

TWEETANEUSE : Fouille de motifs en caractères et plongement lexical à l'assaut du DEFT 2017

Davide Buscaldi¹, Aude Grezka², Gaël Lejeune¹

(1) LIPN, CNRS UMR 7030, 99 avenue Jean-baptiste Clément , 93430 Villetaneuse

`prenom.nom@lipn.univ-paris13.fr`

(2) LDI, CNRS UMR 7187, 99 avenue Jean-baptiste Clément , 93430 Villetaneuse

`prenom.nom@ldi.cnrs.fr`

RÉSUMÉ

Dans cet article, nous présentons les trois méthodes développées pour la participation de l'équipe TWEETANEUSE au Défi Fouille de Textes (DEFT) 2017 qui portait sur la classification de *tweets*. Nous avons développé une méthode fondée sur une extraction de motifs en caractères fermés et fréquents (ou chaînes répétées maximales au sens de l'algorithmique du texte) combinée avec des algorithmes d'apprentissage automatique. Cette méthode est assez proche des méthodes de stylométrie utilisées pour les tâches d'attribution d'auteur. Les deux autres méthodes se fondent sur 13 traits calculés à partir de ressources lexicales (FEEL, LabMT ainsi qu'une ressource interne). Pour une des méthodes, nous avons complété avec une représentation en sac de mots, pour l'autre avec une représentation en plongements lexicaux (*word embeddings*). La méthode au grain caractère s'est avérée la plus prometteuse notamment sur la tâche de détection des *tweets* figuratifs (tâche 2).

ABSTRACT

TWEETANEUSE : character-level patterns and word embeddings for the DEFT 2017

This article describes the methods developed by the TWEETANEUSE team for the 2017 edition of the French text mining challenge (DEFT 2017). This year the challenge was dedicated to *tweet* classification : polarity detection and figurative language detection. The first method we designed relies on character-level patterns used as features for training a One VS Rest classifier. These patterns can be described as "frequent closed patterns without gap" in the sense of the data mining community, according to the text algorithmics community they are called maximal repeated strings. The two other methods use 13 features computed with lexical resources (FEEL, LabMT and a resource of our own). For one of these methods we added a bag of word representation of the tweets while for the other one a word embeddings representation has been added. The character-level method produced the best results in particular for the second task : figurative tweets detection.

MOTS-CLÉS : Fouille, motifs en caractères, grain caractère, chaînes répétées, méthodes alingues, plongement lexical, tweet.

KEYWORDS: Data mining, character-level patterns, repeated strings, language agnostic, word embeddings, tweet.

1 Introduction

Certaines propriétés des langues naturelles constituent des obstacles au traitement automatique (inférence, figement, polysémie, etc.). Ces obstacles montrent que le langage n'est pas réductible à une machine et qu'une machine n'est pas (encore ?) susceptible de résoudre complètement les problèmes de modélisation du comportement humain, notamment du comportement langagier. Traiter du langage chez l'humain est un vaste chantier, car on se trouve au carrefour de plusieurs disciplines, sciences sociales, neurosciences, anthropologie, communication, etc. La complexité des langues naturelles, associée à la complexité du langage humain amène inévitablement à une difficulté de modélisation. Le langage figuratif pose ainsi de nombreux problèmes. En effet, ce type de langage attribue au sens propre des énoncés des sens supplémentaires. Le lecteur ou l'interlocuteur doit donc découvrir voire déchiffrer la véritable signification. Le langage figuratif est un déplacement du sens : il superpose au sens propre des énoncés une ou plusieurs couches de sens supplémentaires, phénomène qu'on peut illustrer entre autres par l'ironie, le sarcasme, l'humour, la métaphore ou encore les jeux de mots. Pour ce DEFT 2017, nous nous intéressons en particulier à l'ironie, au sarcasme et à l'humour.

L'analyse de *tweets*, notamment pour des tâches de classification, est un domaine en forte expansion que ce soit en Analyse de Données Textuelles, en Fouille de Textes ou en TAL. Pour la deuxième fois après l'édition 2015, les organisateurs du Défi Fouille de Textes (DEFT) ont proposé de s'intéresser à ce type de données textuelles. Les trois tâches proposées étaient les suivantes :

- Classification des tweets selon leur polarité (tâches 1 et 3)
 - tweets non figuratifs uniquement (tâche 1)
 - tweets figuratifs et non figuratifs (tâche 3)
- Identification du langage figuratif (tâche 2)

Une description plus complète des tâches et une analyse des résultats globaux est disponible dans (Benamara *et al.*, 2017). Les modalités de l'annotation et de la constitution du corpus ont été décrites par (Karoui *et al.*, 2017). Nous proposerons dans la section 2 une analyse approfondie des tâches et des données puis nous détaillerons dans la section 3 les approches que nous avons utilisées. La section 4 sera consacrée aux résultats obtenus par nos méthodes et nous proposerons quelques conclusions et perspectives dans la section 5.

2 Analyse des tâches et du corpus

Comme le signale Bres (Bres, 2011) : « L'ironie fait partie, à l'instar de la métaphore, de ces plus vieux objets linguistiques du monde qui stimulent la réflexion sans jamais l'épuiser : depuis Platon, Aristote, Quintilien, l'ironie est un objet de recherche qui traverse les âges... sans prendre une ride ».

Si nous nous appuyons sur la classification de Schoentjes (Schoentjes, 2001), l'ironie peut se diviser en quatre axes : ironie socratique, ironie de situation, ironie verbale et ironie romantique. Les travaux sur la seule ironie verbale sont nombreux. Un parcours de la littérature permet de distinguer trois grandes approches, selon lesquelles l'ironie apparaît comme un phénomène : (i) rhétorique (Grice, 1979) (Kerbrat-Orecchioni, 1978, 1980, 1986); (ii) argumentatif (Amossy, 2003; Eggs, 2009); (iii) énonciatif (Sperber & Wilson, 1978, 1998; Berrendonner, 1981, 2002; Ducrot, 1984; Clark & Gerrig, 1984; Perrin, 1996; Recanati, 2000, 2004; Wilson, 2006). L'ironie verbale, qui nous intéresse ici, exprime une contradiction entre la pensée du locuteur et son expression : c'est-à-dire un énoncé dans

lequel ce qui est dit diffère de ce qui est signifié. L'ironie peut être produite de différentes manières, dont certaines correspondent à des figures de style classiques (antiphrase, hyperbole, litote, etc.).

L'ironie est souvent amalgamée au sarcasme. Le sarcasme est en fait une forme d'ironie verbale, mais qui est intentionnellement insultant. À l'inverse de l'ironie, le sarcasme est plus difficile à déceler. En effet, être sarcastique c'est dire le contraire de ce que l'on pense, sans montrer qu'on pense le contraire de ce que l'on dit. Bien que le sarcasme soit considéré comme une « ironie mordante », il comporte plusieurs prédicats de définition qui ne sont pas présents dans l'ironie, tels que l'insulte, la méchanceté, la cruauté ou l'agression.

Enfin, l'humour constitue avec d'autres pratiques linguistiques l'une des formes les plus complexes que peuvent prendre les interactions verbales. Il est donc source de nombreuses difficultés théoriques. Un simple parcours des dictionnaires et de leurs définitions montre qu'il est difficile de nous en remettre à leurs dénominations : amusant, comique, drôle, ridicule ; dérision, fantaisie, ironie, moquerie, plaisanterie, raillerie, etc.

Au niveau linguistique, rhétorique, lexicographique et computationnel, les frontières étant floues, nous ne ferons pas de distinctions entre ironie, sarcasme et humour dans notre analyse des tweets. Les modèles théoriques que nous venons de décrire brièvement sont malgré tout à la base d'approches computationnelles de détection de l'ironie, de l'humour ou du sarcasme. Parmi les critères stylistiques et syntactico-sémantiques qu'il est nécessaire de prendre en compte pour une reconnaissance automatique efficace, nous retiendrons :

- Les émoticônes
- La ponctuation
- Les citations
- Les mots en lettres majuscules
- Les hashtags #ironie, #sarcasme, #humour
- Les interjections
- Les verbes de discours rapporté
- Les expressions comparatives
- Les séquences figées
- Le nombre d'adjectifs
- Les mots/expressions :
 - d'opposition
 - d'opinion positive (mélioratif, appréciatif, laudatif)
 - d'opinion négative (dépréciatif, péjoratif)
 - de surprise/d'étonnement
 - de subjectivité
 - d'intensité

On constate par rapport aux tweets du corpus que le figuratif est plutôt produit par les adjectifs et les adverbes (ou locutions adverbiales) que par les substantifs et les verbes. Ces derniers sont a priori moins employés de façon ironique. Ainsi dans des tweets figuratifs, les termes mélioratifs ne sont jamais employés dans leurs sens usuels, mais simplement pour signifier leurs contraires : par exemple, beau, bel, belle, joli(e) et beauté, pour laid(e), moche, mauvais(e) et laideur ; bon(ne) et bonté pour méchant(e) ou vilain(e) et méchanceté ; chanceux(euse) et chance pour malchanceux(euse) et malchance ; charmant(e), agréable, adorable pour désagréable :

être **beau** dans les livres d'Histoire. (*figurative négative*)

271 Vous voulez du **bon** goût ? RDV au #FN!!! #Honteux #JeSuisSansVoix (*figurative négative*)

3 @sormiou12 J'ai hâte de voir qui est aveuglés par la propagande... Kadhafi est tellement **bon** pour la libye... #Libye (*figurative négative*)

Pour faire de l'ironie, il est possible d'employer tous les adjectifs mélioratifs – admirable, agréable, amusant, beau, bon, capable, captivant, chanceux, compétent, fort, généreux, habile, heureux, intelligent, qualifié, etc. – dans leurs sens contraires, pour désigner des adjectifs péjoratifs. Les discours ironiques, pour disqualifier quelqu'un ou quelque chose, exploitent massivement les termes mélioratifs.

Ainsi, les qualificatifs *beau, fort, grand*, avec leurs variables en genre et en nombre, sont très productifs, car ils participent de beaucoup d'expressions ironiques et d'antiphrases :

627 On commence par de la #Politique et #Sarkozy qui dit de très **belles** choses sur le **beau** monde qui l'entoure <http://huff.to/1tFIzF6> (*figurative négative*)

77 @najatvb "L'ancien exécutif a obtenu le silence de PSA" Trop **fort** le Gouvernement #Fillon #déresponsabilisation (*figurative négative*)

On peut former avec ces qualificatifs une infinité de mots ou d'expressions ironiques, sarcastiques ou humoristiques :

272 T'as vu JohnEdwardJR , Laurent Blanc a mis Zlatan sur le terrain c'est vraiment un **grand** entraîneur non ? Cc kalla_munasawa (*figurative négative*)

264 Ahhh vivement les nrj music awards ce soir !! Un **grand** moment de musique chaque année, des artistes au top... Le rêve !! (*figurative négative*)

421 " Poutine est un **grand** démocrate ! " Poison d'avril #poisondavril <http://www.tonvoisin.net> (*figurative négative*)

En outre, pour faire de l'ironie, les adverbes d'intensité comme *très, trop* ou *plus* sont aussi très productifs. Ils sont souvent employés pour produire de nouveaux énoncés figuratifs. En ce sens, l'on comprend que l'ironie peut être classée comme figure de mots. Car ce seront les mots employés pour leurs antonymes, qui produiront l'ironie :

652 Faite une loi macarons plutôt Ca serait **trop** cool ! #politique #Macron (*figurative négative*)

56 Nadine Morano a eu la chance de pouvoir rencontrer Sarkozy ! Elle a eu une photo et un autographe. **Trop** de chance j'suis deg' :((*figurative négative*)

Cependant, pour que le figuratif soit perçu à ce niveau, il faudra que le *tweet* ne corresponde pas à l'observation du réel, c'est-à-dire révèle une contradiction par rapport au réel. Car, il va sans dire, que les adjectifs mélioratifs peuvent être utilisés en leurs sens propres, référentiels et dénotatifs :

1063 Putain DSK il est trop fort noooooon jisouifan (*non figurative*)

Pour savoir s'ils sont employés dans leurs sens réels ou pas, c'est la dimension pragmatique qui va jouer un rôle important.

Enfin, les guillemets vont également être un indice dans le repérage de tweets figuratifs. Le principal usage des guillemets est de mettre en relief une expression, un terme ou une citation :

494 La commémoration de l'anniversaire du décès de Bourguiba est "un devoir sacré" déclare Marzouki <http://goo.gl/arWPb4> (*objective*)

Les guillemets s’emploient ainsi pour le discours rapporté et les dialogues. Selon le Dictionnaire de l’Académie Française, le mot mis entre guillemets « s’emploie pour exprimer une réserve sur un terme que l’on ne prend pas à son compte ». Mais, comme on peut le voir à travers les tweets, les guillemets sont parfois utilisés pour indiquer que le terme ou l’expression mis en exergue n’a pas sa signification littérale ou habituelle. On parlera alors de guillemets d’ironie. Autrement dit, la mise entre guillemets peut représenter, selon le cas, un type d’ironie.

398 Le "des bries" de meaux à la télé , "copé" le moi en 12, C pour partager ! :) :) (*figurative négative*)

378 Jamais élue #Hidalgo (36.62%) face à l’ump #Goujon (63.37%) est maire de #Paris.. C’est si l’on peut dire, une "création in vitro" (*figurative négative*)

3 Approches

Dans cette section, nous présenterons le fonctionnement des deux approches que nous avons utilisées pour cette édition du DEFT. La première méthode (*run1*) est fondée sur une combinaison de fouille de données et d’apprentissage automatique (Section 3.1). La seconde méthode comporte deux variantes (*run2* et *run3*) et est fondée sur une combinaison d’approche sac de mots et d’indices tirés de trois ressources lexicales (Section 3.2).

3.1 Méthode fondée sur la fouille de motifs (*run1*)

La méthode que nous décrivons ici exploite des chaînes de caractères répétées maximales (ou *rstr_{max}*). Cette méthode est fondée sur les travaux en algorithmique du texte menés notamment par Ukkonen (Ukkonen, 2009) et Kärkkäinen (Kärkkäinen *et al.*, 2006). Ces méthodes ont notamment été utilisées pour des tâches de classification sur des corpus multilingues dans le domaine de la surveillance épidémiologique (Lejeune *et al.*, 2015) ou encore de l’attribution d’auteur (Brixteel *et al.*, 2015). Les *rstr_{max}* sont des motifs définis par les caractéristiques suivantes :

- répétition : le motif apparaît au moins deux fois dans le corpus traité ;
- maximalité : le motif ne peut être étendu à gauche (maximalité à gauche) ou à droite (maximalité à droite) sans réduire le nombre d’occurrences.

Si l’on retranscrit ces caractéristiques dans la terminologie de la fouille de données, la maximalité correspond exactement au concept de fermeture. La répétition est assimilable à la notion de support minimal (ou *minsup*) à ceci près que l’on s’intéresse ici au nombre d’occurrences du motif dans le corpus et non au nombre de transactions (ici des *tweets*) qui supportent le motif.

Nous avons utilisé la même chaîne de traitement pour les trois tâches. Dans cette chaîne, T est l’ensemble du corpus à traiter composé de I *tweets* et où t_i est le $i^{\text{ème}}$ *tweet*. F désigne l’ensemble de traits calculables à partir de T et F_i l’ensemble des traits extraits du *tweet* t_i . Chaque *tweet* t_i est représenté sous la forme d’un vecteur de traits. Soit $o_{(i,j)}$ l’effectif du $j^{\text{ème}}$ trait dans le $i^{\text{ème}}$ *tweet* t_i , $0 \leq j < n$. Nous représentons t_i sous la forme $\{o_{(i,0)}, o_{(i,1)}, \dots, o_{(i,n-1)}\}$ où n est la taille de l’ensemble de traits F .

Dans la phase d’apprentissage nous avons éprouvé le système grâce à une validation croisée en dix strates. Nous avons testé différents classifieurs (SVM, réseaux bayésiens, arbres de décisions...) et

différents paramètres de filtrage des traits (longueur, fréquence relative, fréquence absolue). De ces expériences, nous avons retiré trois enseignements :

- Ajouter une contrainte de fréquence absolue maximale (ou support maximal) avait une influence très variable et globalement marginale sur les résultats ;
- Chercher à optimiser les intervalles de longueur pour les motifs nous semblait difficile à justifier tant les intervalles optimaux variaient selon les tâches ;
- Les meilleurs résultats étaient sans contestation possible obtenus avec des classifieurs de type SVM.

Pour la configuration de test, chaque classifieur est entraîné sur le sous-ensemble de *tweets* figurant dans le corpus d'entraînement puis testé sur le sous-ensemble de *tweets* figurant dans le corpus de test. L'ensemble des traits utilisés correspond à l'intersection des ensembles de traits du corpus de test et du corpus d'entraînement. Dans la version utilisée pour le défi, aucun filtrage, nous avons filtré les motifs en conservant ceux dont la longueur était comprise entre 3 et 7 caractères. Cet intervalle était le plus stable entre les différentes strates et les différentes tâches. Par ailleurs, cet intervalle correspond aux intervalles optimaux dans les travaux sur la stylométrie en langue française (Brixtel *et al.*, 2015). D'autre part, nous utilisons systématiquement un classifieur SVM *One VS Rest*. L'analyse des résultats sur les données de test a confirmé que les intervalles de longueur optimaux étaient variables. Les valeurs les plus efficaces sur le jeu d'entraînement n'étaient pas toujours les mêmes sur le jeu de test ; De plus, les différences dans les résultats parfois très ténues (parfois un ou deux vrais positifs en plus ou en moins). Néanmoins, dans les tableaux de résultats de la section 4 nous ajoutons la configuration *run1_{bis}* qui utilise les valeurs optimales pour chaque tâche sur le jeu de test.

3.2 Méthodes fondées sur des lexiques et des représentations en mots

Les méthodes utilisées pour les *run 2* et *3* sont très similaires. Elles ne diffèrent que par la manière dont les textes sont représentés. Dans le *run2*, les textes sont représentés comme un sac de mots, tandis que dans le *run3*, les textes sont représentés avec des plongements de mots dérivées à l'aide d'un modèle skip-gram entraînée avec WORD2VEC (Mikolov *et al.*, 2013) sur le corpus FRWAC (Baroni *et al.*, 2009), avec 100 dimensions. Les deux méthodes partagent 13 caractéristiques dérivées de plusieurs dictionnaires, ainsi que les méthodes de classification (machines à vecteurs de support) et la chaîne de traitement.

Dans la tâche 1 et la tâche 3, nous n'avons pas utilisé une stratégie *One VS Rest* pour le classifieur pour éviter le biais (en particulier vers la classe objective, majoritaire), donc nous avons plutôt construit un modèle différent pour chaque sous-tâche (objectivité, polarité mixte, polarité), en sélectionnant le sous-ensemble du training set plus pertinent à chaque tâche. Le modèle pour l'objectivité a été construit en utilisant le training set entier (3906 instances pour la tâche 1 et 5118 instances pour la tâche 3), le modèle pour la polarité mixte a été construit en utilisant la partie "non objective" du jeu d'entraînement (2263 et 3400 instances pour la tâche 1 et 3 respectivement), et finalement le modèle pour la polarité simple a été construit en utilisant un sous-ensemble composé uniquement des *tweets* étiquetés *positive* et *negative* (1763 et 2768 instances pour la tâche 1 et 3 respectivement). Les modèles obtenus sont appliqués séquentiellement, en conséquence nous réalisons d'abord la classifications des tweets en objectifs/subjectifs, les tweets subjectifs sont alors soumis au classifieur mixed/not mixed, et finalement ces derniers sont classés en positifs/négatifs.

Les caractéristiques dérivées des lexiques sont les suivantes :

- Moyenne des poids du lexique FEEL (Abdaoui *et al.*, 2016) (positifs et négatifs);
- Moyenne des poids du lexique labMT (Dodds *et al.*, 2011), traduit en français¹;
- 3 caractéristiques obtenues à partir de FixISS (Grezka, 2016) :
 - nombre d’adverbes dénotant des intensités;
 - nombre d’adverbes dénotant des négations;
 - nombre d’adverbes dénotant des interjections.
- 3 caractéristiques obtenues à partir de FEEL :
 - moyenne dans le premier tiers du tweet;
 - moyenne dans la partie centrale du tweet;
 - moyenne dans la partie finale du tweet.
- 3 caractéristiques obtenues à partir de labMT :
 - moyenne dans le premier tiers du tweet;
 - moyenne dans la partie centrale du tweet;
 - moyenne dans la partie finale du tweet.
- Somme des poids des emojis, en utilisant le lexique des sentiments des emojis (Novak *et al.*, 2015);
- Nombre d’émoticônes.

Pour la représentation en mots, nous avons décidé, pour réduire les dimensions des vecteurs des caractéristiques, de sélectionner un sous-ensemble des mots observés dans le jeu d’entraînement. En premier lieu, nous avons éliminé conjonctions, prépositions et articles. Ensuite, le critère adopté pour sélectionner les mots a été le suivant : étant donné la probabilité $p(w|c)$ d’observer un mot w dans une classe c et la probabilité $p(w|\hat{c})$ d’observer le même mot dans la classe complémentaire \hat{c} , alors on sélectionne pour le vecteur de caractéristiques uniquement les mots avec fréquence supérieure à 5 dans le training set, avec une quantité d’information $I(w) > 0.001$, où la quantité d’information est définie comme :

$$I(w) = p(w|c) (\log(p(w|c)) - \log(p(w|\hat{c}))) .$$

Le résultat de la sélection donne lieu à des vecteurs de 1735 caractéristiques, où 1722 sont les mots et 13 des caractéristiques dérivées des lexiques.

Pour la représentation en plongements de mots, un problème important à résoudre était de dériver un vecteur unique représentant le *tweet* à partir des vecteurs des mots. La solution adoptée, présentée dans (De Boom *et al.*, 2016), a été d’agréger les vecteurs des mots en utilisant le max et le min pour chaque coordonnée. Plus formellement, pour un *tweet* de n mots, on a n vecteurs $v_1, v_2, \dots, v_n \in \mathbf{R}^d$, où d est le nombre de dimensions, dans notre cas 100; on calcule les vecteurs $u^{min} = \min_{i \in 1 \dots d} (v_{1i}, v_{2i}, \dots, v_{ni})$ et $u^{max} = \max_{i \in 1 \dots d} (v_{1i}, v_{2i}, \dots, v_{ni})$, et on utilise leur concaténation $u = u^{min} \cdot u^{max} \in \mathbf{R}^{2d}$ pour représenter le *tweet* entier. Ainsi, dans ce cas, les vecteurs utilisés dans le classifieur sont composés de 213 caractéristiques, 200 obtenues à partir des plongements de mots et 13 dérivées des lexiques.

RUN	Classes	Précision	Rappel	F-mesure
run1 (3-7)	Micro	0.6219	N/A	N/A
	Macro	0.5387	0.5283	0.5312
	positive	0.5714	0.5203	0.5446
	negative	0.5847	0.6289	0.6060
	mixed	0.2613	0.1854	0.2169
	objective	0.7373	0.7785	0.7573
run2	Micro	0.5727	N/A	N/A
	Macro	0.4972	0.4823	0.4880
	positive	0.5098	0.4227	0.4622
	negative	0.5123	0.5849	0.5462
	mixed	0.2	0.2016	0.2008
	objective	0.7668	0.7201	0.7427
run3	Micro	0.5706	N/A	N/A
	Macro	0.5268	0.4854	0.4978
	positive	0.6233	0.3902	0.48
	negative	0.5158	0.5628	0.5383
	mixed	0.1987	0.2661	0.2275
	objective	0.7694	0.7226	0.7452
run1 _{bis} (3-100)	Micro	0.6373	N/A	N/A
	Macro	0.5693	0.5463	0.5545
	positive	0.6373	0.5285	0.5778
	negative	0.5821	0.6352	0.6075
	mixed	0.3077	0.2258	0.2605
	objective	0.75	0.7956	0.7721

TABLE 1 – Résultats obtenus sur la première tâche. Le run1_{bis} est une version du run1 utilisant les valeurs optimales pour cette tâche : longueur minimale de 3 caractères et longueur maximale de 100 caractères.

4 Résultats

4.1 Tâche 1 : Classification des tweets non figuratifs selon leur polarité

Nous présentons dans le tableau 1 les résultats officiels fournis par les organisateurs du défi. De manière globale, nous pouvons observer que la méthode $rstr_{max}$ (run1) est celle qui s’est le mieux comportée. La méthode fondée sur les plongements lexicaux (run3) a offert une meilleure précision, de l’ordre de plusieurs points de pourcentage, pour les classes *positive* et *objective*. Cette méthode a également offert des résultats intéressants en terme de rappel et de F-mesure pour la détection de la classe *mixed*, classe qui a posé le plus de problèmes à nos systèmes. La difficulté rencontrée pour cette classe composite explique la grande différence entre les résultats en configuration « micro » et les résultats en configuration « macro ». La configuration optimale utilisée pour le run1_{bis} permet d’améliorer la macro f-mesure de plus de deux points de pourcentage principalement du fait d’une amélioration importante des résultats sur la difficile classe *mixed*. Les valeurs optimales n’étaient pas les mêmes sur le jeu d’apprentissage montrant ainsi le manque de robustesse de cette tentative d’optimisation.

4.2 Tâche 2 : Identification du langage figuratif

Pour cette tâche, les classes étaient moins nombreuses et plus équilibrées. Nous nous attendions à ce que la méthode du *run3* fonctionne mieux que la méthode *rstr_{max}* que nous avons supposé moins robuste. L'écart est de 3 points de pourcentage en macro F-mesure ce qui n'est pas négligeable (Tableau). Il semble qu'assimiler la tâche de détection de langage figuratif à une tâche de stylométrie soit prometteur. Nous pouvons remarquer que là aussi il est possible d'améliorer significativement les résultats de la méthode en utilisant des contraintes de longueur pour les motifs.

RUN	Classes	Précision	Rappel	F-mesure
run1 (3-7)	Toutes (Micro)	0.7786	N/A	N/A
	Toutes (Macro)	0.7517	0.7418	0.7461
	figurative	0.6814	0.6311	0.6553
	nonfigurative	0.8221	0.8524	0.8370
run2	Toutes (Micro)	0.6741	N/A	N/A
	Toutes (Macro)	0.6259	0.6157	0.6190
	figurative	0.5131	0.4405	0.4740
	nonfigurative	0.7387	0.7909	0.7639
run3	Toutes (Micro)	0.7609	N/A	N/A
	Toutes (Macro)	0.7338	0.7069	0.7160
	figurative	0.6751	0.5450	0.6031
	nonfigurative	0.7925	0.8688	0.8289
run1 _{bis} (1-100)	Toutes (Micro)	0.7992	N/A	N/A
	Toutes (Macro)	0.7756	0.7654	0.7699
	figurative	0.7137	0.6639	0.6879
	nonfigurative	0.8376	0.8668	0.852

TABLE 2 – Résultats obtenus sur la deuxième tâche. Le *run1_{bis}* est une version du *run1* utilisant les valeurs optimales pour cette tâche : longueur minimale de 1 caractères et longueur maximale de 100 caractères.

4.3 Tâche 3 : Classification des tweets figuratifs et non figuratifs selon leur polarité

Les résultats obtenus sur cette tâche par nos méthodes sont présentés dans le tableau 3. Cette tâche est bien évidemment similaire à la tâche 1 et ceci se ressent au niveau des résultats globaux des équipes puisque la moyenne et la médiane sont assez proches. Nous retrouvons cette tendance sur nos trois *run* puisque l'écart entre ces deux tâches et de l'ordre de 1 ou 2 points de pourcentage en Macro F-mesure. Nous remarquons toutefois que le meilleur score (macro F-mesure) du défi est significativement plus faible pour cette tâche (0.5938) que pour la tâche 1 (0.6499). Nous aurions là aussi pu optimiser le filtrage des motifs (configuration *run1_{bis}*) mais il est intéressant de constater que les meilleures valeurs sont différentes de celles de la tâche 1.

RUN	Classes	Précision	Rappel	F-mesure
run1 (3-7)	Toutes (Micro)	0.6268	N/A	N/A
	Toutes (Macro)	0.5305	0.5166	0.5192
	positive	0.5398	0.488	0.5126
	negative	0.6430	0.7200	0.6794
	mixed	0.2333	0.1329	0.1693
	objective	0.7058	0.7255	0.7155
run2	Toutes (Micro)	0.5636	N/A	N/A
	Toutes (Macro)	0.4783	0.4864	0.4818
	positive	0.4328	0.464	0.4478
	negative	0.6122	0.5809	0.5962
	mixed	0.2108	0.1962	0.2032
	objective	0.6572	0.7046	0.6801
run3	Toutes (Micro)	0.5464	N/A	N/A
	Toutes (Macro)	0.4745	0.4830	0.4773
	positive	0.4444	0.48	0.4615
	negative	0.6112	0.5369	0.5716
	mixed	0.1708	0.2151	0.1904
	objective	0.6718	0.7	0.6856
run1 _{bis} (1-7)	Toutes (Micro)	<i>0.6354</i>	N/A	N/A
	Toutes (Macro)	<i>0.5465</i>	<i>0.5347</i>	<i>0.5376</i>
	positive	<i>0.5508</i>	<i>0.52</i>	<i>0.535</i>
	negative	<i>0.6576</i>	0.7200	<i>0.6874</i>
	mixed	<i>0.2647</i>	0.1709	<i>0.2077</i>
	objective	<i>0.713</i>	<i>0.7279</i>	<i>0.7204</i>

TABLE 3 – Résultats obtenus sur la troisième tâche. Le run1_{bis} est une version du run1 utilisant les valeurs optimales pour cette tâche : longueur minimale de 1 caractère et longueur maximale de 7 caractères.

5 Discussion et conclusion

Nous avons présenté dans cet article deux types de méthodes pour la classification de *tweets*. Nous avons développé deux méthodes fondées sur des ressources linguistiques et des représentations en mots (sac de mots et plongement lexical). Nous avons également développé une méthode indépendante de la langue fondée sur la détection de motifs fermés fréquents. L'originalité de cette méthode est que nous avons utilisé des motifs en caractères plutôt que des motifs en mots ou des motifs impliquant un pré-traitement des données (étiquetage morpho-syntaxique par exemple). Il serait intéressant de voir comment cette méthode se comporte dans un contexte multilingue dans la mesure où elle n'utilise pas de connaissance sur la langue. De façon plus générale, nous pouvons nous demander pourquoi ce genre de méthodes issues de la stylométrie se révèle efficace pour des tâches nécessitant *a priori* des connaissances linguistiques fines. Une des hypothèses est que les aspects stylistiques sont fondamentaux pour ce genre de tâche de fouille d'opinion. Ceci ouvre de nouvelles perspectives pour les méthodes dites alingues ou agnostiques de la langue (*language agnostic*) qui semblent en mesure d'être performantes sur des tâches de plus en plus diverses.

Références

- ABDAOUI A., AZÉ J., BRINGAY S. & PONCELET P. (2016). Feel : a french expanded emotion lexicon. *Language Resources and Evaluation*, p. 1–23.
- AMOSSY R. (2003). In *Les ironies balzacienes*, chapter Les fonctions argumentatives de l'ironie balzacienne, p. 143–154. Pirot : Saint-Cyr sur Loire.
- BARONI M., BERNARDINI S., FERRARESI A. & ZANCHETTA E. (2009). The wacky wide web : a collection of very large linguistically processed web-crawled corpora. *Language resources and evaluation*, **43**(3), 209–226.
- BENAMARA F., GROUIN C., KAROUJ J., MORICEAU V. & ROBBA I. (2017). Analyse d'opinion et langage figuratif dans des tweets : présentation et résultats du défi fouille de textes 2017. In *Actes de l'atelier DEFT de la conférence TALN 2017, 26 juin 2017, Orléans.*, p. 1–11, Caen, France : Association pour le Traitement Automatique des Langues.
- BERRENDONNER A. (1981). In *Eléments de pragmatique linguistique*, chapter De l'ironie, p. 173–239. Editions de Minuit : Paris.
- BERRENDONNER A. (2002). Portrait de l'énonciateur en faux naïf. *Semen*, **15**.
- BRES J. (2011). L'ironie, un cocktail dialogique ? *VNU Journal of Science : Foreign Studies*, **27**(3).
- BRIXTEL R., LECLUZE C. & LEJEUNE G. (2015). Attribution d'Auteur : approche multilingue fondée sur les répétitions maximales. In *Actes de la 22e conférence sur le Traitement Automatique des Langues Naturelles (TALN'2015)*, p. 208–219.
- CLARK H. & GERRIG R. (1984). On the pretense theory of irony. *Journal of Experimental Psychology : General*, **113**, 121– 126.
- DE BOOM C., VAN CANNEYT S., DEMEESTER T. & DHOEDT B. (2016). Representation learning for very short texts using weighted word embedding aggregation. *Pattern Recognition Letters*, **80**, 150–156.
- DODDS P. S., HARRIS K. D., KLOUMANN I. M., BLISS C. A. & DANFORTH C. M. (2011). Temporal patterns of happiness and information in a global social network : Hedonometrics and twitter. *PloS one*, **6**(12), e26752.
- DUCROT O. (1984). *Esquisse d'une théorie polyphonique de l'énonciation. Le dire et le dit.*, In *Rhétorique et argumentation : de l'ironie*, p. 171–233. Editions de Minuit : Paris, France.
- EGGS E. (2009). Rhétorique et argumentation : de l'ironie. *Argumentation et Analyse du Discours*, **2**(3).
- GREZKA A. (2016). La ressource multilingue fixiss : figement et séquences adverbiales. *Phraséologie et linguistique appliquée, Cahiers de lexicologie*, **108**, 75–90.
- GRICE H. (1979). Logique et conversation. *Communications*, **30**, 57–72.
- KÄRKKÄINEN J., SANDERS P. & BURKHARDT S. (2006). Linear work suffix array construction. *Journal of the ACM*, **53**(6), 918–936.
- KAROUJ J., BENAMARA F., MORICEAU V., PATTI V., BOSCO C. & AUSSENAC-GILLES N. (2017). Exploring the Impact of Pragmatic Phenomena on Irony Detection in Tweets : A Multilingual Corpus Study (regular paper). In *EACL-European Chapter of the Association for Computational Linguistics, Valence, 03/04/2017-07/04/2017*.
- KERBRAT-ORECCHIONI C. (1978). Problèmes de l'ironie. *Linguistique et sémiologie*, **2**, 10–46.

- KERBRAT-ORECCHIONI C. (1980). L'ironie comme trope. *Poétique*, **41**, 108–127.
- KERBRAT-ORECCHIONI C. (1986). *L'implicite*. Paris, France : Armand Colin.
- LEJEUNE G., BRIXTEL R., DOUCET A. & LUCAS N. (2015). Multilingual event extraction for epidemic detection. *Artificial Intelligence in Medicine*. doi : 10.1016/j.artmed.2015.06.005.
- MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, p. 3111–3119.
- NOVAK P. K., SMAILOVIĆ J., SLUBAN B. & MOZETIČ I. (2015). Sentiment of emojis. *PloS one*, **10**(12), e0144296.
- PERRIN L. (1996). *L'ironie mise en tropes : du sens des énoncés hyperboliques et ironiques*. Paris : Kimé.
- RECANATI F. (2000). *Oratio Obliqua, Oratio Recta : The Semantics of Metarepresentations*. Cambridge, MA : MIT Press.
- RECANATI F. (2004). *Literal Meaning*. Cambridge, UK : Cambridge University Press.
- SCHOENTJES P. (2001). *Poétique de l'ironie*. Paris, France : Seuil.
- SPERBER D. & WILSON D. (1978). Les ironies comme mention. *Poétique*, **36**, 399–412.
- SPERBER D. & WILSON D. (1998). *Relevance Theory : Applications and Implications*, In *Irony and Relevance : a reply to Seto, Hemmamoto and Yamanashi*, p. 283–293. Carston R., Uchida S. : Amsterdam : John Benjamins.
- UKKONEN E. (2009). Maximal and minimal representations of gapped and non-gapped motifs of a string. *Theoretical Computer Science*, **410**(43), 4341–4349.
- WILSON D. (2006). The pragmatics of verbal irony : Echo or pretence ? *Lingua*, **116**, 1722–1743.