ensemble des différentes classes.

Comme on peut le voir sur le Tableau 3, les classes sont fortement déséquilibrées. La classe `severe_toxicity` est rarement activée sur l'ensemble du jeu de données. Pour cette raison, nous avons choisi de retirer cette classe des classes à prédire et de limiter le nombre de classes à six.

Pour l'ensemble des commentaires traités, nous avons appliqué les transformations suivantes :

— Transformer en minuscules,
— Supprimer les balises HTML, les l'URL et les signes diacritiques,
— supprimer les espaces blancs et les valeurs NA ou vides

Sous-type de haine	Nombre
toxicity	159 782
severe_toxicity	13
obscene	10 671
sexual_explicit	5 127
identity_attack	14 761
insult	118 079
threat	4 725

TAB. 3 – Nombre de commentaires pour chaque sous-type de discours de haine.

Les ensembles d'apprentissage et de test utilisés sont les mêmes que ceux proposés lors du Kaggle [2]. On suppose que les distributions des étiquettes et des sous-groupes entre les deux sous-ensembles sont similaires mais non exactes.

Pour traiter le problème du déséquilibre de l'ensemble de données pendant l'entraînement, nous rééquilibrons les différentes classes. Pour ce faire, nous appliquons un ré-échantillonnage négatif : nous gardons seulement 10% des exemples choisis aléatoirement sans toxicité (les 6 classes de haine non activées : classe sur-représentée dans le jeu d'entraînement), et nous gardons tous les exemples avec au moins une des 6 classes activées. A la fin, il y a autant d'exemples avec toutes les classes négatives que d'exemples avec au moins une classe positive. Au total, la taille de l'ensemble d'entraînement est de 310 000 exemples. Il est important de noter qu'aucun rééquilibrage n'est effectué sur le sous-ensemble de test.

2.3 Modèles

La plupart des modèles Transformers utilisés ici sont basés sur BERT(Bidirectional Encoder Representations from Transformers). Google a proposé ce modèle en 2018. Il est composé uniquement d'un empilement de couches encodeurs des modèles Transformers. Le modèle BERT est également utilisé pour la classification, pour cela, il utilise un jeton spécifique `<CLS>` au début de chaque séquence. C'est un modèle qui a montré de très bonnes performances dans le domaine du traitement du langage en termes de résultat et vitesse de calcul par rapport aux autres modèles. C'est pour cette raison qu'on a choisi le modèle BERT comme modèle de référence pour notre comparaison.

Malgré les excellents résultats obtenus dans différents benchmarks, ce modèle présente certaines limites. Depuis la sortie de BERT, différents modèles ont été proposés pour répondre à ces limitations. Pour cette raison, nous allons aussi étudier les performances des variantes récentes de BERT sur la classification des commentaires toxiques : DistilBERT (Sanh et al., 2019), AlBERT (Lan et al., 2019), RoBERTa (Zhuang et al., 2021), XLM RoBERTa (Conneau et al., 2020), BERTweet (Nguyen et al., 2020), HateBERT (Caselli et al., 2021), XLNet (Yang et al., 2019) et Compact Convolutional Transformer (CCT) (Hassani et al., 2021).

DistilBERT a été proposé par Sanh et al. (2019). Il s'agit d'une version distillée (Hinton et al., 2014) du modèle BERT. Le nouveau modèle a 40% de paramètres en moins et est 60% plus rapide en préservant 95% des performances de BERT.

AlBERT (Lan et al., 2019), qui signifie "A Lite BERT", a été mis à disposition dans une version open source par Google en 2019. Le modèle a été construit avec la structure

originale de BERT, mais conçu pour réduire drastiquement les paramètres (de 89%) en utilisant le partage des paramètres à travers les couches cachées du réseau, et en factorisant la couche d'intégration. Tout cela a été accompli avec une réduction de la précision de 82,3% à 80,1% en moyenne sur une liste de jeux de données.

RoBERTa (Zhuang et al., 2021) est une amélioration du modèle BERT proposée par Facebook AI. Roberta est entraîné seulement sur une tâche de modélisation du langage masqué (MLM), avec un masquage dynamique, de sorte que les tokens masqués changent à chaque époque d'entraînement, sur une taille de lot plus grande et pendant plus longtemps.

XLM RoBERTa (Conneau et al., 2020) est une version multilingue de RoBERTa. Il est pré-entraîné sur 2.5 To de données CommonCrawl filtrées contenant 100 langues.

BERTweet (Nguyen et al., 2020) est un modèle basé sur BERT entraîné sur un énorme corpus de tweet anglais proposé par Nvidia en utilisant la même méthode que pour Roberta sur une tâche de modélisation du langage. Le corpus utilisé pour l'entraînement est d'environ 820 millions (80 Go) de tweets anglais. BERTweet s'est montré plus performant que le modèle de base Roberta dans les tâches suivantes liées aux tweets : L'étiquetage morpho-syntaxique, la reconnaissance d'entités nommées et la classification de textes.

HateBERT (Caselli et al., 2021) est un modèle publié lors de la conférence de l'Association for Computational Linguistics. Il utilise un modèle de base BERT pré-entraîné. Ce modèle a été fine-tuné pour une tâche de modélisation du langage sur le jeu de données RAL-E (Reddit Abusive Language English dataset). Ce jeu de données est composé de 1 492 740 phrases différentes provenant de Reddit et contient des discours haineux, des phrases offensantes et abusives. Le modèle a également été fine-tuné sur trois jeux de données différents : OfensEval, AbusEval et HatEval, battant ainsi l'état de l'art sur ces 3 jeux de données.

XLNet (Yang et al., 2019) est un modèle Transformer bidirectionnel de grande envergure qui utilise la permutation de tokens pour capturer des informations contextuelles et améliorer la précision des prédictions. XLNet a surpassé BERT dans 20 tâches, telles que la réponse à des questions, l'inférence en langage naturel, l'analyse de sentiments, etc.

Pour tous ces modèles, nous concaténons la sortie des 4 dernières couches du token `<CLS>` en un grand vecteur de caractéristiques et empilons deux couches denses pour prédire un vecteur de dimension 6 qui correspond aux 6 classes de haine à prédire. Des modèles pré-entraînés ont été utilisés, et les poids des modèles ont été dégelés pendant l'entraînement. De nombreuses recherches ont été effectuées concernant l'extraction de caractéristiques avec les modèles Transformers. Les résultats présentés dans (Devlin et al., 2019)) ont inspiré notre étude et notre comparaison. L'article montre que la concaténation des quatre dernières couches de l'encodeur donne de meilleurs résultats que l'utilisation de la seule dernière couche.

Compact Convolutional Transformer (CCT) (Hassani et al., 2021) est une architecture basée sur les Transformers, utilisée à la base pour de la vision par ordinateur que nous avons ré-utilisé sur du texte. L'article original montre que CCT peut donner de bons résultats sur des ensembles de données d'images et de textes avec moins de paramètres que les modèles basés sur des modèles Transformers. CCT combine des convolutions

et des couches d'attention en utilisant d'abord une tokenisation par convolution sur l'image contrairement aux ViT (Dosovitskiy et al., 2021) qui utilise une tokenisation par patch. L'entraînement a été effectué à partir de zéro, et nous avons utilisé un embedding GloVe pré-entraîné (Pennington et al., 2014) et enrichi pendant l'entraînement.

Global Vectors for Word Representation (GloVe) (Pennington et al., 2014) est un modèle utilisé pour trouver des vecteurs à partir de mots. Il utilise une matrice de co-occurrence pour prendre en compte le contexte global des mots dans la phrase. Les relations sémantiques entre les mots peuvent être extraites de la matrice de co-occurrence.

Pour comparer les modèles basés sur BERT avec d'autres modèles plus traditionnels, nous avons aussi entraîné un GRU bidirectionnel et un LSTM bidirectionnel à partir de zéro. Pour chacun d'entre eux, trois couches de RNN et une couche d'incorporation non gelée GloVe (Pennington et al., 2014) ont été utilisées.

Plusieurs ResNet (He et al., 2016) avec une profondeur de 44 et 56 couches ont également été entraînés à partir de zéro. Pour ces modèles, nous avons utilisé un embedding GloVe pré-entraîné. Dans certaines sessions d'entraînement, nous avons gelé l'embedding pour voir l'impact durant l'apprentissage.

3 Expérimentations

3.1 Entraînement

Tous les modèles [3] sont entraînés sur trois époques, avec une taille de lot de 32 exemples, sauf pour CCT, où nous nous limitons à 8 par lot en raison du manque de VRAM. Nous utilisons l'optimiseur AdamW, avec amsgrad : False, betas : (0.9, 0.999), eps : 1e-08, lr : 1e05, maximize : False et weight_decay : 0.01.

Nous utilisons l'entropie croisée binaire pondérée positive (pwBCE) comme fonction de perte. Cette fonction de perte ajoute des poids sur les échantillons positifs pour les considérer autant que les négatifs. Pour le modèle RoBERTa, nous utilisons trois autres fonctions de perte qui sont l'entropie croisée binaire (BCE), la perte focale (FL) (Lin et al., 2017), et la perte focale pondérée positive (pwFL). La FL réduit la perte attribuée aux exemples bien classés et se concentre sur les exemples mal classés généralement dû à un déséquilibre des classes. pwFL correspond à la même astuce que pour pwBCE appliquée à FL.

Pour mesurer les performances du modèle, nous utilisons des métriques similaires à celles utilisées lors de la compétition Kaggle [2] : Macro AUROC, Macro F1 et Micro F1 avec un seuil de 0.5, Précision et Rappel. Pour mettre en évidence la complexité du modèle, nous mesurons également le temps d'inférence. Le temps d'inférence moyen par lot est calculé à partir de 6 000 lots pendant la phase d'inférence sur le jeu de test.

Comme nous pouvons le voir dans le tableau 4, les modèles de détection des discours haineux pourraient faire des prédictions biaisées pour des identités particulières qui sont déjà la cible de tels abus. Pour mesurer ce biais involontaire du modèle, nous nous appuyons sur les mesures basées sur l'AUC développées par Borkan et al. (2019). Il s'agit de l'AUC du sous-groupe (Sub. AUC), de l'AUC du sous-groupe négatif par rapport aux positifs (BPSN) et de l'AUC du sous-groupe positif par rapport aux négatifs (BNSP).

3. Le code source est disponible sur github à l'adresse `https://github.com/Nigiva/hatespeech-detection-models`

Sub. AUC mesure l'AUROC pour chaque communauté en utilisant les messages toxiques et normaux de l'ensemble de test qui mentionnent la communauté considérée. Une valeur plus élevée signifie qu'un modèle est moins susceptible de confondre un message normal qui mentionne la communauté avec un message toxique qui ne le fait pas.

BPSN AUC mesure l'AUROC pour chaque communauté, en utilisant les messages normaux qui mentionnent la communauté et les messages toxiques qui ne mentionnent pas la communauté considérée. Une valeur plus élevée signifie qu'un modèle est moins susceptible de confondre un message normal qui mentionne la communauté avec un message toxique qui ne le fait pas.

BNSP AUC mesure l'AUROC pour chaque communauté en utilisant les messages toxiques qui mentionnent la communauté et les messages normaux qui ne mentionnent pas la communauté considérée dans l'ensemble de test. Une valeur plus élevée signifie que le modèle est moins susceptible de confondre un message toxique qui mentionne la communauté avec un message normal qui ne la mentionne pas.

Pour combiner ces métriques entre les différentes communautés, nous avons utilisé la moyenne généralisée (MG) ou moyenne de puissance avec comme exposant p, qui a déjà été utilisée par l'équipe Jigsaw/Conversation AI lors d'une compétition Kaggle [2]. Nous présentons donc **GMB-Subgroup-AUC**, **GMB-BPSN-AUC** et **GMB-BNSP-AUC** qui sont respectivement les moyennes généralisées de **Sub. AUC**, **BPSN AUC** et **BNSP AUC**.

Nous limitons l'évaluation au jeu de test uniquement. Cette restriction nous permet d'évaluer les modèles en termes de réduction des biais. Seules les communautés ayant plus de 500 exemples dans l'ensemble de test seront incluses dans l'évaluation.

4 Résultat

			Performance					Biais		
Type	Id	Nom	AUROC	Macro F1	Micro F1	Precision	Recall	GMB Sub.	GMB BPSN	GMB BNSP
BERT	0	AlBERT	0.9790	0.3463	0.4786	0.3247	0.9104	0.8674	0.8998	0.9513
	1	BERTweet	0.9816	0.3616	0.4928	0.3363	0.9216	0.8780	0.8945	0.9603
	2	DistilBERT	0.9804	0.3879	0.5115	0.3572	0.9001	0.8762	0.8740	**0.9644**
	3	HateBERT	0.9791	0.3679	0.4844	0.3292	0.9165	0.8744	0.8915	0.9589
	4	RoBERTa BCE	0.9813	**0.4749**	0.5359	0.3836	0.8891	0.8800	0.8901	0.9616
	5	RoBERTa FL	**0.9818**	0.4648	0.5524	0.4017	0.8839	**0.8807**	**0.9010**	0.9597
	6	RoBERTa pwBCE	0.9809	0.3541	0.4845	0.3284	0.9232	0.8741	0.8982	0.9575
	7	RoBERTa pwFL	0.9809	0.3612	0.4861	0.3297	0.9254	0.8734	0.8920	0.9597
	8	XLM RoBERTa	0.9790	0.3368	0.4680	0.3135	0.9230	0.8689	0.8859	0.9581
CCT	9	CCT	0.9505	0.3428	0.4874	0.3507	0.7983	0.8133	0.8307	0.9447
CNN	10	Freeze GloVe ResNet44	0.9526	0.4189	0.5591	0.4631	0.7053	0.8219	0.7876	0.9499
	11	Unfreeze GloVe ResNet44	0.9660	0.4566	**0.5958**	**0.4835**	0.7759	0.8421	0.8493	0.9540
	12	Unfreeze GloVe ResNet56	0.9639	0.3778	0.5098	0.3604	0.8707	0.8487	0.8445	0.9579
RNN	13	BiGRU	0.9748	0.3492	0.4762	0.3232	0.9036	0.8573	0.8616	0.9600
	14	BiLSTM	0.9754	0.3638	0.5089	0.3586	0.8761	0.8636	0.8758	0.9569
XLNet	15	XLNet	0.9800	0.3336	0.4586	0.3045	**0.9287**	0.8738	0.8834	0.9597

TAB. 4 – Résultats des performances des modèles.

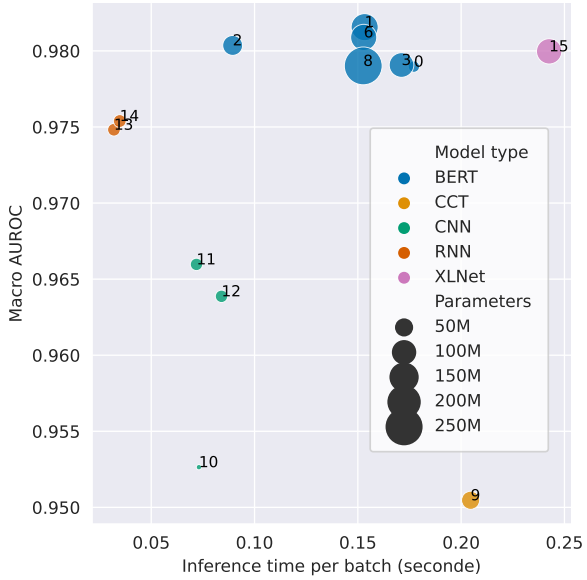

FIG. 1 – Performances du modèle, en fonction du temps d'inférence par lot et du nombre de paramètres entraînables. Les numéros correspondent à l'identifiant du modèle dans le tableau 4. Tous les modèles ont un lot de 32 échantillons, sauf CCT, qui utilise un lot de 8.

4.1 Performances

D'après la Figure 1 et le Tableau 4, BERT, RNN et XLNet ont en général de meilleurs scores AUROC que les autres. Comme les commentaires sont assez courts (27 tokens), les RNN parviennent à garder en mémoire la majorité du message, ce qui à notre avis les aide à faire de bonnes prédictions. Nous aurions probablement constaté un écart de performance plus important entre BERT et RNN si les commentaires avaient été plus longs. RoBERTa et les modèles de perte focale offrent les meilleures performances AUROC pour les biais les plus faibles. Tous les BERT, indépendamment de leur taille et de leurs optimisations, ont des performances très similaires : DistilBERT et AlBERT sont aussi performants qu'un HateBERT. Cependant, on note que l'ajout de la fonction de perte focale sur RoBERTa diminue les biais relatifs aux différentes classes de toxicité tandis que la version XML RoBERTa ne se distingue pas des autres : la pluralité des langages appris ne semble pas améliorer les performances quant à la détection de commentaires haineux.

Pour tous les modèles, le rappel est souvent très élevé, malheureusement la précision reste faible. En d'autres termes, les modèles sont très sensibles aux commentaires haineux mais génèrent plus de faux positifs.

A propos des BERT, si nous examinons les RoBERTa avec les différentes pertes d'apprentissage (BCE, pwBCE, FL, pwFL), nous constatons des scores relativement proches à la fin. Aucune des pertes d'apprentissage testées n'améliore l'apprentissage des modèles par rapport à un simple BCE. Nous notons même que les poids positifs (pwBCE et pwFL) obtiennent de

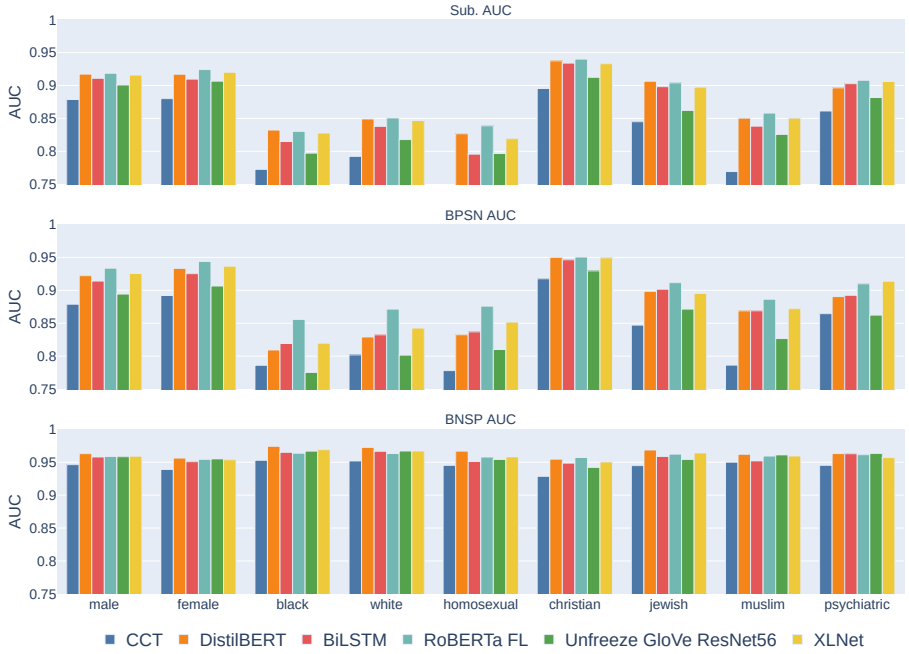

FIG. 2 – Résultats de la communauté pour chaque mesure de biais sur la classe de toxicité. Dans un souci de lisibilité, nous n'avons gardé que les modèles BERT, RNN, et CNN ayant les meilleures performances sur AUROC ou sur la métrique de biais AUC.

moins bons scores F1 que BCE ou FL, mais le rappel est 0.04 plus élevé et la précision est inférieure de 0.1.

Le Bi-GRU et le Bi-LSTM ont des performances équivalentes en termes d'AUROC et de scores F1. Enfin, nous remarquons que tous les modèles ont un peu plus de difficultés à classer les commentaires toxiques et les insultes que les commentaires sexuels explicites.

4.2 Biais

Dans l'ensemble, si nous examinons les résultats présentés dans le tableau 4, nous constatons que les modèles ont un BNSP GMB supérieur à 0.95. En d'autres termes, les modèles n'ont aucune difficulté à différencier les commentaires haineux ciblant une communauté des commentaires génériques (sans cibler une identité particulière). Au contraire, on observe que les scores de GMB BPSN et GMB Sub sont plus faibles que ceux de GMB BNSP qui sont pour leur part souvent inférieurs à 0.90. Ainsi, tous les modèles présentent un biais d'association entre les identités et les insultes. Ils auront tendance à annoter les commentaires positifs sur une communauté comme étant des insultes. Mais ce biais dépend du type de modèle.

D'après le tableau 4, nous voyons que les modèles BERT et RNN sont généralement moins sensibles à ce biais en ayant des GMB BPSN et GMB Sub légèrement plus élevés. En revanche, les modèles basés sur la convolution, tels que CNN et CCT ont tendance à être plus sensibles

à ce biais d'association. On peut expliquer ce comportement par la capacité qu'ont les CNN à capturer des motifs avec des convolutions.

D'après la figure 2, tous les modèles obtiennent en moyenne de moins bons résultats sur BPSN et Sub. AUC pour les communautés `black`, `homosexual`, `muslim`, et `white` par rapport aux autres communautés. Pour les BPSN, cela signifie que les modèles ont du mal à différencier les insultes qui ne visent pas de population en particulier et commentaires neutres à propos d'une communauté. Ainsi, ces mêmes modèles auront tendance à avoir un biais d'association plus important. Pour Sub. AUC, cela signifie que lorsqu'un commentaire cible une communauté telle que `black`, `gay`, `muslim`, ou `white`, les modèles auront simplement plus de difficultés à distinguer les commentaires haineux des commentaires non haineux.

Si nous examinons maintenant plus en détail chaque modèle et chaque identité, nous remarquons à nouveau que *RoBERTa with FL*, *BiLSTM*, and *XLNet* sont moins affectés par ce type de biais que *Unfreeze GloVe ResNet56* et *CCT*. Il y a même une différence de 0.05 sur la Sub. AUC pour les commentaires ciblant des communautés telles que `jewish` ou `muslim` entre *RoBERTa with FL* et *Unfreeze GloVe ResNet56*. De même, il existe une différence allant jusqu'à 0.1 sur l'AUC BPSN pour les communautés `black`, `homosexual` et `muslim`. Cela montre que pour ces communautés, qui sont particulièrement ciblées par des commentaires haineux, les modèles BERT, RNN et XLNet sont moins sujets aux biais d'association que le CNN et le CCT.

4.3 Temps d'inférence

D'après la figure 1, avec des performances à peine moindres que celles des modèles de type BERT et XLNET, les RNNs ont un temps d'inférence par lot au moins 5 à 8 fois inférieur. DistilBERT présente des performances 2 fois supérieures à celle de Bi-GRU et Bi-LSTM, et ce, avec le plus petit temps d'inférence testé dans notre étude, même si la différence de performance est seulement de 0.005 en AUROC.

Le CNN se retrouve avec un temps d'inférence plus court que la plupart des BERT et plus grand que le plus long RNN testé, mais avec des performances bien inférieures à celles des RNN ou des BERT. Avec le même temps d'inférence par lot, DistilBERT présente de meilleures performances.

Nous remarquons également qu'en figeant nos projections, les performances de nos modèles sont réduites sans impact notable sur leur temps d'inférence.

Enfin, le CCT offre des performances décevantes avec un temps d'inférence par lot considérable, surtout quand on sait que nous avons réduit la taille des lots de 32 à 8 pour ce modèle particulier.

5 Conclusions et Perspectives

Tous les BERT ont des performances très similaires, quelle que soit leur taille, les optimisations ou le langage utilisé pour les pré-entraîner. Plus largement, les BERT, RNN et XLNet ont des performances presque semblables. Les RNNs sont beaucoup plus rapides à l'inférence que tous les BERTs testés et restent un bon compromis entre performance et temps d'inférence pour la détection multi-label de commentaires haineux. RoBERTa et les modèles avec une fonction de perte focale offre les meilleures performances sur l'AUROC et le minimum de

biais. Enfin, DistilBERT combine à la fois de bonnes performances de classification et un faible temps d'inférence par lot. Ce résultat nous encourage à développer des modèles Transformers plus petits et donc plus rapide en termes de temps de calcul pour prendre en considération les contraintes de production.

Même si les modèles sont tous affectés par le biais d'association des identités à la toxicité, BERT, RNN et XLNet y sont moins sensibles que CNN et CCT. Ceci nous oblige à être très prudents dans la mise au point d'un système de détection de discours haineux pour éviter ce genre de biais.

Références

Borkan, D., L. Dixon, J. Sorensen, N. Thain, et L. Vasserman (2019). Nuanced metrics for measuring unintended bias with real data for text classification. pp. 491–500.

Caselli, T., V. Basile, J. Mitrović, et M. Granitzer (2021). HateBERT : Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, Online, pp. 17–25. Association for Computational Linguistics.

Conneau, A., K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, et V. Stoyanov (2020). Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online, pp. 8440–8451. Association for Computational Linguistics.

Devlin, J., M.-W. Chang, K. Lee, et K. Toutanova (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Vol. 1*, Minneapolis, Minnesota, pp. 4171–4186.

Dixon, L., J. Li, J. Sorensen, N. Thain, et L. Vasserman (2018). Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '18, New York, NY, USA, pp. 67–73.

Dosovitskiy, A., L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, et N. Houlsby (2021). An image is worth 16x16 words : Transformers for image recognition at scale.

Hassani, A., S. Walton, N. Shah, A. Abuduweili, J. Li, et H. Shi (2021). Escaping the big data paradigm with compact transformers. *CoRR abs/2104.05704*.

He, K., X. Zhang, S. Ren, et J. Sun (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.

Hinton, G., O. Vinyals, et J. Dean (2014). Distilling the knowledge in a neural network. In *Deep Learning and Representation Learning Workshop, NIPS*.

Ibrohim, M. O. et I. Budi (2019). Multi-label hate speech and abusive language detection in Indonesian Twitter. In *Proceedings of the Third Workshop on Abusive Language Online*, Florence, Italy, pp. 46–57. Association for Computational Linguistics.

Lan, Z., M. Chen, S. Goodman, K. Gimpel, P. Sharma, et R. Soricut (2019). ALBERT : A lite BERT for self-supervised learning of language representations. *CoRR abs/1909.11942*.

Lee, Y., S. Yoon, et K. Jung (2018). Comparative studies of detecting abusive language on Twitter. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, Brussels, Belgium, pp. 101–106. Association for Computational Linguistics.

Lin, T., P. Goyal, R. Girshick, K. He, et P. Dollar (2017). Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, Los Alamitos, CA, USA, pp. 2999–3007. IEEE Computer Society.

Liu, P., W. Li, et L. Zou (2019). NULI at SemEval-2019 task 6 : Transfer learning for offensive language detection using bidirectional transformers. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, Minneapolis, Minnesota, USA, pp. 87–91.

Nguyen, D. Q., T. Vu, et A. Tuan Nguyen (2020). BERTweet : A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, pp. 9–14.

Pennington, J., R. Socher, et C. Manning (2014). GloVe : Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Qatar, pp. 1532–1543. Association for Computational Linguistics.

Sanh, V., L. Debut, J. Chaumond, et T. Wolf (2019). Distilbert, a distilled version of bert : smaller, faster, cheaper and lighter. In *NeurIPS EMC2 Workshop*.

Vaswani, A., N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, et I. Polosukhin (2017). Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, et R. Garnett (Eds.), *Advances in Neural Information Processing Systems*, Volume 30. Curran Associates, Inc.

Yang, Z., Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, et Q. V. Le (2019). *XLNet : Generalized Autoregressive Pretraining for Language Understanding*. Red Hook, NY, USA : Curran Associates Inc.
English

Zhuang, L., L. Wayne, S. Ya, et Z. Jun (2021). A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, Huhhot, China, pp. 1218–1227. Chinese Information Processing Society of China.

Summary

Toxic comment detection on social media has proven to be essential for content moderation. This paper compares a wide set of different models on a highly skewed multi-label hate speech dataset. We consider inference time and several metrics to measure performance and bias in our comparison. We show that all BERTs have similar performance regardless of the size, optimizations or language used to pre-train the models. RNNs are much faster at inference than any of the BERT. BiLSTM remains a good compromise between performance and inference time. RoBERTa with Focal Loss offers the best performance on biases and AUROC. However, DistilBERT combines both good AUROC and a low inference time. All models are affected by the bias of associating identities. BERT, RNN, and XLNet are less sensitive than the CNN and Compact Convolutional Transformers.