Prédiction automatique des classes TRUSTED, FAKE et PARODIC dans des articles de presse

Giovanna Favia
Agathe Helman
Nicolas Scarcella

Matière : Apprentissage automatique et outils traitement de corpus
Enseignant : M. Nouvel

Le projet suivant a consisté dans la prédiction de trois classes: “trusted”, “fake”, “parodic” concernant deux corpus de 60 articles de la presse. Le premier corpus est en français, le deuxième en italien.
Pour chacun, une division homogène a attribuée 20 articles dans chaque classe.
La réalisation de cette prédiction s’est effectuée à l’aide de l’outil WEKA, implémenté en Java et exploitable en Python au travers de la librairie python-weka-wrapper.

Nos deux corpus ont été respectivement divisés en deux parties: un premier sous-corpus d’entraînement, contenant 60% des articles, pour exercer l’algorithme et un second, de test, possédant 40% des articles pour effectuer la prédiction.

Précisons que nous travaillons par méthode supervisée. Par conséquent, l’attribution des classes pour chaque article a été réalisée manuellement, en fonction de la notoriété et des avis du site d’où ces articles provenaient. La problématique a donc été de comparer nos deux corpus selon les features utilisées pour la prédiction, ainsi que les résultats.

Ce travail s’est fait en deux étapes :
L’une, résumée dans notre première partie, concerne les divers traitements opérés sur les deux corpus : italien et français
L’autre, traitée en deuxième partie, synthétisera les tâches d’apprentissage automatique réalisées pour la prédiction et présentera, avec les résultats et leurs commentaires, un calcul de distances.

I) TRAITEMENT DES CORPUS

Constitution du corpus en 2 langues (italien et français)

A) Les corpus

A.1 Français

Le corpus francophone a été alimenté par six sites internet dont les articles sont tous issus, pour les classes parodic et trusted, de la presse africaine. Malheureusement, aucun site de fake news n’a été trouvé sur ce continent. Par conséquent, les deux sites de cette classe proviennent de la France métropolitaine.
Donc, 60 articles issus des journaux suivants (10 de chaque):

- Jeune Afrique (TRUSTED) : http://www.jeuneafrique.com/
- Africa News (TRUSTED) : http://fr.africanews.com/
- Lerpess (PARODIC) : https://www.lerpesse.com/
- State Afrique (PARODIC) : http://stateafrique.com/
- Les Moutons Rebelles (FAKE) : http://lesmoutonsrebelles.com/
- Dreuz (FAKE) : https://www.dreuz.info/

Les articles en français ont été sélectionné de manière homogène (3-4/10 articles par classe), selon trois thématiques différentes :

- Politique

- Sport
- Culture

Les articles suivants ont permis de choisir les articles des classes *fake* et *parodic* :
Parodic,
http://www.webdo.tn/2013/11/18/lerpesse-de-tunisie-ou-le-rendez-vous-avec-la-satire-lintox-et-le-rire/
http://www.rfi.fr/hebdo/20171201-afrique-gorafi-site-parodie-information-liberte-expression-state
Fake,
http://journalmetro.com/opinions/metroscope/1181419/voici-les-sites-internet-qui-publient-des-fausses-nouvelles/

Tableau récapitulatif de la répartition des articles en français selon la fréquence :

	Politique	Sport	Culture
Trusted	6	7	7
Parodic	7	7	6
Fake	7	6	7

A.1 Italien

Pour la collection du corpus italien, concernant la classe Trusted, nous nous sommes basés surtout sur des articles de catégories politique, société et sport, issus des principaux sites de presse nationale et internationale italienne :

La Repubblica (http://www.repubblica.it/)
Il Corriere della sera (http://www.corriere.it/)
Il Resto del Carlino (https://www.ilrestodelcarlino.it/)

Pour la classe fake il s'agissait surtout des catégories science, culture et politique, tirées de sites moins connus à niveau national et international :

Il Fatto Quotidiano (https://www.ilfattoquotidiano.it/)
Informare Existere (https://www.informarexresistere.fr)
Pianeta Blu News (http://www.pianetablunews.it/)
Essere Informati (https://www.essere-informati.it/)
Scenari Economici (https://scenarieconomici.it/)
Identità (https://identità.com/)

Tandis que pour la classe parodic, les catégories le plus fréquentes ont été société et politique, dont les articles ont été récupérés d'un seul site:

Lercio (http://www.lercio.it/)

Les deux corpus, construits dans un format XML ont été parsés à l'aide du module Python lxml. Toutefois, nous avons appliqué deux traitements différents pour le français et l'italien :

- Le corpus italien comporte deux traitement différents, visibles à travers deux balises: "treetagger" et "bigrams".
  La première contenant les lemmes des tokens de chaque article, combinés avec les parties du discours (POS), obtenus par l'emploi de l'outil Tree Tagger pour la langue italienne, et la seconde contenant la représentation des articles sous-forme de bigrams, produit à l'aide du module Python NLTK.
  De plus, afin d'effectuer une analyse sur les titres des articles, la function produisant le lemme et la partie du discours a été appliqué dessus, inclue dans une balise "treetagger_title". Ajoutons qu'une dédiacritisation et une minusculinisation ont été appliqué sur ce corpus, ainsi que le remplacement des guillemets françaises par des guillemets américaines et des guillemets doubles et des guillemets française par des guillemets simples sur les titres pour aider la détection des features.

- Une racinisation de chacun des tokens a été réalisée pour le corpus français à l'aide de SnowballStemmer du module NLTK.
  Remarque : la racinisation de certains mots a posé quelques problèmes, par exemple le mot « financier » à la racine « financi » ce qui empêche le jumelage avec le mot « finance », dont la racine est « financ » sans 'i'. L'usage d'une expression régulière a permis de pallier ce souci.
  Aussi, d'autres traitements pour la normalisation du corpus (séparation voire suppression de la ponctuation) ont été appliqué.

B) Les features

B.1: Français

La sélection de ces features a été faite à l'aide du logiciel Lexico3 qui nous a permis de déceler des particularismes dans chaque catégorie.

Voici un tableau répertoriant les features du corpus français qui nous ont semblé les plus pertinents et révélatrices des catégories :

Features	Trusted	Parodic	Fake
!	2	5	26 >>13
?	7	4	20 >> 10
«»	41	65	167 >> 80
Etranger	0	0	5
Ne (negation)	15	14	55 >> 42
Pas (négation)	28	18	100 >> 50

Nouvel Ordre Mondial	0	0	1
National	10	4	3
(T|r)rump	0	0	27 >> 14
américain	2	0	9
antisemit(e|es)	0	0	5
banqu(e|es|ier)	2	0	0
financ(ier|ier|e|es)	3	0	7

Étant donnée que le corpus fake est deux fois plus volumineux que les deux autres, les features sont divisés par deux pour cette classe afin de bien vérifier qu'elles sont symptomatique. Ainsi, le chiffre qui suit ">>" serait plus révélateur.
Les fréquences du fake inférieur à 10 n'ont pas été divisé.
Ce constat laisse prévoir du surapprentissage du côté du fake.
La couleur rouge indique lorsque la classe a une prédominance particulière.

Quelques remarques sur ces features:
La présence particulière de la ponctuation et de la négation dans nos corpus a constitué les premiers critères pour influencer nos algorithme.
L'expression "Nouvel Ordre Mondiale" n'apparaît qu'une foi mais elle est symptomatique, d'autant qu'on ne la relève que dans le corpus fake.
D'ailleurs, les fake news produisant des informations souvent rattachés au domaine du complotisme, il nous a semblé intéressant de relever des racines de mots tel que américain, financier ou banque. Par ailleurs, nous avons constatés que ce dernier était absent de nos classes *fake* et *parodic*. Également, le radical "antisemit", présent uniquement dans la classe fake nous a paru pertinent d'être noté.
À titre indicatif,précisons que l'utilisation des features nécessitent de se pencher plus profondément dans le corpus afin de vérifier si les occurrences témoignent bien du sujet qu'on leur laisse prévoir. Ainsi, "américain" ne fait pas forcément référence aux pourvoir de domination États-Unien. Simplement et puisqu'il apparaît plus de fois dans le corpus fake que dans les autres, il améliorera la prédiction.
Un personnage politique était particulièrement surreprésenté dans le corpus fake: 'Trump', c'est pourquoi nous l'avons retenu.

B.1: Italien

Pour la détection des features sur la langue italienne, on s'est servis aussi bien des lemmes et des POS, que des bigrams et du texte de départ.

La grande présence de certaines mots, expressions et punctuations dans une des trois classes, a sans aucun doute aidé à la mise en place des features, c'est le cas du point d'interrogation et du mot *gouvernement* pour la classe *Trusted*, du point d'exclamation et des guillemets pour le *parodic* ou du mot *italiens* pour le *fake*.

Toutefois, on a voulu se baser sur des caractéristiques plus générales, faisant référence et aux POS et aux POS combinées avec des lemmes, c'est le cas du nom commun *president*, suivi par un adjectif, dont on a eu quelques apperçu dans la catégorie *fake*, des adverbes et des noms propres de manière générale ou bien du numéral suivi du mot *milions*.

Le traitement des titres des articles y a vu son utilité dans la recherche des guillemets et des adjectifs, sachant que ces éléments aident à rendre une nouvelle plus attirante ou particulière et ont donc des chances d'appartenir à la classe *fake* ou bien *parodic*.

D'après l'exploration effectuée par le logiciel de textométrie lexico3, aucune des trois classes ne présentait une forte occurrence de bigrams répétés, c'est pourquoi l'utilité des bigrams a surtout été celle de rechercher des nom propres composé, comme "Etats Unis" ou bien des expressions devenus courantes dans le langage journalistique, comme "foreign fighters".

Lors de l'analyse des deux corpus sur le logiciel Lexico3, des différences ainsi que des corrélations ont été remarquées entre nos deux langues.

Features	Français	Italien
'!' et '"'	Fake	Parodic
'?'	Fake	Trusted
'Trump' et 'américain(s)'	Fake	Fake
italien		Fake
français	fake	

Notons, que la racine "franc", pour les mots tel que français et française n'a pas été conservé en tant que features car elle est surtout présente dans la classe *fake,* dont le contenu provient de journaux non africains.

II) APPRENTISSAGE AUTOMATIQUE: RÉSULTATS ET COMMENTAIRES

A) Algorithme de WEKA

Nous avons fait tourner sur nos corpus 4 algorithmes de WEKA:
- KNN ou K plus proche voisins, intitulé « IBk » dans WEKA.
- NaiveBayes ou Bayésien naïf, intitulé « NaiveBayesMultinomial » dans WEKA
- Perceptron ou réseau de neurones à une couche, intitulé « MultilayerPerceptron » dans WEKA

Notre but a été de pouvoir, sur un choix assez éclectique d'algorithmes, voir ceux qui apporteraient une bonne prédiction.

Voici le tableau récapitulatif des résultats obtenus:
Remarquons que nous nous sommes basé que sur la prédiction car l'on cherchait seulement à savoir si la classe était correctement prédite.

Précision (italien)	Trusted	Fake	Parodic
KNN	1.0	1.0	1.0
Bayes	0.625	0.625	0.75
Perceptron	0.75	1.0	1.0

Précision (français)	Trusted	Fake	Parodic
KNN	1.0	1.0	1.0
Bayes	0.5	0.25	0.75
Perceptron	0.875	0.875	0.875

Il ressort des tableaux ci-dessus, que l'algorithme des k plus proches voisins obtient les meilleurs scores. Un sans faute et cela, dans les deux corpus.
Nous précisons qu'au départ, nos deux corpus n'étaient constitués que d'une trentaine d'articles chacun. Là aussi, KNN avait la première place. Nous avons doublé les corpus avec de nouveaux articles et il s'avéra que KNN prédisait toujours très bien.
À la différence de KNN qui regarde si tel article appartient à telle classe, le classifieur bayésien a une approche probabiliste. Ainsi, il regarde la probabilité que tel article soit fake, trusted ou parodic, sachant la présence des features.
Notre conclusion est que KNN a bien fonctionné car le corpus reste petit et que le repérage des features crée un éparpillement qui ne gêne pas l'algo pour rapprocher un articles des autres. Naive Bayes lui, construit comme des groupes de classes, le fait que les critères soient éparpillés ne lui profite pas. Le surapprentissage de la classe *fake* se démarque bien par cette algorithme. La classe parodic avait le plus petit corpus, elle a donc le score le plus bas avec 0.25.
Le perceptron, quand à lui, obtient des scores satisfaisants, toutefois, nous ne savons pas comment weka ajuste les poids de ce réseau de neurone à une couche.

Enfin, un calcul des similarités à été appliqué.

B) Calcul des distances entre les différentes classes du corpus

Nous avons décidé de calculer la distance entre chaque classe du corpus afin de voir si elles diffèrent peu ou beaucoup, ce qui peut aider à comprendre certains résultats obtenus avec Weka.

B.2.1 - Traitement des corpus

Afin de calculer les distances, les 2 corpus ont été lemmatisés à l'aide de TreeTagger

B.2.2 - Les calculs utilisés

Nous avons décidé de calculer les distances à l'aide de 3 calculs :

- L'indice de Jaccard

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

- L'indice de Sørensen-Dice

$$s = \frac{2|X \cap Y|}{|X| + |Y|}$$

- La similarité cosinus

$$\cos\theta = \frac{A \cdot B}{\|A\| \cdot \|B\|}$$

B.2.3 - Les résultats obtenus

Voici le tableau des résultats obtenus sur le corpus français :

classes	JACCARD	DICE	COSINUS
trusted – parodic :	0.9285979918185199	0.9629772464327034	0.9857949102947855
trusted - fake :	0.5125535420098847	0.6777327582451096	0.6886289388059863
fake - parodic :	1.1675502540313674	1.0772993630573249	1.1630279922284275

Voici le tableau des résultats obtenus sur le corpus italien :

classes	JACCARD	DICE	COSINUS
trusted - parodic :	0.7868715550359111	0.880725369227893	0.8877484563094874
trusted - fake :	0.7318147304479878	0.8451420554191512	0.8464636002289485
fake - parodic :	0.7360020705719955	0.8479276413875361	0.8500230243846698

On constate globalement que les classes sont très proches, ce qui indique que le vocabulaire utilisé dans chaque classe est généralement le même.
Cependant, sur le corpus Français, le vocabulaire des classes trusted et fake diffère un peu plus entre ces deux classes, en effet, la distance calculée est comprise entre 0,5 et 0,7 , ce qui reste tout de même relativement proche.

CONCLUSION

Pour la phase d'apprentissage sur corpus italien, l'utilisation de bigrams ne semble finalement pas utile car, il est aussi bien possible de chercher directement des paires de mots dans le texte brute. Du côté des algorithmes et suite aux résultats donnés par le calcul des k plus proches voisins, nous avons choisi de tester un quatrième algorithme (KStar), de la même catégorie, à savoir "lazy". Les résultats furent aussi excellent que le KNN.
Il serait intéressant de tester à nouveaux ces algorithmes avec un corpus ayant des catégories bien spécifique de sorte que les features ne soient pas trop éparpillées.