Etude expérimentale de classification textuelle multi-étiquette pour la relation client

Gil Francopoulo*, Léon-Paul Schaub**
Lynda Ould Younes***

*AKIO
gfrancopoulo@akio.com,
http://www.akio.com
**AKIO + LIMSI-CNRS
lpschaub@akio.com
***AKIO
louldyounes@akio.com

Résumé. La gestion de la relation avec les clients (GRC ou CRM selon le sigle anglais) est l'analyse des interactions des clients. Notre étude porte sur l'analyse du sens des textes pour en synthétiser les opinions et les sujets abordés par des clients qui s'expriment en plusieurs langues. L'approche de cette classification consiste à annoter les documents en différentes langues avec le même jeu de catégories, sachant que l'annotation est faite en une langue dite source ou native et qu'ensuite des algorithmes d'apprentissage automatique sont appliqués aux autres langues qui sont désignées comme les langues cibles ou non natives. Nous avons essayé différentes stratégies et comparé les options avec ou sans traitements linguistiques, de même que les différents algorithmes qu'ils soient neuronaux ou non. Les résultats de notre étude prouvent l'efficacité de notre approche quand elle est appliquée à des logiciels opérationnels.

1 Introduction

La gestion de la relation client est l'analyse des données des interactions des clients que l'on désigne sous le vocable de "voix du client'. Dans les systèmes modernes, le client a le choix entre différentes langues et canaux de communication comme le courrier électronique, le téléphone, le tchat, les médias sociaux et différentes enquêtes de satisfaction. À l'autre extrémité de l'analyse des données, les managers, les data-analystes et personnels du marketing qui consultent et paient pour le service veulent aussi avoir le choix entre une vaste gamme d'options sur le niveau de détail depuis le tableau de bord synthétique jusqu'à une sélection portant sur un sujet précis avec une classification à grains fins. Toutes ces navigations doivent être implémentées de façon à réagir à la volée en temps réel. Cela signifie qu'à la différence d'autres travaux, notre objectif n'est pas de catégoriser un document comme étant globalement bon ou mauvais (Pang et Lee, 2005). Ce n'est pas suffisant. Nous avons besoin de scruter profondément le sens des textes afin d'en extraire des indices sémantiques relativement fins comme le

sujet sur lequel porte l'option. Cela ne signifie pas que tous ces détails seront nécessairement présentés au lecteur final en toutes circonstances ; en fait, ces catégories peuvent être agrégées en des catégories plus abstraites, mais ces actions sont réalisées au sein d'un système de "business intelligence" qui est hors du champ du présent article. Notre étude se focalise donc sur la classification à grains fins. La ligne directrice de nos utilisateurs est : il faut comprendre pour pouvoir agir.

La classification monolingue de texte est l'affectation automatique d'une ou plusieurs catégories sémantiques dans une langue donnée. La catégorisation interlingue (Bel et al., 2003) est l'affectation des catégories à un texte d'une autre langue. Le raisonnement sous-jacent est de réduire le coût d'annotation à une unique langue considérée comme la langue native, et puis, à partir de cette langue, d'appliquer des algorithmes d'apprentissage automatique à d'autres langues qui sont considérées comme non natives ou langues cibles.

2 Contexte industriel

Notre société AKIO opère dans différents secteurs d'activité et les ressources associées à la personnalisation peuvent varier d'un secteur à l'autre. Ces domaines sont bien identifiés : 1) hôtels-restaurants 2) transport aérien 3) banque 4) sites de rencontres 5) e-commerce et boutiques 6) assurance. Les langues couvertes sont le français (considéré comme la langue native), l'anglais (natif mais moins développé donc non utilisé lors de l'apprentissage), l'espagnol, l'allemand, le portugais et l'italien. Le logiciel s'appelle AKIO Analytics. Toutes les paires secteur-langue sont implémentées selon la même stratégie avec des résultats similaires, mais pour la simplicité de l'exposé, nous nous focaliserons sur la paire e-commerce et boutiques pour l'espagnol.

3 Travaux connexes

Un certain nombre de classifications interlingues soit nécessitent des corpus parallèles soit ont besoin de documents annotés à la fois dans la langue source et cible (Xiao et Guo, 2013). Le problème principal est le manque de ressources à la fois fiables, diverses et volumineuses. La catégorisation dans de multiples langues peut être résolue en transférant la connaissance depuis une langue bien dotée vers une langue peu dotée. C'est pourquoi la plupart des systèmes emploient des ressources lexicales anglaises telles que SentiWordNet comme décrit dans l'état de l'art de (Dashtipour et al., 2016) et transfèrent vers d'autres langues. Certaines méthodes ne sont pas flexibles dans le choix des catégories tout au long de leur cycle de vie : en changeant les catégories, l'annotation sera à refaire ou bien nécessitera de gros efforts de transcodage manuel. Certaines approches sont basées sur des pivots entre la source et la cible. Ces pivots peuvent agir comme des filtres pour la classification et peuvent être utilisées dans du co-apprentissage comme dans (Wei et Pal, 2010). Une autre stratégie consiste à utiliser le texte traduit terme à terme avec un dictionnaire bilingue et ensuite à augmenter une sélection de documents de la langue source. Puis, un algorithme comme LSA (Latent Semantic Analysis) est appliqué pour obtenir une représentation interlingue (Gliozzo et Strapparava, 2006). Mais, pour autant que l'on sache, ces systèmes sont principalement appliqués à l'analyse d'opinions où la phrase est facilement transformable dans des pivots au sein de ces différentes langues. La

difficulté pour nous est de prendre en compte les phrases mal formées, les formes idiomatiques, l'ironie et les insultes, ce qui est très fréquent. La même remarque peut être faite à propos de CLESA (Cross-lingual Explicite Semantic Analysis) (Song et al., 2016), (Sorg et Cimiano, 2012). Nous faisons différemment, comme nous allons le voir par la suite.

4 Prétraitement et corpus

Concernant le prétraitement, nous avons développé en interne un pipeline linguistique robuste comprenant un tokeniseur, un correcteur orthographique et grammatical, un tagger-chunker statistique (Francopoulo, 2008), un analyseur syntaxique en dépendance, un annotateur de la négation et un détecteur d'entités nommées associé à un résolveur de co-référence. L'entrée peut être de niveau grand public avec d'importantes variations par rapport à la façon standard de s'exprimer. L'objectif est de normaliser l'entrée autant que possible. Tous ces outils sont optionnels dans le sens où, lors de nos expérimentations nous avons l'option de les utiliser ou non en fonction de l'évaluation globale. Ainsi, l'entrée peut être de quatre types : a) la chaîne brute d'origine, b) une suite de formes fléchies corrigées, c) une suite pleine de formes lemmatisées, encore appelées lemmes, d) une suite filtrée de lemmes corrigés. Précisons que les caractères tabulation et guillemet sont nettoyés pour tous les niveaux d'entrée car ce sont des caractères qui posent des problèmes de format pour certains logiciels et comme nous ne voulons pas introduire de biais en faveur d'un algorithme, nous les avons enlevés pour tous les logiciels. Le filtrage des lemmes consiste à ne prendre que les parties du discours comme les noms ou les adverbes de négation et d'ignorer d'autres mots comme les déterminants. On notera que le lemme est désambiguïsé car il est le résultat de la correction, tagging, chunking et de la résolution des entités nommées.

Concernant le système natif français, la classification est un système fondé sur des règles qui est accroché à la sortie du pipeline. En partant des lemmes corrigés, l'objectif est d'annoter le texte avec un ensemble de catégories. Le classifieur n'est pas un simple jeu de règles à plat, mais plutôt un système complexe fondé sur une organisation hiérarchique de composants sémantiques. Il y a trois types de composants : a) des composants transversaux qui sont valides pour tous les secteurs, b) une vingtaine de composants factorisables et c) six composants spécifiques aux secteurs. Un composant factorisable est par exemple le composant de la livraison qui est importé par le secteur du ecommerce (pour recevoir une robe) et par les restaurants (pour recevoir un repas) mais pas par le secteur des sites de rencontre. L'objectif est de partager les règles entre les secteurs afin de faciliter l'évolution et la maintenance. Le système est doté d'un dictionnaire de synonymes fondé sur CRISCO [1]. En prenant en compte l'héritage (donc en aplatissant virtuellement les composants pour un secteur), le secteur du e-commerce mobilise 9 000 règles. Le système est complexe mais bien organisé donc il est gérable et évolutif. La totalité du système, tous secteurs confondus comporte 17 500 règles.

Nous n'avons pas trouvé de corpus public pour notre étude, de ce fait, nous avons collecté un corpus par nous-mêmes. Notre corpus est constitué de nombreux textes venant de différents clients et canaux. Aucune modification manuelle n'est effectuée sur le contenu originel. La taille de chaque verbatim est petite, typiquement d'une longueur de cinq lignes. La plupart des documents sont des plaintes, le deuxième type de documents étant des questions qui sont sou-

1. http://crisco.unicaen.fr/des

vent des demandes de renseignement. La source étant le grand public, les textes contiennent de nombreuses fautes d'orthographe, des idiomes informels avec fréquemment une combinaison de tournures ironiques et d'insultes. Dit en d'autres termes, l'entrée est de mauvaise qualité et quelquefois, même pour un lecteur humain, le texte est difficile à comprendre. Les textes sont très différents de la Presse ou de Wikipedia tels qu'on peut les trouver dans des études comme (Zaid et al., 2017) par exemple. Le coût de l'annotation manuelle étant très élevé, nous adoptons une stratégie hybride afin de collecter une grande masse de données annotées. Une partie est annotée manuellement et le reste est engendré automatiquement à la manière des préparations de reconnaissance d'images au sein desquelles les images sont retournées pour faire grossir le corpus annoté. Nous gérons trois sous-corpus. La première sous-partie (appelée le corpus gold) est annotée par le système des règles et est constamment maintenue à l'aide d'un outil de vérification des tests de non régression. Le corpus gold comporte lui-même deux sous-parties : 90% pour la partie apprentissage et 10% pour le test. Le deuxième corpus (le corpus bronze) est automatiquement engendré depuis le corpus gold avec substitution des synonymes les plus fréquents en utilisant CRISCO. Pour éviter un biais dans le sens positif, nous ne créons pas de textes automatiquement pour les inclure dans le corpus de test car ils sont trop similaires aux textes du gold, donc seuls les textes de la partie apprentissage sont engendrés. Pour les corpus gold et bronze, le prétraitement et la classification ont été développés de manière à obtenir une annotation parfaite, ainsi la F-mesure est de 100% [2]. Le troisième corpus est un corpus du même domaine après application du système de règles. Le système français étant aux alentours d'une F-mesure de 85% sur les documents inconnus du même domaine, l'objectif est d'agrandir la taille du corpus même s'il existe une petite pénalité sur la qualité. Pour conclure, nous avons ainsi constitué un corpus d'une taille moyenne qui est représentative de l'activité de la relation client, comme présenté dans le tableau 1.

sous-corpus	mécanisme de constitution en français	FM	verbatim#	mots#
gold	développement manuel et écriture de règles	100	8 757	439 201
bronze	expansion automatique de synonymes depuis gold	100	15 269	1 299 180
silver	application automatique des règles	85	15 028	636 826
total			39K	2,4M

TAB. 1 – *Corpus.*

5 Catégories

Nous gérons trois types de catégories : les modalités d'expression, les thèmes et les opinions. Ces types ont été spécifiés après des études approfondies des flux de nos clients en complétant par les lectures de (Liu, 2012) et SentiWordNet (Esuli et Sebastiani, 2006), (Cambria et al., 2010). Comme il peut être observé dans le tableau 2, les catégories sont précises et nombreuses. La liste est relativement stable dans le temps, mais elle peut varier légèrement

2. Cela peut sembler inhabituel dans le cadre d'une annotation de corpus, mais le gold qualifie le système au sens industriel du terme dans la mesure où il fait fonction de test de non régression tout au long du cycle de vie. Ainsi, si des modifications sont apportées, le système ne sera pas mis en production si l'un des tests échoue

après une étude éditoriale prudente. Chaque texte peut être annoté par zéro (ce qui est rare), une ou plusieurs catégories. Il est à noter qu'il existe des opinions spéciales qui s'appellent NoSpecificTopicNeg et NoSpecificTopicPos utilisées pour annoter les rares situations où le locuteur ne donne aucune justification, thème ou détail comme "c'est nul" mais exprime clairement une opinion. Ajoutons aussi que l'opinion sur un thème est souvent désignée comme ABSA (aspect-based sentiment analysis) (Liu, 2012) dans la littérature scientifique et plusieurs études ont été menées sur les revues de films (Thet Tun et al., 2010), les produits électroniques (Hu et Liu, 2004), (Brody et Elhabad, 2010), les services (Long et al., 2010) et les restaurants (Ganu et al., 2009). Comparativement à SemVal sur ABSA, ce que nous nommons "thème" est appelé "Aspect Category Detection" et ce que nous appelons "opinion" est nommé "Aspect Category Polarity" (Pontiki et al., 2014).

type	définition	exemple	catégories#
modalité	forme générale de l'expression	injonction, question	4
thème	sujet de l'expression du locuteur	StoreDelivery, BankTransfer	117
opinion	jugement personnel du locuteur portant sur un thème et exprimé selon une polarité négative ou positive	MissingItemNeg, PricePos	58
total			179

TAB. 2 – *Catégories.*

6 Architecture du transfert interlingue

Concernant le flux de données, rappelons que nous n'avons pas de corpus parallèle. En outre, nous ne voulons pas passer du temps à annoter les corpus non natifs. Au moins deux architectures sont possibles : la première option consiste à traduire le texte espagnol en français lors de l'exploitation, et ensuite, à appliquer immédiatement le catégoriseur natif. La seconde option est de traduire un corpus de développement (lors de la phase de développement) puis d'apprendre et personnaliser un modèle de classification. Lors de l'exploitation, ce modèle est appliqué, cette phase étant nommée phase d'inférence. Le volume des messages est plutôt élevé : il peut être de 100K par jour, ainsi le coût en termes de service de traduction et de consommation CPU étant trop élevé avec la première option, nous adoptons la seconde option. Donc en résumé, nous faisons de la classification monolingue (en espagnol) au sein d'un système interlingue (français espagnol). Notre architecture de transfert est présentée dans la figure 1 avec Google Translation pour la traduction vers l'espagnol.

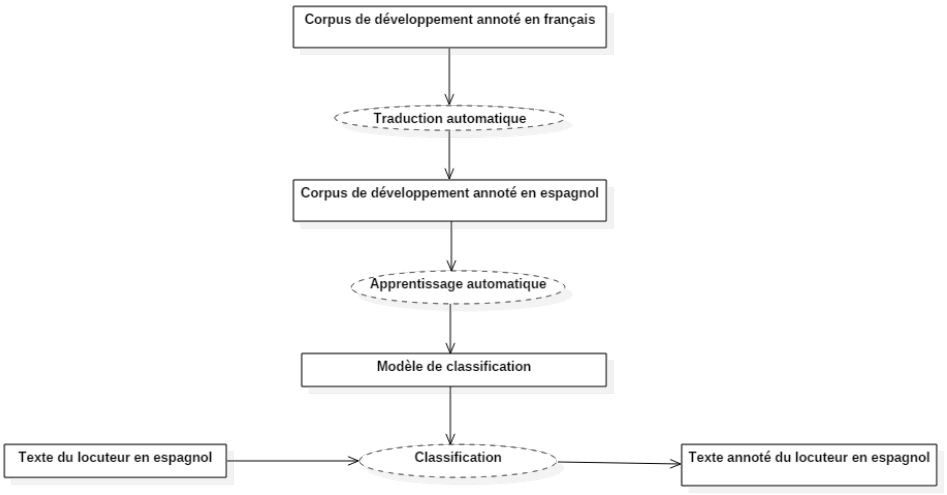

FIG. 1 – *Flux de transfert.*

7 Format d'entrée et algorithmes

Les options sur lesquelles nous pouvons agir sont :
— le niveau linguistique : chaîne brute, formes fléchies corrigées, lemmes corrigés, lemmes filtrés corrigés,
— le choix de l'algorithme.

Les algorithmes peuvent avoir les propriétés intrinsèques suivantes :
— sac de mots (bag of words / BOW) versus plongement de mots (word embedding / WE),
— la capacité de multi-étiquette (à la fois multi-classe et multi-label) : soit un modèle unique est capable de prendre en charge plusieurs catégories durant la phase d'apprentissage, soit il n'en est pas capable.

Nous avons étudié les algorithmes suivants : NB, classifieur SGD, SVM, SMO, FastText, CNN et BiLSTM.

NB, classifieur SGD, SVM et SMO sont des algorithmes linéaires d'apprentissage automatique. Nous utilisons l'implémentation de Weka [3]. NB est utilisé comme option de base avec ses hyper-paramètres par défaut. Ainsi, à la différence des autres algorithmes, aucune personnalisation des hyper-paramètres n'est effectuée pour NB. Le classifieur SGD de Weka implémente la descente de gradient stochastique avec ses paramètres par défaut : la fonction hinge loss avec un taux d'apprentissage de 0,01 est la meilleure combinaison pour nos données. SVM est une encapsulation de LIBSVM (Dashtipour et al., 2016), après réglage, la fonction noyau linéaire et un type C-SVM est le meilleur choix. SMO implémente l'optimisation séquentielle minimale de John Platt pour ensuite appeler un noyau SVM. Nous adoptons l'option de normalisation et un paramètre de tolérance de 0,001. Le classifieur SGD, SVM et SMO ne permettant

3. https://www.waikato.ac.nz/ml/weka

nom	nom complet	bibliothèque	version	origine
NB	Naive Bayes	Weka	3-8-3	Univ. de Waikato
classifieur SGD	Stochastic Gradient Descent	Weka	3-8-3	Univ. de Waikato
SVM	Support Vector Machine	Weka+LIBSVM	3-8-3	Univ. de Waikato
SMO	Sequential Minimal Optimization	Weka	3-8-3	Univ. de Waikato
FastText	FastText	FastText	0-3	Facebook
CNN	Convolutional Neural Network	TensorFlow	1-10	Google
BiLSTM	Bi-directional Long Short-Term Memory	TensorFlow	1-10	Google

TAB. 3 – *Algorithmes.*

pas le traitement multi-étiquette, nous effectuons un apprentissage binaire et de multiples inférences sont appliquées en séquence, donc la distribution des catégories est considérée comme indépendante. Une description complète de ces algorithmes peut être consultée dans le livre Weka (Witten et al., 2016).

FastText [4] est une régression logistique multi-classe à partir des moyennes des enchâssements des n-gram de caractères (Joulin et al., 2017). Ces enchâssements sont appris au préalable avec l'extension morphologique de skip-gram avec échantillonnage négatif (Bojanowski et al., 2017). L'apprentissage préalable est effectué sur un sur-ensemble deux fois plus vaste de nos corpus avec des textes du secteur d'activité concerné. Nous utilisons les paramètres par défaut car nous n'avons pas observé d'améliorations en faisant varier les réglages.

CNN est un réseau de neurones profond comprenant différentes couches comme les couches de convolution, des couches complètement connectées et des couches de normalisation (Kim, 2014). Nous utilisons TensorFlow [5] avec 300 époques pour une taille de batch de 128.

BiLSTM est un algorithme neuronal profond basé sur des unités de réseaux neuronaux récurrents (Hochreiter et Schmidhuber, 1997). Nous utilisons la fonction sigmoïd, la cross-entropie comme fonction de perte, ADAM pour l'optimisation et 300 époques pour une taille de batch de 128.

CNN et BiLSTM sont présentés selon deux formes : la première sans corpus pré-entraîné et la seconde (notée CNN-W2V et BiLSTM-W2V) avec le corpus pré-entraîné sur le Wikipedia espagnol [6].

8 Méthodologie d'expérimentation

Le corpus gold est découpé en 90% pour l'apprentissage et 10% pour le test. Les corpus bronze et silver sont utilisés à 100% pour l'apprentissage comme indiqué dans le chapitre sur le corpus. Pour déterminer les valeurs des hyper-paramètres mentionnées au paragraphe précédent, nous avons effectué une "grid selection" sur le corpus gold. Nous le faisons uniquement

4. https://fasttext.cc
5. https://www.tensorflow.org
6. https://github.com/facebookresearch/fastText/blob/master/pretrained-vectors.md

sur le gold car nous n'avons pas d'autre choix du fait du nombre trop élevé de calculs que cela nécessite. Nous ne faisons varier que les paramètres les plus significatifs en développant toutes les valeurs pour les énumérations symboliques, en énumérant les puissances de 2 pour les grandes valeurs et par puissance de 10 pour les petites valeurs. Nous limitons chaque session à une semaine de calcul maximum pour chaque algorithme. Une fois fixés les hyper-paramètres, nous évaluons sur le gold, bronze et silver.

9 Comparaison des temps d'apprentissage et d'inférence

Les vitesses des différents algorithmes sont extrêmement variables. D'un point de vue scientifique, le temps de calcul n'est pas essentiel, mais il est important au plan pratique car plus l'algorithme est rapide, plus on peut faire d'essais de réglage. D'autre part, cela facilite la mise en place industrielle car nous avons 6 secteurs d'activité pour 4 langues, ce qui fait 24 sessions. Certaines implémentations utilisent les cartes graphiques basées sur les coeurs CUDA alors que d'autres ne le font pas. Tous les apprentissages CUDA sont calculés sur NVIDIA GTX-1080. Les apprentissages sont réalisés sur un Xeon W2155 en utilisant 12 fils d'exécution. Toutes les inférences sont effectuées sur un seul fil d'exécution. Le tableau 4 donne les temps pour deux niveaux linguistiques d'entrée : les chaînes brutes et les lemmes filtrés corrigés. Les mesures des autres niveaux ne sont pas présentées mais figurent entre ces deux extrêmes. Deux considérations expliquent ces différences : premièrement, quand on procède avec des lemmes corrigés, la diversité des indices pour l'apprentissage automatique est plus petite, et deuxièmement, après filtrage pour chaque phrase le nombre de mots est plus petit.

nom	multi-étiquette	ordre	CUDA	temps d'appr. chaînes brutes	temps d'appr. lemmes corrigés	temps d'inférence
NB	non	BOW	non	2 h 30	50 mn	7 mn
SGD	non	BOW	non	6 h	2 h	31 s
SVM	non	BOW	non	4 h 50	1 h 44	39 s
SMO	non	BOW	non	5 jours 16 h	15 h	21 s
FastText	oui	WE	non	15 mn	15 mn	2 s
CNN	oui	WE	oui	1 h 3	37 mn	2 s
BiLSTM	oui	WE	oui	27 h	14 h 30	10 s

TAB. 4 – *Temps de traitement.*

10 Comparaison de la qualité selon le niveau linguistique et choix des options

Les mesures de qualité des différentes sessions de calcul sont présentées dans le tableau 5 en fonction du niveau linguistique de l'entrée de la catégorisation, avec les conventions que R signifie rappel, P précision et FM F-mesure, définie comme la moyenne harmonique de R et P. Pour faciliter la lecture, les FM au dessus de 70 sont entourées.

nom	chaînes brutes			formes fléchies corrigées			lemmes corrigés non filtrés			lemmes corrigés filtrés		
	R	P	FM	R	P	FM	R	P	FM	R	P	FM
NB	68	19	30,0	74	21	32,6	72	22	34,2	73	25	37,4
SGD	68	79	72,9	70	76	73,3	69	73	71,0	69	72	70,4
SVM	58	87	69,8	57	87	68,7	50	88	64,0	48	87	61,6
SMO	68	83	74,4	70	81	75,2	67	78	72,2	65	79	71,3
FastText	45	61	51,6	45	55	49,7	44	47	45,7	45	48	46,4
CNN	67	40	50,8	65	36	46,8	70	38	49,4	69	32	44,7
BiLSTM	74	36	48,7	76	37	50,0	77	38	51,4	78	40	53,2
CNN-W2V	72	32	45,2	72	29	41,8	73	31	44,3	72	30	42,4
BiLSTM-W2V	79	47	59,6	78	48	59,8	78	47	59,1	81	48	60,4

TAB. 5 – *Qualité.*

Tout d'abord, après avoir écarté NB, nous observons que SGD, SVM, SMO, FastText donnent une meilleure précision en comparaison du rappel. C'est le contraire pour CNN et BiLSTM. Maintenant, concernant strictement la FM pour SVM, SMO, FastText, CNN, l'entrée brute donne de meilleures résultats que les lemmes filtrés. Pour BiLSTM, c'est le contraire. Pour SGD, les valeurs sont très proches. Concernant la différence avec ou sans corpus préentraîné, la différence de l'ajout est positive pour BiLSTM mais elle est négative pour CNN. On observe aussi que SMO est toujours meilleur que SVM même si la différence interne entre les deux n'est que l'ajout d'un pré-traitement avec ensuite l'appel à un noyau SVM. Il faut maintenant évaluer le surapprentissage pour déterminer si certaines options produisent des modèles qui sont trop étroitement liés aux données d'apprentissage et ne prévoient pas les observations futures du même secteur d'activité. Nous procédons selon une intervalidation de 10 plis. Ainsi, les jeux de tests sont déplacés par fractions d'un dixième du corpus gold et l'apprentissage est recalculé sur les 90% restants en ajoutant le bronze et le silver. Ensuite nous faisons la moyenne que nous comparons avec la valeur initiale du tableau 5. Comme cela prend 10 fois plus de temps que l'évaluation en un pli, nous n'avons pas eu le temps de refaire les calculs pour la totalité des options. Nous l'avons fait pour SGD, car il est rapide à la fois pour les chaînes brutes et les lemmes filtrés. Pour les chaînes brutes, la FM passe de 72,9 à 69,8, pour les lemmes filtrés, la FM passe de 70,4 à 70,0. On observe donc que l'apprentissage sur les chaînes brutes a tendance à se comporter en surapprentissage, tout ceci avec un temps de calcul trois fois plus long.

Concernant le choix des options pour la mise en production, si on se focalise sur les FM au dessus de 70, l'option des formes brutes avec SMO n'est pas réaliste car le temps d'apprentissage est trop long pour les moyens de calcul dont nous disposons actuellement. En l'état actuel de nos évaluations, nous optons pour le classifieur SGD avec les lemmes corrigés filtrés, étant entendu que ce choix pourrait être remis en question à la lumière de futurs développements.

11 Discussion

Jusqu'à présent, nous n'avons pas parlé des variations que nous avons testées et qui n'ont pas donné de bons résultats comme la prise en compte des entités nommées, soit en les enlevant, soit en les remplaçant par leur type, par exemple : nom de société, de ville ou de personne. Cela ne donne pas une différence statistique significative. Actuellement, nous n'inscrivons que la valeur "nom propre" dans le filtrage des lemmes. D'autre part, nous avons essayé de créer des corpus via des allers-retours de traduction pour construire automatiquement des corpus similaires au gold en bénéficiant des annotations du français. Nous avons essayé vers l'espagnol ainsi que vers l'anglais. Cela n'a pas donné de bons résultats car même si l'on pouvait comprendre le sens, ce ne sont pas des formes qu'un francophone écrirait. A contrario, pour le corpus bronze (celui des synonymes) et silver (celui qui est automatique) : les gains ont été respectivement de 6% et 5%, ce qui explique que nous combinons maintenant gold, bronze et silver. Concernant les algorithmes, nous avons essayé de combiner BiLSTM avec deux couches de CNN, mais la qualité n'était pas bonne.

A cause du manque de place, nous ne pouvons pas justifier en détail toutes nos choix d'architecture. L'une de celles-ci concerne la flexibilité de choisir l'espace d'annotation qui n'est pas complètement figé et nécessite d'être adapté légèrement de temps en temps. Avec notre architecture, tout est contrôlé par le système natif associé avec des tests stricts de non régression. Comparé avec d'autres architectures où les catégories sont réparties sur plusieurs langues ou sur de nombreux documents de manière incontrôlée, notre système est facilement gérable car il est localisé au sein du système natif, la distribution en direction des systèmes non natifs étant entièrement automatique. En contre-partie, une des faiblesses de notre approche est que si dans une langue, les clients ont des préoccupations qui ne sont pas exprimées dans la version française pour des raisons culturelles, nous ne traitons pas ces préoccupations, puisque tout part du français. Jusqu'à présent, ce n'est pas un problème que nous avons rencontré. Si tel était le cas, alors nous devrions inclure ces formes dans la langue source et ensuite, relancer les calculs. Une autre limitation est que la qualité globale dépend de la qualité de la traduction automatique. A ce propos, lors des tests, nous avons détecté une dizaine de documents avec une mauvaise traduction. Nous avons implémenté un mécanisme d'exception pour prendre en compte la traduction manuelle rectificative. Nous n'avons pas eu le temps de travailler sur ce sujet qui concerne principalement la traduction des injures et des formes idiomatiques pour lesquelles Google Translation n'est pas très bon. Nous avons simplement observé qu'il était préférable de fournir à Google Translation non pas la forme initiale, mais la forme corrigée dans la mesure où nous disposons d'un correcteur orthographique bien adapté à notre domaine.

12 Futurs développements

Il se pourrait que les algorithmes neuronaux donnent de meilleurs résultats avec plus de données. De ce fait, nous allons continuer à faire grossir les corpus tout en continuant à comparer les options. D'autre part, il va falloir mesurer le surapprentissage pour les meilleures options, mais les temps de calculs sont très importants. Pour CNN et BiLSTM, nous prévoyons aussi de comparer la qualité avec un modèle construit à partir des données dont nous disposons dans le domaine de la relation client au lieu d'utiliser Wikipedia. Pour les systèmes neuronaux, nous avons fait quelques essais avec un mécanisme d'attention servant de première couche de

chaque réseau et d'entrée pour la première couche que ce soit pour CNN ou BiLSTM, mais pour l'instant les essais ne sont pas très concluants : d'autres tentatives sont nécessaires. Enfin, suite à nos recherches, nous avons remarqué que plusieurs systèmes désormais utilisaient en dernière couche d'activation des CRF afin d'avoir des probabilités markoviennes et ainsi une meilleure représentation du texte. Concernant les canaux d'entrée, actuellement l'entrée est purement textuelle en différents genres. Pour l'instant, nous traitons tous les canaux de la relation client sauf le téléphone mais nous prévoyons de travailler sur l'intégration d'un module de parole vers texte (STT) car la voix reste un canal majeur en termes de volume d'interactions.

13 Conclusion

Dans la présente étude, nous avons donc décrit une série d'expériences qui montrent qu'il est possible de classifier les documents en de multiples langues à partir d'un espace d'annotation français. Nos résultats ne sont pas très loin de l'état de l'art qui est de 75% comme présenté dans (Banea et al., 2010), malgré le fait que la qualité en entrée de nos textes est bien moindre que la leur, ce qui rend la tâche plus difficile. Notre étude soulève une question importante pour le traitement automatique des langues : a t-on besoin d'effectuer des prétraitements linguistiques avec les outils traditionnels, sachant que l'apprentissage automatique profond peut constuire des couches internes qui sont plus ou moins équivalentes à ce qui est construit par les outils traditionnels ? Ces idées sont mentionnées par exemple dans (Dias et al., 2018). Nous pensons qu'il est un peu trop tôt pour trancher.

Références

Banea, C., R. Mihalcea, et J. Wiebe (2010). Multilingual subjectivity : Are more languages better ? *Proceedings of COLING*.

Bel, N., C. Koster, et M. Villegas (2003). Cross-lingual text categorization. *Research and Advanced Technology for Digital Libraries*.

Bojanowski, P., E. Grave, A. Joulin, et T. Mikolov (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*.

Brody, S. et N. Elhabad (2010). An unsupervised aspect-sentiment model for online reviews. *Proceedings of NAACL*.

Cambria, E., R. Speer, C. Havast, et A. Hussain (2010). Senticnet : A publicly available semantic resource for opinion mining. *Proceedings of AAAI Fall Symposium*.

Dashtipour, K., S. Poria, A. Hussain, E. Cambria, A. A. Hawalah, A. Gelbukh, et Q. Zhou (2016). Multilingual sentiment analysis : State of the art and independent comparison of techniques. *Cognitive Computation*.

Dias, C.-E., C. Gainon de Forsan de Gabriac, V. Guigne, et P. Gallinari (2018). RNN et modèles d'attention pour l'apprentissage de profils textuels personnalisés. *Proceedings of CORIA*.

Esuli, A. et F. Sebastiani (2006). Sentiwordnet : A publicly available lexical resource for opinion mining. *Proceedings of LREC*.

Francopoulo, G. (2008). Tagparser : well on the way to ISO-TC37 conformance. *Proceedings of the International Conference on Global Interoperability*.

Ganu, G., N. Elhadad, et A. Marian (2009). Beyond the stars : Improving rating predictions using review text content. *Proceedings of WebDB*.

Gliozzo, A. et C. Strapparava (2006). Exploiting comparable corpora and bilingual dictionaries for cross-language text categorization. *Proceedings of the ICCL-ACL*.

Hochreiter, S. et J. Schmidhuber (1997). Long short-term memory. *Neural Computation*.

Hu, M. et B. Liu (2004). Mining and summarizing customer reviews. *Proceedings of AAAI*.

Joulin, A., E. Grave, P. Bojanowski, et T. Mikolov (2017). Bag of tricks for efficient text classification. *Proceedings of the European Chapter of ACL*.

Kim, Y. (2014). Convolutional neural networks for sentence classification. *Proceedings of EMNLP*.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Morgan and Claypool.

Long, C., J. Zhang, et X. Zhu (2010). A review selection approach for accurate feature rating estimation. *Proceedings of COLING*.

Pang, B. et K. Lee (2005). Seeing stars : Exploiting class relationships for sentiment categorization with respect to rating scales. *Proceedings of ACL*.

Pontiki, M., D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, et S. Manandhar (2014). Semeval-2014 task 4 : Aspect bases sentiment analysis. *Proceedings of the 8th International Workshop on Semantic Evaluation at COLING 2014*.

Song, Y., S. Upadhyay, H. Peng, et D. Roth (2016). Cross-lingual dataless classification for many languages. *Proceedings of IJCAI*.

Sorg, P. et P. Cimiano (2012). Exploiting wikipedia for cross-lingual and multilingual information retrieval. *Data and Knowledge Engineering*.

Thet Tun, T., J.-C. Na, et C. Koo (2010). Aspect-based sentiment analysis of movie reviews on discussion boards. *Information Science*.

Wei, B. et C. Pal (2010). Cross lingual adaptation : An experiment on sentiment classifications. *Proceedings of ACL*.

Witten, I., E. Frank, M. Hall, et C. Pal (2016). *Data Mining : Practical Machine Learning Tools and Techniques, 4th edition*. Morgan Kaufmann.

Xiao, M. et Y. Guo (2013). Semi supervised representation learning for cross-lingual text classification. *Proceedings of the EMNLP*.

Zaid, E., T. Lehinevych, et A. Glybovets (2017). Cross-language text classification with convolution neural networks from scratch. *Computer Sciences and Mathematics*.

Summary

The paper deals with cross-language classification for customer relationship management within a commercial running system. The aim is to start from a source language where resources are avalaible, then to translate automatically and to learn within one target language. Various linguistic options and modern algorithms are presented and compared.