Normalisation lexicale de contenus générés par les utilisateurs sur les réseaux sociaux

Lydia Nishimwe

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

Normalisation lexicale de contenus générés par les utilisateurs sur les réseaux sociaux

Lydia Nishimwe
Inria Paris, 2 rue Simone IFF, 75012 Paris, France
lydia.nishimwe@inria.fr

RÉSUMÉ

L'essor du traitement automatique des langues (TAL) se vit dans un monde où l'on produit de plus en plus de contenus en ligne. En particulier sur les réseaux sociaux, les textes publiés par les internautes sont remplis de phénomènes « non standards » tels que les fautes d'orthographe, l'argot, les marques d'expressivité, etc. Ainsi, les modèles de TAL, en grande partie entraînés sur des données « standards », voient leur performance diminuer lorsqu'ils sont appliqués aux contenus générés par les utilisateurs (CGU). L'une des approches pour atténuer cette dégradation est la normalisation lexicale : les mots non standards sont remplacés par leurs formes standards. Dans cet article, nous réalisons un état de l'art de la normalisation lexicale des CGU, ainsi qu'une étude expérimentale préliminaire pour montrer les avantages et les difficultés de cette tâche.

ABSTRACT

Lexical normalisation of user-generated content on social media

The boom of natural language processing (NLP) is taking place in a world where more and more content is produced online. On social networks especially, textual content published by users are full of "non-standard" phenomena such as spelling mistakes, jargon, marks of expressiveness, etc. Thus, NLP models, which are largely trained on "standard" data, suffer a decline in performance when applied to user-generated content (UGC). One approach to mitigate this degradation is through lexical normalisation where non-standard words are replaced by their standard forms. In this paper, we review the state of the art of lexical normalisation of UGC, as well as run a preliminary experimental study to show the advantages and difficulties of this task.

MOTS-CLÉS : normalisation lexicale, contenus générés par les utilisateurs (CGU), réseaux sociaux, modèles de langue.

KEYWORDS: lexical normalisation, user-generated content (UGC), social media, language models.

1 Introduction

Pour développer des systèmes de traitement automatique des langues (TAL) capables de traiter les « contenus générés par les utilisateurs » (CGU), il est nécessaire de se pencher soit sur les moyens de rendre les modèles robustes aux variations linguistiques associés à ces contenus, soit sur la normalisation de ces contenus afin qu'ils ressemblent le plus possible à la langue standard sur laquelle ces modèles sont généralement entraînés. Dans cet article, nous étudions la seconde de ces deux approches et nous consacrons ainsi à la tâche de normalisation lexicale des CGU, qui consiste à remplacer les formes non standard par leurs variantes standard (« normalisées »).

Actes des 16e Rencontres Jeunes Chercheurs en RI (RJCRI) et 25e Rencontre des Étudiants Chercheurs en Informatique pour le Traitement Automatique des Langues (RÉCITAL), pages 160–183.

Nous commençons par un état de l'art du domaine : nous décrivons d'abord les spécificités des CGU et les problèmes qu'ils posent pour les systèmes de TAL (section 2). Nous détaillons ensuite les méthodes proposées dans la littérature (section 3.1) pour la normalisation des CGU, mais également pour des tâches connexes telles que la normalisation phonétique, la correction post-OCR, la correction grammaticale et la normalisation des variantes dialectales ou historiques (section 3.2). Nous poursuivons avec un bref panorama des jeux de test et des métriques pour la tâche en question (section 3.3). Nous proposons ensuite une étude expérimentale préliminaire (section 4) dont le but est d'illustrer certaines des difficultés de la tâche en termes de modélisation et d'évaluation. La méthode que nous avons choisie repose sur un processus en deux étapes : (i) la détection supervisée de tokens non standards modélisée comme une tâche d'étiquetage de séquences, (ii) la normalisation des tokens détectés lors de la première étape à l'aide d'une approche couplant modélisation de langue par masquage (*masked language modelling*) et distance d'édition. L'idée sous-jacente est de choisir une forme standard qui soit à la fois appropriée en contexte et formellement similaire au mot non standard d'origine.

Nos expériences explorent le poids relatif à donner à la distance d'édition, en comparant plusieurs modèles de langue par masquage, dont certains entraînés sur des données CGU non standards. Nous menons nos évaluations au moyen de plusieurs métriques automatiques, y compris des métriques qui s'appuient sur la précision et des métriques développées pour la traduction automatique, qui ont été utilisées précédemment dans la littérature. Notre évaluation nous permet de constater que l'évaluation de la normalisation des CGU est loin d'être simple. L'approche que nous testons présente des limites évidentes, dont certaines sont mal prises en compte par les métriques usuelles.

2 Le TAL et les CGU : une relation amour-haine

2.1 Les CGU sur les réseaux sociaux

Sproat *et al.* (2001) ont utilisé le terme de « mots non standards » (*non-standard words, NSW*) pour décrire des mots et symboles (chiffres, abréviations, dates, devises monétaires, acronymes) qui ne se trouvent pas dans un dictionnaire, ou dont la prononciation ne peut se déduire des règles usuelles [1].

Avec l'expansion des messages textuels envoyés par téléphone (*Short Message Service, SMS*) au tournant du XXI$^{\text{ème}}$ siècle, d'autres phénomènes non standards sont apparus dans les textes écrits : la simplification de l'orthographe (p. ex. la suppression d'accents) et de la grammaire (p. ex. l'omission de pronoms), la substitution phonétique (`a 2m1` pour `à demain`), l'utilisation d'émoticônes, etc. Alors que les mots non standards étaient considérés comme grammaticalement corrects, ces nouveaux phénomènes n'étaient pas encore formalisés en linguistique (Aw *et al.*, 2006).

Après les SMS, les textes non standards ont connu un essor sur les réseaux sociaux, les forums de discussion, les tchats et d'autres plate-formes où les internautes interagissent. Cela a marqué l'émergence des CGU [2], qui ont été largement qualifiés de « bruités » dans le domaine du TAL. Pour

1. D'autres termes similaires employés dans la littérature sont : « mots mal formés » (*ill-formed words*, Han & Baldwin (2011)) ou encore « tokens non standards » (*non-standard tokens*, Liu *et al.* (2012)).

2. Une meilleure traduction serait « contenus *produits* par les utilisateurs », mais *générés* est globalement accepté par symétrie à l'anglais *user-generated content (UGC)*. D'autres appellations rencontrées dans la littérature sont : « langage texto » (*texting language*, Choudhury *et al.* (2007)), « textes bruités » (Formiga & Fonollosa, 2012) ou encore « textes bruités générés par les utilisateurs » (*noisy user-generated text*) (Baldwin *et al.*, 2015).

quantifier cette affirmation, Baldwin *et al.* (2013) ont mené une étude linguistique et statistique sur un corpus de CGU provenant de sources différentes et ont démontré qu'il était effectivement plus « bruité » qu'un corpus composé de textes standards édités. Par ailleurs, Eisenstein (2013) a expliqué des raisons fréquentes pour lesquelles les utilisateurs écrivent « si mal », à savoir : l'illettrisme, le nombre de caractères limité (p. ex. Twitter), le système de saisie du texte (clavier externe *versus* clavier tactile avec autocomplétion), des phénomènes pragmatiques, et certaines variables sociales.

Certains mots non standards présents dans les CGU sont propres aux réseaux sociaux utilisés, comme les hashtags (`#JeuxOlympiques`), les mentions (`@gouvernementFR`) et leur méta-langage (`RT` pour `Retweet`). De plus, l'argot évolue constamment et il y a toujours de nouveaux mots émergents (p. ex. des néologismes comme `burka + bikini` → `burkini`), ou encore l'utilisation du *leetspeak* pour censurer des jurons ou des propos offensifs (`!d10t` pour `idiot`). Un autre phénomène omniprésent est l'emploi de mots empruntés d'autres langues ou même le mélange de plusieurs langues (l'alternance codique).

Faire une liste exhaustive de tous les phénomènes non standards spécifiques aux CGU n'est pas une tâche aisée, cependant quelques tentatives ont été faites. Par exemple, Seddah *et al.* (2012) ont proposé une classification des phénomènes CGU rencontrés dans des forums de discussion et réseaux sociaux français. Ils les ont définis selon trois axes : (1) les phénomènes ergographiques qui visent à simplifier l'écriture, par exemple l'omission d'accents, la phonétisation, et certaines fautes d'orthographe (`son` pour `sont`) ; (2) les phénomènes transversaux comme la contraction (`nimp` pour `n'importe quoi`) et la segmentation typographique (`c a dire` pour `c'est-à-dire`) ; (3) les marques d'expressivité comme l'étirement des graphèmes ou de ponctuation (`superrr !!!`) et les émoticônes. Sanguinetti *et al.* (2020) se sont appuyés sur cette classification et y ont rajouté les phénomènes d'autocensure, ainsi qu'un quatrième axe des phénomènes d'influence de langues étrangères comme la translitération, la formation de nouveaux verbes et l'autocorrection. Par ailleurs, van der Goot *et al.* (2018) ont élaboré une taxonomie des spécificités CGU en anglais. Ils ont considéré trois types d'« anomalies » : (1) les anomalies non intentionnelles comme les fautes typographiques, orthographiques ou de segmentation ; (2) les anomalies intentionnelles telles que les abréviations d'expressions (`mdr` pour `mort de rire`), les répétitions, les contractions, les transformations phonétiques et l'argot ; (3) les anomalies inconnues.

2.2 L'impact négatif de CGU sur le TAL

Les modèles de TAL étant généralement entraînés sur des données standards, ils s'attendent à traiter des données du même type. En présence de phénomènes CGU, la performance de plusieurs tâches de TAL est négativement affectée, à savoir : l'analyse syntaxique (Foster, 2010 ; Seddah *et al.*, 2012), la détection de thèmes (*topic detection*) (Muñoz-García *et al.*, 2012), la tokénisation (Aminian *et al.*, 2012), la reconnaissance d'entités nommées (Moon *et al.*, 2018), l'analyse des dépendances (van der Goot & van Noord, 2018 ; van der Goot, 2019a), la traduction automatique (Belinkov & Bisk, 2017 ; Michel & Neubig, 2018 ; Rosales Núñez *et al.*, 2021a), l'analyse de sentiment (Kumar *et al.*, 2020), etc.

3 La normalisation lexicale : le chevalier blanc ?

Dans leur analyse statistique et linguistique des corpus CGU, Baldwin *et al.* (2013) ont montré que l'application de certaines tâches de TAL, comme l'identification de la langue, la normalisation lexicale et l'étiquetage morphosyntaxique, peuvent réduire le niveau de bruit dans ces corpus. Pour pallier la dégradation de performance des modèles de TAL causée par la présence de phénomènes CGU, Eisenstein (2013) a recensé deux approches principales : (1) la normalisation, qui vise à adapter les données à ce que les modèles attendent, et (2) l'adaptation de domaine, qui consiste à adapter les modèles aux données, par exemple en entraînant sur des données réelles CGU (Nguyen *et al.*, 2020) ou encore à entraîner les modèles sur des données bruitées synthétiques (Karpukhin *et al.*, 2019). Une autre approche consiste à changer l'architecture du modèle, par exemple en passant à l'échelle des caractères (Riabi *et al.*, 2021; Rosales Núñez *et al.*, 2021b), ou des segments de phrases (Rosales Núñez *et al.*, 2019a) pour la traduction automatique.

L'approche privilégiée est restée la normalisation lexicale, utilisée en amont d'autres tâches de TAL. Elle a fortement amélioré la performance dans plusieurs tâches : la reconnaissance d'entités nommées simples (Nguyen *et al.*, 2016) ou imbriquées (Plank *et al.*, 2020), l'étiquetage morphosyntaxique (van der Goot *et al.*, 2017; van der Goot & Çetinoğlu, 2021), l'analyse de dépendances (van der Goot *et al.*, 2020), ou encore la compréhension des CGU par des locuteurs non natifs (Ehara, 2021).

3.1 Méthodes

Approches modulaires L'une des approches classiques est de concevoir un système qui combine plusieurs modules, soit pour normaliser différents types de mots non standards, soit pour aborder la normalisation sous plusieurs angles (orthographe, phonétique, modèle de langue, ...). Par exemple, Sproat *et al.* (2001) ont implémenté un système qui, après avoir tokénisé le texte, classifie les tokens par un arbre de décision. Ensuite, il crée un treillis de mots à partir des étiquettes de classe, et un modèle n-grammes en déduit le meilleur candidat de normalisation. Liu *et al.* (2012) ont introduit un modèle à 4 composantes : un module de transformation de lettres, un module d'amorçage visuel, un correcteur orthographique, et un module qui combine les meilleurs candidats proposés par les 3 autres modules. Ahmed (2015) a aussi proposé une approche qui, dans un premier temps, génère un ensemble de candidats sur base d'une distance d'édition. Ensuite, cet ensemble est raffiné par une technique de similarité phonétique. Simultanément, l'algorithme de Peter Norvig pour la correction orthographique [3] est aussi appliqué sur ces candidats, et les deux résultats sont comparés. Si les deux méthodes produisent le même candidat, alors la normalisation est terminée. Sinon, un modèle 5-grammes est appliqué sur les contextes phonétiques pour sélectionner le meilleur candidat. Melero *et al.* (2016) ont proposé une stratégie qui utilise un correcteur orthographique pour détecter les mots non standards et générer des candidats, et un module de sélection constitué d'une interpolation linéaire de 4 modèles de langue encodant des informations linguistiques différentes (vraie casse, minuscules, lemmes, morphosyntaxe). van der Goot & van Noord (2017) ont conçu MoNoise, un modèle modulaire qui comprend, entre autres, un correcteur orthographique et un modèle de plongements de mots. Ils ont entraîné une forêt aléatoire pour sélectionner la meilleure normalisation parmi les candidats générés par les différents modules. van der Goot (2019b) a implémenté des améliorations au modèle MoNoise, et a démontré qu'il obtient la meilleure performance (à l'époque) de normalisation lexicale sur plusieurs langues. van der Goot (2021) a ensuite implémenté une version de MoNoise qui effectue

3. http://norvig.com/spell-correct.html

un transfert entre les différentes langues : le modèle est entraîné sur des données monolingues et annotées dans la langue source, qui sont remplacées par des données monolingues dans la langue cible pendant l'inférence.

Approches statistiques Choudhury *et al.* (2007) ont développé un modèle bigramme à base d'un modèle de Markov caché (MMC) pour corriger les erreurs dans le langage texto ; Xu *et al.* (2015) se sont appuyés sur cette approche et l'ont adapté au chinois en proposant un modèle à base de champs aléatoires conditionnels (*Conditional Random Fields, CRF*) pour segmenter les mots non standards en syllabes. Han & Baldwin (2011) ont implémenté une stratégie en trois temps. D'abord, pour chaque mot hors vocabulaire, un ensemble de candidats est généré selon des variations morphophonémiques. Ensuite, un classifieur détecte si le mot est « mal formé » à partir de cet ensemble, et le meilleur candidat de normalisation est sélectionné lors de la dernière étape. Supranovich & Patsepnia (2015) ont proposé un système à 2 composantes : un modèle à base de CRF pour détecter les mots bruités, et une étape de normalisation qui remplace les mots détectés par leurs variantes dans le lexique. Plus récemment, Jiang *et al.* (2022) ont introduit une approche de normalisation lexicale à grande échelle qui utilise des familles LSH (*Locality-Sensitive Hashing*) pour générer des candidats en se basant sur la morphologie des mots.

Méthodes à base de règles Clark & Araki (2011) ont proposé une méthode qui recherche les mots non standards dans une base de données et les remplace selon des règles définies. Baranes & Sagot (2014) ont développé une approche qui repose sur un système d'induction par analogie des règles apprises sur un corpus de fautes lexicales annotées en français. Pour la normalisation de tweets en espagnol, Ruiz *et al.* (2014) ont combiné des règles dans une étape de pré-traitement avec un modèle de distance d'édition et un modèle n-grammes pour sélectionner les candidats. Par ailleurs, Cotelo *et al.* (2015) ont proposé un schéma modulaire dont un module de calcul de distance d'édition, et d'autres modules à base de règles selon le type de mots hors vocabulaire. D'autre part, Cerón-Guzmán & León-Guzmán (2016) ont implémenté un modèle qui utilise des transducteurs finis pour générer des ensembles de candidats à partir de règles graphémiques et phonémiques. Beckley (2015) a implémenté une architecture simple qui consiste en une liste de substitutions obtenues de manière semi-supervisée, quelques composantes à base de règles, et un algorithme de Viterbi (Viterbi, 1967) sélectionnant le meilleur candidat. Kogkitsidou & Antoniadis (2016) ont proposé un modèle hybride pour la normalisation de SMS qui, d'une part, produit une représentation intermédiaire du message par l'application de grammaires locales et, d'autre part, utilise un modèle de traduction automatique à base de règles pour convertir cette représentation vers une forme standard.

La normalisation vue comme une tâche de traduction Aw *et al.* (2006) ont adapté un modèle de traduction statistique à base de segments pour « traduire de l'anglais des SMS vers l'anglais standard ». Pour la normalisation de SMS en français, Kobus *et al.* (2008) ont aussi utilisé un tel modèle de traduction. Afin d'améliorer sa performance, ils l'ont combiné avec un module inspiré de la reconnaissance automatique de la parole pour proposer des hypothèses pour les mots hors vocabulaires, et un modèle de langue pour sélectionner le meilleur candidat. Par ailleurs, Pennell & Liu (2011) ont implémenté un modèle de traduction à base de caractères pour normaliser spécifiquement les abréviations dans les SMS. Li & Liu (2012) ont proposé de combiner un correcteur orthographique avec un modèle de traduction à base de blocs de caractères générés selon des règles phonétiques chinoises. Formiga & Fonollosa (2012) ont utilisé un modèle de traduction à base de caractères pour traduire le texte « bruité » en texte « propre » en amont d'une tâche de traduction de l'anglais vers l'espagnol. Ce premier modèle produit un treillis de variantes orthographiques qui est passé en entrée d'un modèle de traduction bilingue à base de segments. Matos Veliz *et al.* (2019) ont évalué deux modèles de traduction automatique, statistique et neuronale, pour la normalisation de divers CGU en

anglais et en néerlandais. Ils ont conclu que, pour la traduction statistique, il est mieux d'entraîner le modèle de langue sous-jacent sur un corpus issu d'un domaine similaire à celui des UGC et que, pour la traduction neuronale, il est préférable d'ajouter plus de données d'entraînement que de les augmenter artificiellement. Ils ont aussi proposé d'envisager une approche modulaire pour le modèle statistique, et une technique d'augmentation de données basée sur des règles pour le modèle neuronal.

Approches par apprentissage profond Tiwari & Naskar (2017) ont proposé un modèle encodeur-décodeur de réseaux de neurones récurrents (*Recurrent Neural Network, RNN*) à mémoire court et long terme (*Long Short-Term Mermory, LSTM*) avec un méchanisme d'attention, et ils ont aussi créé des données artificielles pour entraîner ce modèle. Lourentzou *et al.* (2019) ont introduit un modèle hybride encodeur-décodeur à base de mots et de caractères, la composante à base de caractères étant entraîné sur des exemples antagonistes synthétiques. Stewart *et al.* (2019) ont utilisé un modèle de réseaux de neurones récurrents à portes (*Gated Recurrent Unit, GRU*) au niveau des mots et ont présenté de meilleures performances que les modèles au niveau des caractères et d'autres méthodes par apprentissage profond existantes.

Modèles détecteur-correcteur Une autre approche consiste à découpler la détection des mots non standards de leur normalisation lexicale. Par exemple, Leeman-Munk *et al.* (2015) ont utilisé deux modèles de réseaux de neurones à propagation avant (*Feed-forward Neural Network, FFNN*) augmentés : un « signaleur » pour identifier les tokens à normaliser et un « normalisateur » qui corrige un token à la fois. Par ailleurs, Tian *et al.* (2017) ont proposé un modèle de réseaux de neurones convolutifs (*Convolutional Neural Network, CNN*) à base de caractères pour la détection de mots non standards dans les tweets. Cette étape de détection est prévue en amont d'une normalisation lexicale.

Modèles de langue pré-entraînés Muller *et al.* (2019) ont apporté des modifications à l'architecture de BERT (Devlin *et al.*, 2019) et l'ont affiné pour la normalisation lexicale en tant que tâche de prédiction de tokens. Par ailleurs, Scherrer & Ljubešić (2021) ont proposé un système basé sur un modèle BERT affiné pour la classification de tokens qui prédit le type de transformation nécessaire pour corriger le mot non standard. Un modèle de traduction automatique à base de caractères est utilisé pour appliquer les corrections proposées par BERT. De plus, Bucur *et al.* (2021) ont aussi considéré la normalisation lexicale comme une tâche de traduction et ont proposé un modèle séquentiel au niveau de la phrase basé sur mBART (Liu *et al.*, 2020). Kubal & Nagvenkar (2021) ont quant à eux affiné un modèle BERT multilingue (Devlin *et al.*, 2019) pour la normalisation lexicale comme une tâche d'étiquetage de séquences et l'ont combiné avec une technique d'alignement de mots. Ainsi, ils ont pu utiliser le même modèle pour effectuer la normalisation sur plusieurs langues. van der Goot & Çetinoğlu (2021) ont aussi utilisé un modèle BERT multilingue dans la normalisation lexicale de CGU présentant de l'alternance codique, notamment pour l'identification de langue.

3.2 Tâches connexes

Nous avons vu que la normalisation lexicale peut être assimilée à une tâche de traduction d'une version non standard d'une langue vers sa version standard. En effet, Kobus *et al.* (2008) ont catégorisé les approches de normalisation lexicale de SMS selon trois « métaphores » : la correction orthographique, la traduction et la reconnaissance de la parole. De même, il existe aussi d'autres tâches qui sont plus ou moins connexes à la normalisation lexicale et qui peuvent lui inspirer d'autres approches, à savoir :

La normalisation phonétique Elle peut être considérée comme une sous-tâche de la normalisation lexicale puisque certains des phénomènes non standards observés sont en effet d'ordre phonétique.

D'ailleurs, certaines méthodes décrites dans la section 3.1 intègrent un module de calcul de similarité phonétique. Rosales Núñez *et al.* (2019b) ont proposé un modèle de normalisation phonétique pour améliorer la traduction des CGU du français vers l'anglais. Cette tâche est particulièrement utile pour normaliser les CGU dans des langues riches en homophonies comme le chinois (Qin *et al.*, 2021). Elle a aussi été appliquée à la correction orthographique dans les moteurs de recherche du commerce en ligne (Yang *et al.*, 2022).

La correction post-OCR et post-ASR Les textes résultant de la reconnaissance optique de caractères (*Optical Character Recognition, OCR*) doivent souvent être corrigés en post-traitement car ils contiennent des caractères mal reconnus et donc des mots non standards. De même, les transcriptions résultant de la reconnaissance automatique de la parole (*Automatic Speech Recognition, ASR*) contiennent des mots non standards provenant des phonèmes mal compris.

La correction d'erreurs grammaticales En contrepartie de la tâche normalisation lexicale, qui vise à corriger les erreurs d'ordre lexical, la correction grammaticale vise à corriger les erreurs d'ordre grammatical. Elle est aussi souvent découpée en deux sous-tâches : détection et correction. En pratique, la frontière entre erreur lexicale et erreur grammaticale n'est pas bien définie dans les CGU : certains phénomènes peuvent appartenir aux deux classes. Cependant, les annotateurs de données de normalisation lexicale essaient de se limiter à corriger les mots non standards d'un point de vue lexical, même si la phrase résultante reste agrammaticale.

La normalisation de variantes linguistiques En comparant grossièrement le langage non standard des CGU à un « dialecte » du langage standard, la tâche de normalisation lexicale peut être assimilée à celle de la normalisation de variantes linguistiques. En particulier, certains travaux sur la normalisation de dialectes (Partanen *et al.*, 2019) et des créoles (Liu *et al.*, 2022), de textes produits par des locuteurs non natifs (Sarkar *et al.*, 2020; Alam & Anastasopoulos, 2020), et de langue non contemporaine (Bawden *et al.*, 2022) peuvent s'avérer intéressants pour la normalisation lexicale.

La simplification de textes Un parallèle peut être établi entre cette tâche et la normalisation lexicale si nous considérons que le lexique non standard des CGU, difficile à comprendre en dehors d'un public restreint, doit être renvoyé vers un lexique standard, plutôt facile à comprendre et accessible à un plus grand public.

3.3 Évaluation

Bien que la normalisation lexicale soit une bonne solution pour le problème des mots non standards dans les CGU, elle reste une tâche qui est difficile à évaluer en raison du manque de ressources annotées d'une part, et du manque d'homogénéité dans le choix des conventions d'annotation et des métriques utilisées.

Ressources Malgré l'abondance de CGU sur internet, peu de données parallèles annotées pour la normalisation lexicale sont disponibles [4]. Néanmoins, la campagne d'évaluation MultiLexNorm2021 (van der Goot *et al.*, 2021) comprend des données annotées en plusieurs langues issues d'autres campagnes d'évaluations, à savoir : en danois [DA] (Plank *et al.*, 2020), en allemand [DE] (Sidarenka *et al.*, 2013), en anglais [EN] (Baldwin *et al.*, 2015), en espagnol [ES] (Alegria *et al.*, 2013), en croate [HR] (Ljubešić *et al.*, 2017a), en indonésien-anglais [ID-EN] (Barik *et al.*, 2019), en italien [IT] (van der Goot *et al.*, 2020), en néerlandais [NL] (Schuur, 2020), en slovène [SL] (Erjavec *et al.*, 2017),

4. Bikaun *et al.* (2021) ont créé un bon outil d'annotation à cet effet.

en serbe [SR] (Ljubešić *et al.*, 2017b), en turc [TR] (Çolakoğlu *et al.*, 2019), et en alternance codique turc-allemand [TR-DE] (van der Goot & Çetinoğlu, 2021). D'autres données parallèles annotées sont disponibles en anglais et en néerlandais (De Clercq *et al.*, 2014), et en japonais (Higashiyama *et al.*, 2021). Il existe aussi des données parallèles, non pas pour la normalisation lexicale en soi, mais pour l'évaluation des tâches en aval comme la traduction de CGU (Michel & Neubig, 2018; Rosales Núñez *et al.*, 2019a, 2021a).

Métriques Plusieurs métriques ont été utilisées pour évaluer la tâche de normalisation lexicale : le taux d'erreur de mots (*Word Error Rate, WER*) (Sproat *et al.*, 2001; Kobus *et al.*, 2008; Matos Veliz *et al.*, 2019) ; le taux d'erreur de phrases (*Sentence Error Rate, SER*) (Kobus *et al.*, 2008) ; le taux de couverture (Liu *et al.*, 2012) ; l'exactitude, la précision, le rappel et la F-mesure (Baldwin *et al.*, 2015) ; la précision sur les mots hors vocabulaire (Alegria *et al.*, 2013) ; et le BLEU (Aw *et al.*, 2006; Kobus *et al.*, 2008; Han & Baldwin, 2011), qui est une métrique de traduction. van der Goot (2019b) a considéré que ces métriques sont soit trop complexes pour la tâche, soit difficiles à interpréter et à comparer. Il a donc préconisé l'utilisation du « taux de réduction de l'erreur » (*Error Reduction Rate, ERR*), qui correspond à l'exactitude normalisée par le nombre de mots remplacés :

$$\text{ERR} = \frac{\%\text{exactitude} - \%\text{mots non normalisés}}{100 - \%\text{mots non normalisés}} \tag{1}$$

Cette métrique a été définie par van der Goot (2019c) et utilisée dans la campagne d'évaluation MultiLexNorm2021 (van der Goot *et al.*, 2021). Elle permet de comparer la performance d'un modèle sur plusieurs jeux de données différents, voire plusieurs langues.

4 Étude expérimentale

Dans cette partie, nous allons réaliser une étude expérimentale préliminaire pour illustrer la normalisation lexicale de CGU décrite dans l'état de l'art précédent. Nous allons comparer des modèles de langue pré-entraînés pour montrer la difficulté de la tâche (il est difficile de faire mieux qu'un modèle de base qui ne fait rien !) et celle d'en trouver une métrique adéquate.

4.1 Données

Nous avons utilisé les données parallèles de la campagne d'évaluation LexNorm2015 (Baldwin *et al.*, 2015). Elles consistent en tweets publiés en anglais, alignés avec leurs normalisations lexicales.

Jst read a tweet **lol** and **l o v e** it
↓
just read a tweet **laughing out loud** and **love** it

FIGURE 1 – Un exemple de tweet en anglais avec sa normalisation lexicale.
(Traduction : *Je viens de lire un tweet, mort de rire, et j'adore !*)

La figure 1 est un exemple de tweet normalisé. Elle illustre les trois types de normalisations lexicales distingués par les organisateurs de la campagne d'évaluation :

— la normalisation **1-1** : un mot non standard est remplacé par un mot standard (`Jst` → `just`) ;

— la normalisation **1-N** : un mot non standard est remplacé par plusieurs mots standards (`lol` → `laughing out loud`);
— la normalisation **N-1** : une séquence non standard de mots ou de sous-mots est remplacée par un seul mot standard (`l o v e` → `love`).

Les données étaient déjà prétokénisées [5] (par espaces et ponctuation), tout en tenant compte des spécificités de Twitter (liens URL, hashtags, mentions). Ces dernières n'ont pas été modifiées lors de l'annotation. Par ailleurs, les normalisations effectuées étaient toutes en minuscules (`Jst` → `just`) [6].

Les données LexNorm2015 comprennent un jeu d'entraînement et un de test. Le tableau 1 résume les statistiques [7] de ces deux jeux de données : le nombre de tweets et le pourcentage de mots normalisés.

Remarque. Les tweets étant déjà tokénisés, nous appelons « mot » toute suite de caractères délimitée par un espace. Ainsi, la séquence `l o v e` comprend quatre mots. En revanche, elle correspond à une seule occurrence dans le nombre de normalisations N-1.

Jeu de données	**# tweets**	**% mots normalisés**	*dont*		
			% 1-1	*% 1-N*	*% N-1*
Entraînement	2950	8,85	*73,25*	*26,55*	*0,20*
Test	1967	9,40	*73,92*	*25,68*	*0,40*

TABLE 1 – Statistiques des données LexNorm2015.

4.2 Modèles

Le pourcentage de mots à normaliser étant inférieur à 10% pour les données d'entraînement et de test (cf. le tableau 1), nous avons opté pour un modèle détecteur-correcteur afin de cibler la normalisation lexicale uniquement sur les mots qui la nécessitent. En plus, notre approche combine deux autres des méthodes citées dans l'état de l'art : elle est basée sur des modèles de langue par masquage (MLM) pré-entraînés, et elle est modulaire (elle inclut une distance d'édition).

4.2.1 Le détecteur

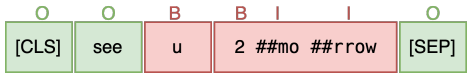
La détection de mots à normaliser peut être assimilée à une tâche de classification de tokens : pour chaque token de la phrase source, le modèle de détection doit prédire une étiquette qui correspond à sa classe d'appartenance (ici, standard ou non standard).

Concrètement, nous avons pris le MLM pré-entraîné BERT [8] (Devlin *et al.*, 2019) et nous l'avons affiné pour la tâche de classification de tokens sur le jeu d'entraînement de LexNorm2015. Nous avons utilisé le schéma d'étiquetage **B-I-O** : **B** (Beginning, *début*) pour étiqueter le premier token d'un mot non standard, **I** (Inside, *intérieur*) pour un token à l'intérieur d'un mot non standard, et **O** pour les tokens des mots standards.

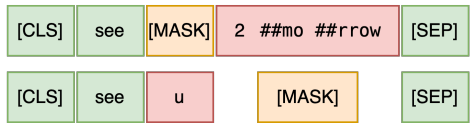
5. par le tokeniseur CMU-ARK (https://github.com/myleott/ark-twokenize-py).
6. Un choix discutable des annotateurs.
7. calculées par nous.
8. https://huggingface.co/bert-base-uncased

Prenons l'exemple de la phrase `see` **`u 2morrow`** (→ `see` **`you tomorrow`**, *à demain*). La figure 2a illustre sa tokenisation et son étiquetage B-I-O des tokens [9] par notre détecteur BERT.

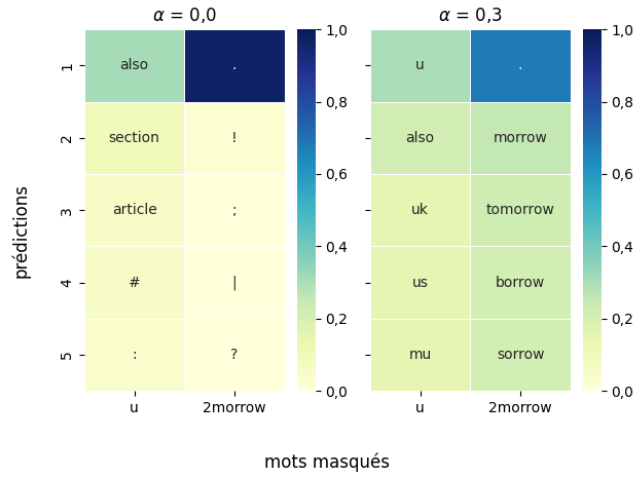
Remarque. Il est pertinent de noter qu'un « mot », tel que défini dans la section 4.1, peut être tokénisé en plusieurs tokens différents par chaque MLM. Ainsi, même si le détecteur dans la figure 2a a étiqueté quatre tokens comme non standards, ceux-ci correspondent seulement à deux mots à normaliser.



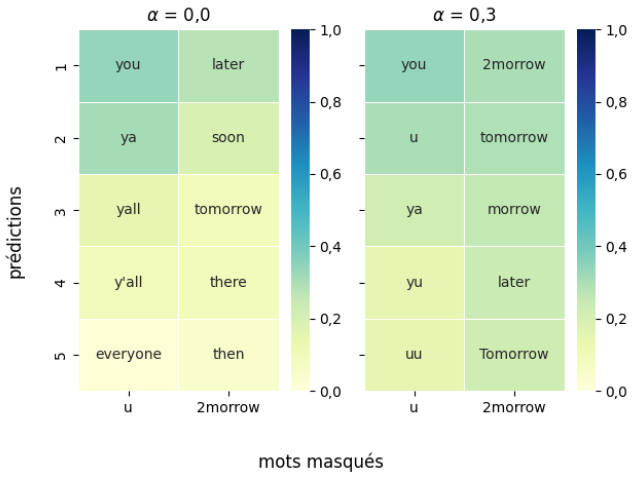
(a) Détection de mots à normaliser par un BERT affiné



(b) Stratégie de masquage des mots à normaliser



(c) BERT



(d) BERTweet

FIGURE 2 – Détection (a) et masquage (b) de mots à normaliser par un détecteur BERT. Prédictions des mots masqués par les correcteurs BERT (c) et BERTweet (d) pour $\alpha = 0{,}0$ et $0{,}3$, triées par score. (Normalisation attendue : `see` **`u 2morrow`** → `see` **`you tomorrow`**, *à demain*)

4.2.2 Le correcteur

Après avoir détecté les mots à normaliser, il reste à les corriger. Une approche consiste à masquer ces mots et à utiliser un MLM pré-entraîné pour les prédire selon le contexte environnant. Ainsi, en fonction du vocabulaire auquel le MLM a été exposé pendant son entraînement, les prédictions appartiendraient à un lexique standard et, par conséquent, les mots non standards seraient normalisés.

L'inconvénient de cette approche est le fait que les modèles de langue soient entraînés à prédire *un* token [10] qui convient au sens du contexte. Il peut donc y avoir plusieurs bonnes prédictions possibles d'un point de vue sémantique, mais aucune du point de vue lexical. Nous avons donc utilisé une distance d'édition pour guider le MLM vers une prédiction non seulement sémantiquement appropriée, mais aussi lexicalement proche du mot masqué.

Concrètement, nous avons défini un nouveau score, qui est une combinaison linéaire, de paramètre $\alpha \in [0, 1]$, entre le score du MLM et la distance de Damerau-Levenshtein (Damerau, 1964) [11]

9. Les tokens spéciaux `[CLS]` et `[SEP]` sont automatiquement considérés comme standards.
10. selon la tokénisation en sous-mots du MLM.
11. choisie car elle considère l'opération de transposition, contrairement à la distance de Levenshtein (Levenshtein, 1966).

normalisée sur une échelle de $[0, 1]$.

Soient $\mathbf{x} = (x_1 \dots x_\ell)$ la séquence source, x_i le token à normaliser, et $\bar{\mathbf{x}} = (x_1 \dots x_{i-1}, [\text{MASK}], x_{i+1} \dots x_\ell)$ la séquence source masquée au token x_i. Soient $\text{MLM}(\bar{\mathbf{x}})$ le vecteur de scores prédits par le modèle de langue pour tous les tokens de son vocabulaire $\mathcal{V}$, et $\text{Lev}_{\text{norm}}(x_i, \mathcal{V})$ le vecteur des distances de Damerau-Levenshtein normalisées entre x_i et tous les tokens du vocabulaire. Alors, la prédiction $\hat{x}_i$ du correcteur est :

$$\hat{x}_i = \arg\max_{\mathcal{V}} \quad [(1-\alpha)\text{MLM}(\bar{\mathbf{x}}) + \alpha(1 - \text{Lev}_{\text{norm}}(x_i, \mathcal{V}))] \tag{2}$$

Les MLM pré-entraînés que nous avons comparés pour la correction sont :

1. **BERT** : entraîné sur le corpus de livres BookCorpus et Wikipedia, tous des textes édités, donc « standards » ;
2. **RoBERTa** [12] (Liu *et al.*, 2019) : entraîné sur les mêmes données que BERT, en plus d'autres ressources éditées (CC-News, OpenWebText, Stories). Il a la même architecture que BERT mais avec des optimisations, et a été entraîné plus longtemps ;
3. **ELECTRA** [13] (Clark *et al.*, 2020) : ayant la même architecture que BERT et entraîné sur les mêmes données, mais avec un objectif d'entraînement différent : c'est un modèle discriminant entraîné à distinguer les tokens remplacés des tokens d'origine masqués ;
4. **BERTweet** [14] (Nguyen *et al.*, 2020) : identique à BERT, mais entraîné entièrement sur des données Twitter ;
5. **Twitter RoBERTa** [15] (Barbieri *et al.*, 2020) : le modèle RoBERTa pour lequel l'entraînement a été poursuivi sur des données Twitter.

Tous ces modèles ont été entraînés sur des données en anglais. Parmi les cinq, seuls BERTweet et Twitter Roberta ont été exposés à des CGU issus de Twitter ; nous les avons choisis sous l'hypothèse qu'ils seraient plus robustes au bruit des CGU. Le tableau 2 résume les propriétés de ces MLM.

Modèle	**Données d'entraînement**	**Taille du vocabulaire**
BERT	*BookCorpus, Wikipedia*	30 522
RoBERTa	= BERT + *CC-News, OpenWebText, Stories*	50 265
ELECTRA	= BERT	30 522
BERTweet	***Twitter***	64 000
Twitter RoBERTa	= RoBERTa + ***Twitter***	50 265

TABLE 2 – Propriétés des modèles de langue pour la correction.

Revenons à l'exemple de la phrase `see` **`u 2morrow`**. La figure 2b illustre la stratégie de masquage des deux mots à normaliser détectés : la séquence source est dupliquée autant de fois que de mots à normaliser. Pour chacune de ces copies, un mot à normaliser à la fois est masqué, c'est-à-dire remplacé par le token de masquage du MLM (`[MASK]` pour BERT).

12. https://huggingface.co/roberta-base
13. https://huggingface.co/google/electra-base-generator
14. https://huggingface.co/vinai/bertweet-base
15. https://huggingface.co/cardiffnlp/twitter-roberta-base

Après le masquage, les séquences sont passées au correcteur. Les figures 2c et 2d illustrent les cinq meilleures prédictions des modèles BERT et BERTweet, triées par score. Elles montrent en particulier les sorties des MLM purs ($\alpha = 0{,}0$) et celles des MLM + distance d'édition ($\alpha = 0{,}3$) [16].

Nous remarquons que, sans prendre en compte le mot masqué ($\alpha = 0{,}0$), les deux modèles prédisent des mots lexicalement divers et variés. Par contre, BERTweet prédit des mots plus sémantiquement adaptés. Par exemple, lorsque **`2morrow`** est masqué, BERT ne privilégie que des signes de ponctuation (et il est très confiant pour le point) alors que BERTweet prédit *à plus tard* (**`later`**), *à bientôt* (**`soon`**) et *à demain* (**`tomorrow`**), la cible, en troisième place. En outre, il normalise déjà **`u`** en **`you`**. Cela pourrait être attribué au fait qu'il ait été entraîné sur des données Twitter. Il est donc plus robuste face à une phrase source avec des mots non standards.

En combinant les MLM avec la distance d'édition ($\alpha = 0{,}3$), leurs prédictions se rapprochent lexicalement du mot masqué. À la place de **`2morrow`**, BERT prédit toujours un point en premier, mais avec moins de confiance, et ensuite des mots proches lexicalement (dont la cible en troisième place). D'autre part, BERTweet exhibe un réordonnancement des prédictions, avec la cible en deuxième place. Il prédit en premier le mot même à normaliser : comme il fait déjà partie du lexique de BERTweet, ce dernier le considère comme standard et, par conséquent, ne le normalise pas.

4.3 Expériences

Dans un premier temps, nous avons entraîné le détecteur (cf. la section 4.2.1) sur le jeu d'entraînement de LexNorm2015 et évalué sa performance de détection de mots à normaliser sur le jeu de test. Ensuite, nous avons comparé les cinq MLM pré-entraînés (cf. la section 4.2.2) pour la correction des mots détectés dans le jeu de test. Nous avons choisi comme système de base le modèle « identité » (qui consiste à ne rien changer). Pour chaque correcteur, nous avons généré les prédictions en tenant de plus en plus compte de la distance d'édition par rapport aux mots masqués, c'est-à-dire en augmentant la valeur α (de 0 à 1 par pas de $0{,}1$).

Nous avons utilisé les métriques automatiques (exactitude, précision, rappel et F-mesure) pour évaluer le détecteur. Pour les correcteurs, nous nous aussi sommes servi des métriques automatiques, comme dans la campagne d'évaluation LexNorm2015 (Baldwin *et al.*, 2015), et de l'ERR utilisée dans la campagne d'évaluation MultiLexNorm2021 (van der Goot *et al.*, 2021). En outre, nous avons évalué les sorties normalisées par deux métriques usuelles de traduction : BLEU (Papineni *et al.*, 2002), en particulier l'implémentation SacreBLEU [17] (Post, 2018), et COMET [18] (Rei *et al.*, 2020). Afin de pouvoir comparer les différents modèles, toutes les métriques ont été effectuées à l'échelle des mots.

4.4 Résultats

Nous avons évalué la performance du détecteur sur le jeu de test de LexNorm2015 et nous avons obtenu les scores suivants : $97{,}82\%$ d'exactitude, $90{,}14\%$ de précision, $86{,}41\%$ de rappel et $88{,}24\%$ de F-mesure. Dans les sections suivantes, nous analyserons la performance des différents correcteurs.

16. Nous avons choisi la valeur $0{,}3$ car elle obtient quasiment les meilleurs scores dans les expériences (cf. la section 4.4).
17. https://huggingface.co/spaces/evaluate-metric/sacrebleu
18. https://huggingface.co/spaces/evaluate-metric/comet

4.4.1 Analyse qualitative

Le tableau 3 illustre les normalisations d'un tweet du jeu de test LexNorm2015 par les correcteurs BERT et BERTweet. Nous avons choisi les sorties pour $\alpha = 0$ (MLM seul), $\alpha = 0{,}3$ (MLM + distance d'édition) et $\alpha = 1$ (distance d'édition seule).

Source	rt @tehreelhov : **wen ur** at a restaurant **nd u c ur** food **comin** http://t.co/ducpxt7dry
Cible	rt @tehreelhov : **when you're** at a restaurant **and you see your** food **coming** http://t.co/ducpxt7dry
$\alpha = 0$	rt @tehreelhov : r ##d at a restaurant . : : a food . http://t.co/ducpxt7dry
$\alpha = 0{,}3$	rt @tehreelhov : wen ur at a restaurant **and** u c ur food **coming** http://t.co/ducpxt7dry
$\alpha = 1$	rt @tehreelhov : wen ur at a restaurant **and** u c ur food **coming** http://t.co/ducpxt7dry

(a) BERT

Source	rt @tehreelhov : **wen ur** at a restaurant **nd u c ur** food **comin** http://t.co/ducpxt7dry
Cible	rt @tehreelhov : **when you're** at a restaurant **and you see your** food **coming** http://t.co/ducpxt7dry
$\alpha = 0$	rt @tehreelhov : **when** ur at a restaurant **and** u **see** ur food <@@ http://t.co/ducpxt7dry
$\alpha = 0{,}3$	rt @tehreelhov : **when** ur at a restaurant **and** u **see** ur food comin http://t.co/ducpxt7dry
$\alpha = 1$	rt @tehreelhov : wen ur at a restaurant nd u c ur food comin http://t.co/ducpxt7dry

(b) BERTweet

TABLE 3 – Normalisation d'un tweet par les correcteurs BERT et BERTweet pour $\alpha = 0$, 0,3 et 1.
(Traduction : *Retweet [Utilisateur] : quand tu es dans un restaurant et tu vois ta nourriture arriver [Lien URL]*)

BERT (cf. le tableau 3a) Lorsque le correcteur comprend le MLM uniquement ($\alpha = 0$), les mots à normaliser sont remplacés par des tokens fréquents (des lettres ou de la ponctuation) qui ne leur sont ni lexicalement ni sémantiquement proches. D'ailleurs, nous pouvons avancer que le sens du tweet d'origine se dégrade, voire se perd complètement.

En combinant le MLM avec la distance d'édition ($\alpha = 0{,}3$), seulement deux mots sont correctement normalisés. Même si les autres mots à normaliser restent inchangés (puisqu'ils appartiennent au vocabulaire de BERT), cette sortie est une amélioration car le modèle a corrigé quelques mots sans y introduire plus de bruit.

Enfin, lorsque nous prenons la distance d'édition seule ($\alpha = 1$), le correcteur prédit les mêmes sorties obtenues pour $\alpha = 0{,}3$. Comme BERT a été entraîné sur des données standards, nous conjecturons qu'il n'a pas réussi à se représenter correctement la phrase source (assez bruitée) et n'a donc pas su prédire de tokens pertinents. Par conséquent, il semble que toute la normalisation avait été effectuée par le calcul de la distance d'édition.

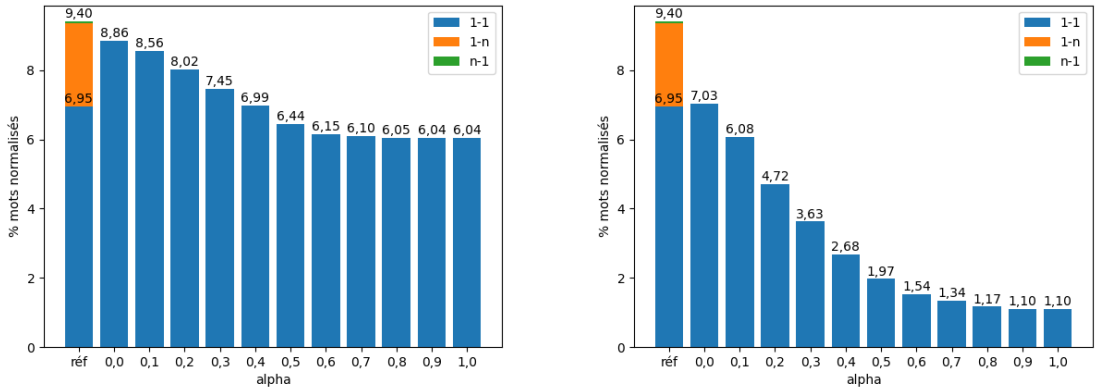
BERTweet (cf. le tableau 3b) Ayant été entraîné sur des données Twitter, BERTweet est plus apte à modéliser la phrase source et à prédire des tokens sémantiquement pertinents à la place des mots masqués. À lui tout seul ($\alpha = 0$), il normalise déjà correctement trois mots, et certains mots restent inchangés car ils appartiennent à son vocabulaire. Par ailleurs, il introduit un seul token erroné **<@@** [19].

Similairement à BERT, lorsque BERTweet est combiné avec la distance d'édition ($\alpha = 0{,}3$), il n'introduit plus de bruit : il remplace tous les tokens précédemment mal prédits par les mots masqués.

Lorsque $\alpha = 1$, la sortie du correcteur est aussi bruitée que la phrase source : tous les mots non

19. qui se colle au lien URL pour former `<http://t.co/ducpxt7dry`.

standards du tweet font partie du vocabulaire. Nous perdons aussi toutes les normalisations que ce dernier avait réussi à réaliser. Nous pouvons donc conclure que la distance d'édition ne suffit pas pour corriger la phrase. Par contre, nous voyons aussi les limites de cette distance : la normalisation attendue pour **c** est **see**, qui ne lui est pas lexicalement proche [20] alors qu'elle l'est phonétiquement.



(a) BERT

(b) BERTweet

FIGURE 3 – Statistiques des mots normalisés par les correcteurs BERT et BERTweet.

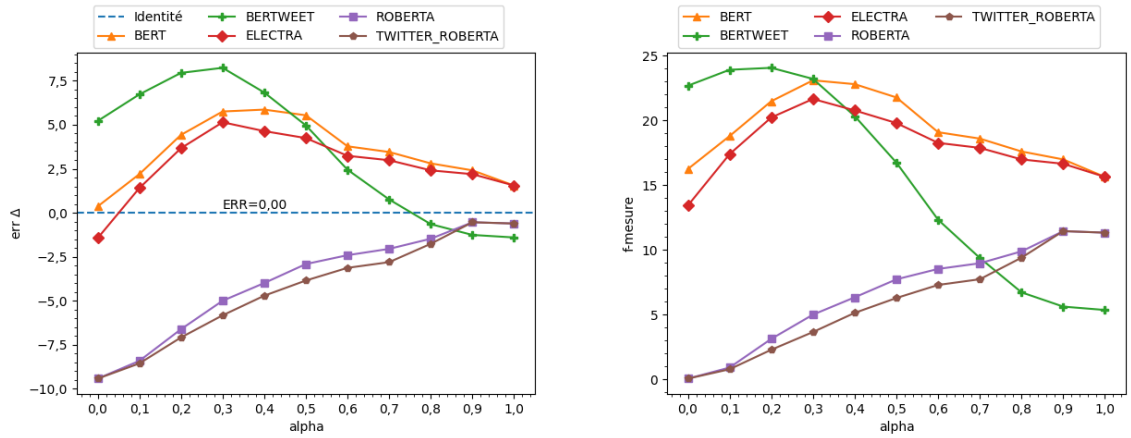
Ces phénomènes sont aussi observables sur la figure 3. D'une part, BERT tout seul « normalise » plus de mots que nécessaire : il introduit du bruit dans la phrase. Or, plus le poids de la distance d'édition augmente, plus le pourcentage de mots normalisés se rapproche de la référence. D'autre part, BERTweet tout seul modifie presque autant de mots que la référence. Cependant, le pourcentage de mots normalisés diminue exponentiellement lorsque α augmente.

Remarque. Les correcteurs utilisés sont à base de MLM entraînés pour prédire un seul token pour chaque masque. Les normalisations de type 1-N et N-1 ne sont donc pas réalisables. De plus, même les normalisations 1-1 réalisées sont parfois suboptimales car les modèles ne peuvent prédire que des mots composés d'un seul token. Ils prédisent même parfois des tokens isolés comme **##d** et **<@@** dans les tableaux 3a et 3b respectivement.

4.4.2 Analyse quantitative avec métriques automatiques

La figure 4 illustre les scores d'ERR (4a) et de F-mesure (4b) des correcteurs considérés pour différentes valeurs de α. Nous remarquons trois comportements différents. D'abord, BERTweet a déjà des scores au dessus du système de base pour $\alpha = 0$. Cela rejoint l'hypothèse qu'il est plus robuste au bruit et qu'il arrive à extraire des informations sémantiques des tweets malgré leurs mots non standards. En le combinant avec la distance d'édition, les scores augmentent pour les premières valeurs de $\alpha \leq 0{,}3$ avant de dégrader progressivement jusqu'en dessous du système de base. Ensuite, BERT et ELECTRA (qui lui reste légèrement en dessous), sont proches du système de base pour $\alpha = 0$. Ils augmentent progressivement lorsque $\alpha \leq 0{,}5$ et se dégradent tout en restant au dessus du système de base. Finalement, RoBERTa et Twitter RoBERTa restent toujours en dessous du système de base, même si leurs scores augmentent avec α. Lorsque $\alpha = 1$, nous observons la convergence des modèles qui ont été entraînés sur les mêmes données et la même taille de vocabulaire.

20. La distance de Damerau-Levenshtein entre **c** et **see** est 3 (une substitution et 2 insertions).



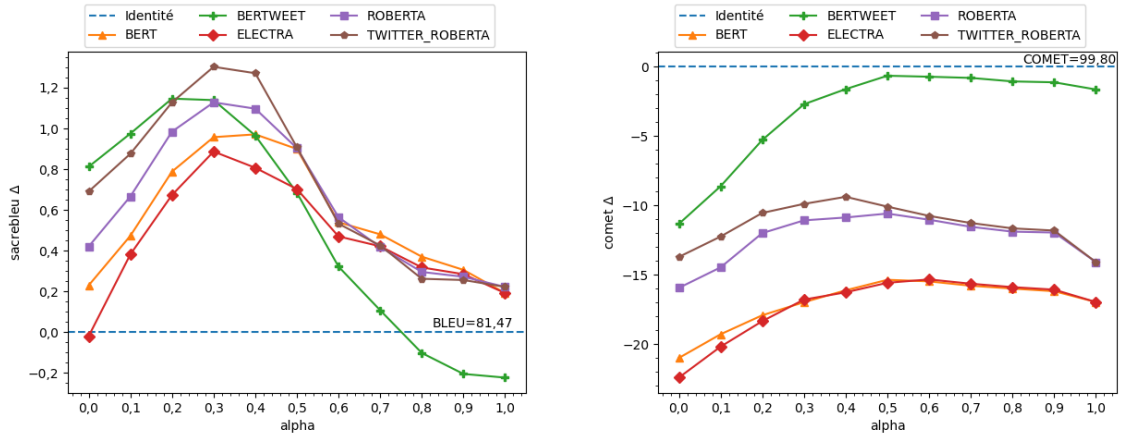
(a) ERR (b) F-mesure

FIGURE 4 – Évaluation des correcteurs par métriques automatiques.

Les courbes de scores d'exactitude suivent les mêmes tendances que celles de l'ERR. De même, celles de précision et de rappel suivent les mêmes tendances que celles de la F-mesure.

Remarque. Comme le système de base ne change rien aux phrases sources, il n'y a pas de mots normalisés (pas de prédictions positives). Nous ne pouvons donc rien conclure sur la précision, ni calculer la F-mesure (d'où l'absence du modèle identité sur la figure 4b).

4.4.3 Analyse quantitative avec métriques de traduction



(a) BLEU (b) COMET

FIGURE 5 – Évaluation des correcteurs par métriques de traduction.

La figure 5 illustre les scores BLEU (5a) et COMET (5b) des correcteurs pour différentes valeurs de α. Nous observons des comportements très différents sur ces deux métriques. D'une part, tous les modèles montrent une amélioration de score BLEU par rapport au modèle identité (sauf BERTweet lorsque α est grand). Nous observons des pics autour des valeurs de $\alpha = 0{,}3_{\pm 0{,}1}$. En plus, les modèles qui ont été exposés à des données Twitter lors de leur entraînement (BERTweet et Twitter RoBERTa)

ont les meilleurs scores, suivis de RoBERTa, puis BERT et ELECTRA. D'autre part, nous remarquons que tous les scores COMET augmentent progressivement avec α mais restent toujours en dessous du modèle de base. En particulier, BERTweet est le meilleur, suivi de RoBERTa et Twitter RoBERTa, et enfin de BERT et ELECTRA.

Les différences observées entre les deux métriques sont frappantes (pour BLEU, la plupart des scores sont supérieurs à la ligne de base, alors que pour COMET, ils sont tous inférieurs), mais elles s'expliquent simplement : alors que BLEU est une métrique de *surface* qui repose sur le comptage de n-grammes partagés entre une référence et une prédiction, COMET est une métrique neuronale dont l'objectif est de dépasser la variation de surface pour comparer le *sens* des deux textes. Selon BLEU, une prédiction erronée qui est plus éloignée de la forme normalisée de référence ne sera pas plus fortement pénalisée. D'autre part, la fonction identité obtient un score COMET élevé parce que celui-ci est assez robuste pour juger que le texte de départ est sémantiquement très similaire à la référence normalisée. À l'inverse, toute erreur introduite par un modèle de normalisation sera sanctionnée à des scores COMET plus faibles. Ainsi, pour la tâche de normalisation, aucune de ces mesures n'est totalement adéquate à elle seule ; il serait donc intéressant de les examiner ensemble.

5 Discussion et Conclusion

Cet article a pour vocation d'illustrer l'utilité mais aussi les limites de la tâche de normalisation lexicale des contenus générés par les utilisateurs (CGU), et d'ouvrir la porte à plusieurs perspectives de travaux de recherche. Dans un premier temps, nous avons présenté un état de l'art de la normalisation lexicale des CGU sur les réseaux sociaux. Nous avons montré qu'ils sont un fléau pour les modèles de TAL entraînés sur des données standards à cause de leur multitude de phénomènes de langage non standard, et que la normalisation lexicale est l'une des approches pratiques pour pallier ce problème.

Dans un second temps, nous avons fait une étude expérimentale pour montrer que, malgré tous ses avantages, la normalisation lexicale de CGU reste une tâche difficile à réaliser et à évaluer. Premièrement, nous observons que, même avec des modèles de langue pré-entraînés, il est difficile de faire mieux qu'un système de base qui ne fait rien. Certes, l'intégration d'une distance d'édition et l'entraînement sur des données CGU améliorent la performance des modèles. Ensuite, nous observons l'inadéquation des métriques classiques pour ce genre de tâche. Par exemple, le pire modèle pour les métriques automatiques (Twitter RoBERTa) est l'un des meilleurs pour la métrique de traduction BLEU, mais n'a pas de bon score pour une autre métrique de traduction, COMET.

Nos expériences étant préliminaires, plusieurs pistes d'amélioration peuvent être envisagées, notamment : filtrer les sorties de BERTweet par un lexique standard, apprendre la valeur optimale de α pendant l'entraînement, intégrer un module de similarité phonétique, utiliser des MLM qui masquent tous les tokens d'un mot (*whole-word masking*) ou plusieurs tokens adjacents (*span masking*), affiner ELECTRA pour la détection de mots à normaliser (Yuan *et al.* (2021) ont fait l'hypothèse que c'est un meilleur détecteur car il a un objectif d'entraînement discriminant), normaliser la séquence autorégressivement (Sun & Jiang, 2019), garder le mot d'origine si la normalisation prédite est pire (van der Goot, 2019b), etc.

Remerciements

Un grand merci à mes encadrants Rachel Bawden et Benoît Sagot pour leur soutien, aux relecteurs RECITAL pour leurs commentaires précieux, et à Menel Mahamdi pour sa relecture judicieuse. Ce travail a été financé par la chaire de R. Bawden à l'institut PRAIRIE financé par l'agence nationale française ANR dans le cadre du programme "Investissements d'avenir" sous la référence ANR-19-P3IA-0001 et également par le projet Emergence, DadaNMT, financé par Sorbonne Université.

Références

AHMED B. (2015). Lexical normalisation of Twitter Data. In *Proceedings of the 2015 Science and Information Conference*, p. 326–328, London, UK : IEEE. DOI : 10.1109/SAI.2015.7237164.

ALAM M. M. I. & ANASTASOPOULOS A. (2020). Fine-Tuning MT systems for Robustness to Second-Language Speaker Variations. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, p. 149–158, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.wnut-1.20.

ALEGRIA I., ARANBERRI N., FRESNO-FERNÁNDEZ V., GAMALLO P., PADRÓ L., VICENTE I. S., TURMO J. & ZUBIAGA A. (2013). Introducción a la Tarea Compartida Tweet-Norm 2013 : Normalización Léxica de Tuits en Español. In *Proceedings of the XXIX Congreso de la Sociedad Española de Procesamiento de Lenguaje Natural*, p. 38–46, Madrid, Spain.

AMINIAN M., AVONTUUR T., BALEMANS I., ELSHOF L., NEWELL R., NOORD N. V., NTAVELOS A., VAN ZAANEN M. & AZAR E. Z. (2012). Assigning part-of-speech to Dutch tweets.

AW A., ZHANG M., XIAO J. & SU J. (2006). A phrase-based statistical model for SMS text normalization. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, p. 33–40, Sydney, Australia : Association for Computational Linguistics.

BALDWIN T., COOK P., LUI M., MACKINLAY A. & WANG L. (2013). How noisy social media text, how diffrnt social media sources ? In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, p. 356–364, Nagoya, Japan : Asian Federation of Natural Language Processing.

BALDWIN T., DE MARNEFFE M. C., HAN B., KIM Y.-B., RITTER A. & XU W. (2015). Shared tasks of the 2015 workshop on noisy user-generated text : Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, p. 126–135, Beijing, China : Association for Computational Linguistics. DOI : 10.18653/v1/W15-4319.

BARANES M. & SAGOT B. (2014). Analogy-based text normalization : the case of unknowns words (normalisation de textes par analogie : le cas des mots inconnus) [in French]. In *Proceedings of TALN 2014 (Volume 1 : Long Papers)*, p. 137–148, Marseille, France : Association pour le Traitement Automatique des Langues.

BARBIERI F., CAMACHO-COLLADOS J., ESPINOSA ANKE L. & NEVES L. (2020). TweetEval : Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics : EMNLP 2020*, p. 1644–1650, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.findings-emnlp.148.

BARIK A. M., MAHENDRA R. & ADRIANI M. (2019). Normalization of Indonesian-English Code-Mixed Twitter Data. In *Proceedings of the 5th Workshop on Noisy User-generated Text*

(W-NUT 2019), p. 417–424, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5554.

BAWDEN R., POINHOS J., KOGKITSIDOU E., GAMBETTE P., SAGOT B. & GABAY S. (2022). Automatic normalisation of early Modern French. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, p. 3354–3366, Marseille, France : European Language Resources Association.

BECKLEY R. (2015). Bekli :A Simple Approach to Twitter Text Normalization. In *Proceedings of the Workshop on Noisy User-generated Text*, p. 82–86, Beijing, China : Association for Computational Linguistics. DOI : 10.18653/v1/W15-4312.

BELINKOV Y. & BISK Y. (2017). Synthetic and Natural Noise Both Break Neural Machine Translation. *ICLR*.

BIKAUN T., FRENCH T., HODKIEWICZ M., STEWART M. & LIU W. (2021). LexiClean : An annotation tool for rapid multi-task lexical normalisation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing : System Demonstrations*, p. 212–219, Online and Punta Cana, Dominican Republic : Association for Computational Linguistics. DOI : 10.18653/v1/2021.emnlp-demo.25.

BUCUR A.-M., COSMA A. & DINU L. P. (2021). Sequence-to-sequence lexical normalization with multilingual transformers. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 473–482, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.wnut-1.53.

CERÓN-GUZMÁN J. A. & LEÓN-GUZMÁN E. (2016). Lexical normalization of spanish tweets. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, p. 605–610, Montréal, Québec, Canada : International World Wide Web Conferences Steering Committee. DOI : 10.1145/2872518.2890558.

CHOUDHURY M., SARAF R., JAIN V., MUKHERJEE A., SARKAR S. & BASU A. (2007). Investigation and modeling of the structure of texting language. *International Journal of Document Analysis and Recognition (IJDAR)*, **10**(3-4), 157–174. DOI : 10.1007/s10032-007-0054-0.

CLARK E. & ARAKI K. (2011). Text Normalization in Social Media : Progress, Problems and Applications for a Pre-Processing System of Casual English. *Procedia - Social and Behavioral Sciences*, **27**, 2–11. DOI : 10.1016/j.sbspro.2011.10.577.

CLARK K., LUONG M.-T., LE Q. V. & MANNING C. D. (2020). ELECTRA : Pre-training Text Encoders as Discriminators Rather Than Generators. In *Proceedings of the Eighth International Conference on Learning Representations*, Online.

COTELO J., CRUZ F., TROYANO J. & ORTEGA F. (2015). A modular approach for lexical normalization applied to Spanish tweets. *Expert Systems with Applications*, **42**(10), 4743–4754. DOI : 10.1016/j.eswa.2015.02.003.

DAMERAU F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM*, **7**(3), 171–176. DOI : 10.1145/363958.363994.

DE CLERCQ O., SCHULZ S., DESMET B. & HOSTE V. (2014). Towards Shared Datasets for Normalization Research. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, p. 1218–1223, Reykjavik, Iceland : European Language Resources Association (ELRA).

DEVLIN J., CHANG M.-W., LEE K. & TOUTANOVA K. (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the*

North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers), p. 4171–4186, Minneapolis, Minnesota : Association for Computational Linguistics. DOI : 10.18653/v1/N19-1423.

EHARA Y. (2021). To What Extent Does Lexical Normalization Help English-as-a-Second Language Learners to Read Noisy English Texts ? In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 451–456, Online : Association for Computational Linguistics.

EISENSTEIN J. (2013). What to do about bad language on the internet. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 359–369, Atlanta, Georgia : Association for Computational Linguistics.

ERJAVEC T., FIŠER D., ČIBEJ J., ARHAR HOLDT Š., LJUBEŠIĆ N. & ZUPAN K. (2017). CMC training corpus janes-tag 2.0. Slovenian language resource repository CLARIN.SI.

FORMIGA L. & FONOLLOSA J. A. R. (2012). Dealing with input noise in statistical machine translation. In *Proceedings of COLING 2012 : Posters*, p. 319–328, Mumbai, India : The COLING 2012 Organizing Committee.

FOSTER J. (2010). "cba to check the spelling" : Investigating Parser Performance on Discussion Forum Posts. In *Human Language Technologies : The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, p. 381–384, Los Angeles, California : Association for Computational Linguistics.

HAN B. & BALDWIN T. (2011). Lexical Normalisation of Short Text Messages : Makn Sens a #twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics : Human Language Technologies*, p. 368–378, Portland, Oregon, USA : Association for Computational Linguistics.

HIGASHIYAMA S., UTIYAMA M., WATANABE T. & SUMITA E. (2021). User-generated text corpus for evaluating Japanese morphological analysis and lexical normalization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 5532–5541, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.naacl-main.438.

JIANG N., LUO C., LAKSHMAN V., DATTATREYA Y. & XUE Y. (2022). Massive Text Normalization via an Efficient Randomized Algorithm. In *Proceedings of the ACM Web Conference 2022*, p. 2946–2956, Virtual Event, Lyon France : ACM. DOI : 10.1145/3485447.3512015.

KARPUKHIN V., LEVY O., EISENSTEIN J. & GHAZVININEJAD M. (2019). Training on Synthetic Noise Improves Robustness to Natural Noise in Machine Translation. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 42–47, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5506.

KOBUS C., YVON F. & DAMNATI G. (2008). Normalizing SMS : are Two Metaphors Better than One ? In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, p. 441–448, Manchester, UK : Coling 2008 Organizing Committee.

KOGKITSIDOU E. & ANTONIADIS G. (2016). L'architecture d'un modèle hybride pour la normalisation de SMS (a hybrid model architecture for SMS normalization). In *Actes de la conférence conjointe JEP-TALN-RECITAL 2016. volume 2 : TALN (Posters)*, p. 355–363, Paris, France : AFCP - ATALA.

KUBAL D. & NAGVENKAR A. (2021). Multilingual Sequence Labeling Approach to solve Lexical Normalization. In *Proceedings of the Seventh Workshop on Noisy User-generated*

Text (W-NUT 2021), p. 457–464, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.wnut-1.51.

KUMAR A., MAKHIJA P. & GUPTA A. (2020). Noisy Text Data : Achilles' Heel of BERT. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, p. 16–21, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.wnut-1.3.

LEEMAN-MUNK S., LESTER J. & COX J. (2015). NCSU_sas_sam : Deep Encoding and Reconstruction for Normalization of Noisy Text. In *Proceedings of the Workshop on Noisy User-generated Text*, p. 154–161, Beijing, China : Association for Computational Linguistics. DOI : 10.18653/v1/W15-4323.

LEVENSHTEIN V. I. (1966). Binary Codes Capable of Correcting Deletions, Insertions and Reversals. In *Soviet Physics Doklady*, volume 10 de *8*, p. 707–710.

LI C. & LIU Y. (2012). Improving text normalization using character-blocks based models and system combination. In *Proceedings of COLING 2012*, p. 1587–1602, Mumbai, India : The COLING 2012 Organizing Committee.

LIU F., WENG F. & JIANG X. (2012). A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, p. 1035–1044, Jeju Island, Korea : Association for Computational Linguistics.

LIU Y., GU J., GOYAL N., LI X., EDUNOV S., GHAZVININEJAD M., LEWIS M. & ZETTLEMOYER L. (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, **8**, 726–742. DOI : 10.1162/tacl_a_00343.

LIU Y., OTT M., GOYAL N., DU J., JOSHI M., CHEN D., LEVY O., LEWIS M., ZETTLEMOYER L. & STOYANOV V. (2019). Roberta : A robustly optimized BERT pretraining approach. *CoRR*, **abs/1907.11692**.

LIU Z., NI S., AW A. T. & CHEN N. F. (2022). Singlish message paraphrasing : A joint task of creole translation and text normalization. In *Proceedings of the 29th International Conference on Computational Linguistics*, p. 3924–3936, Gyeongju, Republic of Korea : International Committee on Computational Linguistics.

LJUBEŠIĆ N., ERJAVEC T., MILIČEVIĆ M. & SAMARDŽIĆ T. (2017a). Croatian twitter training corpus ReLDI-NormTagNER-hr 2.0. Slovenian language resource repository CLARIN.SI.

LJUBEŠIĆ N., ERJAVEC T., MILIČEVIĆ M. & SAMARDŽIĆ T. (2017b). Serbian twitter training corpus ReLDI-NormTagNER-sr 2.0. Slovenian language resource repository CLARIN.SI.

LOURENTZOU I., MANGHNANI K. & ZHAI C. (2019). Adapting Sequence to Sequence models for Text Normalization in Social Media. In *Proceedings of the Thirteenth International AAAI Conference on Web and Social Media (ICWSM 2019)*, p. 335–345, Münich, Germany.

MATOS VELIZ C., DE CLERCQ O. & HOSTE V. (2019). Comparing MT Approaches for Text Normalization. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, p. 740–749, Varna, Bulgaria : INCOMA Ltd. DOI : 10.26615/978-954-452-056-4_086.

MELERO M., COSTA-JUSSÀ M. R., LAMBERT P. & QUIXAL M. (2016). Selection of correction candidates for the normalization of Spanish user-generated content. *Natural Language Engineering*, **22**(1), 135–161. Publisher : Cambridge University Press, DOI : 10.1017/S1351324914000011.

MICHEL P. & NEUBIG G. (2018). MTNT : A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, p. 543–553, Brussels, Belgium : Association for Computational Linguistics. DOI : 10.18653/v1/D18-1050.

MOON S., NEVES L. & CARVALHO V. (2018). Multimodal Named Entity Recognition for Short Social Media Posts. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long Papers)*, p. 852–860, New Orleans, Louisiana : Association for Computational Linguistics. DOI : 10.18653/v1/N18-1078.

MUÑOZ-GARCÍA Ó., NAVARRO C., AVONTUUR T., AZAR Z., BALEMANS I., ELSHOF L., NEWELL R., NOORD N. V., NTAVELOS A., MAYNARD D., BONTCHEVA K., ROUT D., STRASSEL S., ISMAEL S., SONG Z. & LEE H. (2012). Comparing user generated content published in different social media sources.

MULLER B., SAGOT B. & SEDDAH D. (2019). Enhancing BERT for Lexical Normalization. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 297–306, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5539.

NGUYEN D. Q., VU T. & TUAN NGUYEN A. (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*, p. 9–14, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.emnlp-demos.2.

NGUYEN V. H., NGUYEN H. T. & SNASEL V. (2016). Text normalization for named entity recognition in Vietnamese tweets. *Computational Social Networks*, **3**(1), 10. DOI : 10.1186/s40649-016-0032-0.

PAPINENI K., ROUKOS S., WARD T. & ZHU W.-J. (2002). BLEU : a Method for Automatic Evaluation of Machine Translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, p. 311–318, Philadelphia, Pennsylvania, USA : Association for Computational Linguistics. DOI : 10.3115/1073083.1073135.

PARTANEN N., HÄMÄLÄINEN M. & ALNAJJAR K. (2019). Dialect Text Normalization to Normative Standard Finnish. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 141–146, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5519.

PENNELL D. & LIU Y. (2011). A character-level machine translation approach for normalization of SMS abbreviations. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, p. 974–982, Chiang Mai, Thailand : Asian Federation of Natural Language Processing.

PLANK B., JENSEN K. N. & VAN DER GOOT R. (2020). DaN+ : Danish nested named entities and lexical normalization. In *Proceedings of the 28th International Conference on Computational Linguistics*, p. 6649–6662, Barcelona, Spain (Online) : International Committee on Computational Linguistics. DOI : 10.18653/v1/2020.coling-main.583.

POST M. (2018). A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation : Research Papers*, p. 186–191, Brussels, Belgium : Association for Computational Linguistics. DOI : 10.18653/v1/W18-6319.

QIN W., LI X., SUN Y., XIONG D., CUI J. & WANG B. (2021). Modeling Homophone Noise for Robust Neural Machine Translation. In *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, p. 7533–7537, Toronto, ON, Canada. DOI : 10.1109/ICASSP39728.2021.9413586.

REI R., STEWART C., FARINHA A. C. & LAVIE A. (2020). COMET : A Neural Framework for MT Evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 2685–2702, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.emnlp-main.213.

RIABI A., SAGOT B. & SEDDAH D. (2021). Can Character-based Language Models Improve Downstream Task Performances In Low-Resource And Noisy Language Scenarios ? In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 423–436, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.wnut-1.47.

ROSALES NÚÑEZ J. C., SEDDAH D. & WISNIEWSKI G. (2019a). Comparison between NMT and PBSMT Performance for Translating Noisy User-Generated Content. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, p. 2–14, Turku, Finland : Linköping University Electronic Press.

ROSALES NÚÑEZ J. C., SEDDAH D. & WISNIEWSKI G. (2019b). Phonetic Normalization for Machine Translation of User Generated Content. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 407–416, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5553.

ROSALES NÚÑEZ J. C., SEDDAH D. & WISNIEWSKI G. (2021a). Understanding the Impact of UGC Specificities on Translation Quality. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 189–198, Online : Association for Computational Linguistics.

ROSALES NÚÑEZ J. C., WISNIEWSKI G. & SEDDAH D. (2021b). Noisy UGC Translation at the Character Level : Revisiting Open-Vocabulary Capabilities and Robustness of Char-Based Models. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 199–211, Online : Association for Computational Linguistics.

RUIZ P., CUADROS M. & ETCHEGOYHEN T. (2014). Lexical Normalization of Spanish Tweets with Rule-Based Components and Language Models. *Procesamiento del Lenguaje Natural*, **52**, 45–52.

SANGUINETTI M., BOSCO C., CASSIDY L., ÇETINOĞLU Ö., CIGNARELLA A. T., LYNN T., REHBEIN I., RUPPENHOFER J., SEDDAH D. & ZELDES A. (2020). Treebanking User-Generated Content : A Proposal for a Unified Representation in Universal Dependencies. In *Proceedings of the 12th Language Resources and Evaluation Conference*, p. 5240–5250, Marseille, France : European Language Resources Association.

SARKAR R., MAHINDER S. & KHUDABUKHSH A. (2020). The Non-native Speaker Aspect : Indian English in Social Media. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, p. 61–70, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2020.wnut-1.9.

SCHERRER Y. & LJUBEŠIĆ N. (2021). Sesame Street to Mount Sinai : BERT-constrained character-level Moses models for multilingual lexical normalization. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 465–472, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.wnut-1.52.

SCHUUR Y. (2020). Normalization for Dutch for improved POS tagging. Mémoire de master, University of Groningen.

SEDDAH D., SAGOT B., CANDITO M., MOUILLERON V. & COMBET V. (2012). The French Social Media Bank : a Treebank of Noisy User Generated Content. In *Proceedings of COLING 2012*, p. 2441–2458, Mumbai, India : The COLING 2012 Organizing Committee.

SIDARENKA U., SCHEFFLER T. & STEDE M. (2013). Rule-Based Normalization of German Twitter Messages. In *Proceedings of the International Conference of the German Society for Computational Linguistics and Language Technology*, Darmstadt, Germany.

SPROAT R., BLACK A. W., CHEN S., KUMAR S., OSTENDORF M. & RICHARDS C. D. (2001). Normalization of non-standard words. *Computer Speech & Language*, **15**(3), 287–333. DOI : 10.1006/csla.2001.0169.

STEWART M., LIU W. & CARDELL-OLIVER R. (2019). Word-level lexical normalisation using context-dependent embeddings. *CoRR*, **abs/1911.06172**.

SUN Y. & JIANG H. (2019). Contextual Text Denoising with Masked Language Model. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 286–290, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5537.

SUPRANOVICH D. & PATSEPNIA V. (2015). IHS_rd : Lexical Normalization for English Tweets. In *Proceedings of the Workshop on Noisy User-generated Text*, p. 78–81, Beijing, China : Association for Computational Linguistics. DOI : 10.18653/v1/W15-4311.

TIAN T., TELLIER I., DINARELLI M. & CARDOSO P. (2017). Détection des mots non-standards dans les tweets avec des réseaux de neurones (detecting non-standard words in tweets with neural networks). In *Actes des 24ème Conférence sur le Traitement Automatique des Langues Naturelles. Volume 2 - Articles courts*, p. 174–182, Orléans, France : ATALA.

TIWARI A. S. & NASKAR S. K. (2017). Normalization of social media text using deep neural networks. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, p. 312–321, Kolkata, India : NLP Association of India.

VAN DER GOOT R. (2019a). An In-depth Analysis of the Effect of Lexical Normalization on the Dependency Parsing of Social Media. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, p. 115–120, Hong Kong, China : Association for Computational Linguistics. DOI : 10.18653/v1/D19-5515.

VAN DER GOOT R. (2019b). MoNoise : A Multi-lingual and Easy-to-use Lexical Normalization Tool. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics : System Demonstrations*, p. 201–206, Florence, Italy : Association for Computational Linguistics. DOI : 10.18653/v1/P19-3032.

VAN DER GOOT R. (2019c). *Normalization and parsing algorithms for uncertain input*. Thèse de doctorat, University of Groningen.

VAN DER GOOT R. (2021). CL-MoNoise : Cross-lingual lexical normalization. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 510–514, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.wnut-1.56.

VAN DER GOOT R., PLANK B. & NISSIM M. (2017). To normalize, or not to normalize : The impact of normalization on Part-of-Speech tagging. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, p. 31–39, Copenhagen, Denmark : Association for Computational Linguistics. DOI : 10.18653/v1/W17-4404.

VAN DER GOOT R., RAMPONI A., CASELLI T., CAFAGNA M. & DE MATTEI L. (2020). Norm it ! lexical normalization for Italian and its downstream effects for dependency parsing. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, p. 6272–6278, Marseille, France : European Language Resources Association.

VAN DER GOOT R., RAMPONI A., ZUBIAGA A., PLANK B., MULLER B., SAN VICENTE RONCAL I., LJUBEŠIĆ N., ÇETINOĞLU Ö., MAHENDRA R., ÇOLAKOĞLU T., BALDWIN T., CASELLI T. & SIDORENKO W. (2021). MultiLexNorm : A Shared Task on Multilingual Lexical Normalization. In *Proceedings of the Seventh Workshop on Noisy User-generated Text (W-NUT 2021)*, p. 493–509, Online : Association for Computational Linguistics.

VAN DER GOOT R. & VAN NOORD G. (2017). MoNoise : Modeling Noise Using a Modular Normalization System. *Computational Linguistics in the Netherlands Journal*, **7**, 129–144.

VAN DER GOOT R. & VAN NOORD G. (2018). Modeling Input Uncertainty in Neural Network Dependency Parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, p. 4984–4991, Brussels, Belgium : Association for Computational Linguistics. DOI : 10.18653/v1/D18-1542.

VAN DER GOOT R., VAN NOORD R. & VAN NOORD G. (2018). A taxonomy for in-depth evaluation of normalization for user generated content. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, p. 684–688, Miyazaki, Japan : European Language Resources Association (ELRA).

VAN DER GOOT R. & ÇETINOĞLU Ö. (2021). Lexical Normalization for Code-switched Data and its Effect on POS Tagging. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics : Main Volume*, p. 2352–2365, Online : Association for Computational Linguistics. DOI : 10.18653/v1/2021.eacl-main.200.

VITERBI A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*, **13**(2), 260–269. DOI : 10.1109/TIT.1967.1054010.

XU K., XIA Y. & LEE C.-H. (2015). Tweet Normalization with Syllables. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1 : Long Papers)*, p. 920–928, Beijing, China : Association for Computational Linguistics. DOI : 10.3115/v1/P15-1089.

YANG F., BAGHERI GARAKANI A., TENG Y., GAO Y., LIU J., DENG J. & SUN Y. (2022). Spelling Correction using Phonetics in E-commerce Search. In *Proceedings of The Fifth Workshop on e-Commerce and NLP (ECNLP 5)*, p. 63–67, Dublin, Ireland : Association for Computational Linguistics. DOI : 10.18653/v1/2022.ecnlp-1.9.

YUAN Z., TASLIMIPOOR S., DAVIS C. & BRYANT C. (2021). Multi-Class Grammatical Error Detection for Correction : A Tale of Two Systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, p. 8722–8736, Online and Punta Cana, Dominican Republic : Association for Computational Linguistics. DOI : 10.18653/v1/2021.emnlp-main.687.

ÇOLAKOĞLU T., SULUBACAK U. & TANTUĞ A. C. (2019). Normalizing Non-canonical Turkish Texts Using Machine Translation Approaches. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics : Student Research Workshop*, p. 267–272, Florence, Italy : Association for Computational Linguistics. DOI : 10.18653/v1/P19-2037.