"Random Forest is all you need" : Classification de discours parlementaires par parti politique

Amina Bouteldja
Université Paris Nanterre

Juliette Massy
Université Sorbonne Nouvelle

Yaasmine Nohur
Université Paris Nanterre

Abstract

L'analyse d'opinion à partir de documents textuels possède un large panel d'applications, de l'évaluation de retours clients au suivi d'une image publique ou médiatique. Pour son édition 2009, l'organisation du Défi Fouille de Texte propose de confronter différentes méthodes sur la reconnaissance automatique du parti politique d'un orateur à partir d'interventions parlementaires multilingues. Nous détaillons ainsi plusieurs modèles fine tunés basés sur trois types de classifieurs dont des modèles parallèles Random Forest qui atteignent une performance moyenne de 78%. Nous présentons ainsi une amélioriation significative par rapport aux résultats obtenus par les équipes participantes.

1 Introduction

L'appartenance à un parti politique peut transparaître ou influencer le jugement exprimé par quelqu'un sur un sujet. Dans le cadre d'interventions politiques officielles, cela s'illustre d'autant plus dans le discours par mesure de distinction avec d'autres partis. De ce constat, l'identification du parti politique d'un intervenant parlementaire à partir de la retranscription de son intervention parait être une tâche intéressante. Proposée lors la cinquième édition du DEFT en 2009, cette tâche s'est révélé ardue pour les participants au défi avec des modèles dont les performances avoisinent les 30%.

Depuis les dernières décennies, l'augmentation du volume de données disponibles et la complexification des tâches justifie l'omniprésence des méthodes computationnelles pour les tâches de classifications textuelles (Manning et al., 2008). Des méthodes comme des classifieurs d'apprentissage automatique réalisent ainsi de bonnes performances sur des tâches telles la détection du caractère objectif/subjectif global d'un texte (Bestgen and Lories, 2009). En ce qui concerne la classification de textes par mouvement politique, la tâche s'avère complexe et ce d'autant plus lors du traitement de données multilingues. Étant donné que les modèles sont généralement entraînés sur une seule langue, il est courant de construire des modèles parallèles ou de traduire les données et de les traiter comme des corpus monolingues (Courtney et al., 2020). Aujourd'hui, le développement rapide des méthodes neuronales permet des gains de performances conséquents sur des tâches de classifications comme celle ci, et ce même avec des données multilingues. Ces modèles s'illustrent notamment dans les cas avec des frontières de décisions floues qui peuvent poser problèmes aux humains et aux modèles d'apprentissages artificiels (Nicholls and Culpepper, 2022).

Pour cette tâche, nous nous plaçons dans la continuité des travaux réalisés lors du DEFT'09 et utilisons ainsi des modèles de classifieurs d'apprentissage artificiel. Nous fournissons donc trois types de classifieurs (SVM, Random Forest et Naive Bayes) avec plusieurs modèles entraînés. Dans un premier temps, nous présentons le corpus utilisé et les opérations complètes de pré-traitement et de vectorisation effectuées. Nous proposons ensuite des modèles selon deux approches : tout d'abord des modèles entraînés sur des corpus multilingues puis des modèles parallèles, entraînés par langues. Enfin, nous réalisons une analyse comparative de nos résultats face à ceux présentés dans les actes du DEFT puis proposons des pistes de réflexions et d'améliorations sur le travail effectué.

2 Traitement des données

2.1 Données et pré-traitements

La tâche consiste donc à identifier le parti politique d'appartenance d'un intervenant parlementaire à partir de son intervention. Elle se conçoit donc comme une tâche de classification textuelle multi-classes.

Le corpus associé à cette tâche est la retranscription multilingue des débats de l'ordre du jour pour les 313 séances parlementaires qui se sont tenues entre

1999 et 2004 au Parlement européen. C'est un corpus multilingue avec trois des langues officielles de l'Union européenne représentées (français, italien et anglais). Chaque intervention est associée au parti politique de l'orateur. Les interventions choisies sont celles des cinq partis les plus représentés : le parti Européen des Libéraux, Démocrates et Réformateurs (ELDR), le Parti Populaire Européen et Démocrates Européens (PPE-DE), le Parti Socialiste Européen (PSE), les Verts, Alliance Libre Européenne (Verts/ALE) et le groupe confédéral de la Gauche Unitaire Européenne et Gauche Verte Nordique (GUE/NGL).
Le corpus mis à notre disposition a été préalablement anonymisé et nettoyé. Il a été divisé en corpus d'apprentissage (60%) et corpus de test (40%). En tout, 19370 interventions par langues composent le corpus d'apprentissage pour 12917 interventions par langues pour le corpus de test. Pour chaque langue, les corpus comprennent les mêmes interventions présentées aléatoirement.
Le corpus est nettoyé des expressions numériques et de la ponctuation. Il est ensuite mis en minuscules et débarrassé des mots vides. Le vocabulaire de mots vides utilisé pour chaque langues provient de Spacy et est composé de plusieurs centaines de mots.

2.2 Vectorisation

La vectorisation des documents est réalisée au moyen d'une représentation fondée sur le TF*IDF. Cette méthode particulièrement efficace pour des opérations de classification textuelle (Chen et al., 2016), présente l'avantage de faire ressortir les mots pertinents en prenant en compte non seulement leur fréquence mais aussi leur répartition dans les documents. Les mots conservés sont ceux qui apparaissent au minimum dans 10 documents sans maximum d'apparition. Finalement, les matrices obtenues sont de très hautes dimensions avec un grand nombre de zéros (*sparse matrix*). Deux types de vectorisation sont testés : une vectorisation du corpus multilingue pour les modèles multilingues et une vectorisation par corpus de langues pour les modèles parallèles.

3 Modèles

3.1 Baseline : Modèles multilingues

La première piste explorée est celle de l'entrainement de classifieurs sur un seul corpus d'apprentissage multilingue (anglais, français et italien).

3.1.1 Naive Bayes (NB)

Multinomial Naïve Bayes (MNB) est un classifieur linéaire Bayésien combiné à une loi multinomiale. Ce modèle est entrainé avec nos données et son exactitude est de 0.48, il classifie mal plus de la moitié des documents du test. La classe GUE-NGL obtient les meilleurs résultats en précision avec un score de 0.87 mais a un rappel très bas de 0.26. La classe PPE-DE est la classe possédant la meilleure F-mesure (0.60) avec un score de précision de 0.44 et un rappel de 0.94. Les résultats sont donc assez faibles dans l'ensemble et disparates selon les classes.

3.1.2 Support Vector Machine (SVM)

SVM linéaire est un type de classifieur fréquemment utilisé dans les tâches de classifications car il présente plusieurs avantages. Réputé pour gérer les données complexes et multidimensionnelles, un classifieur SVM se prête à cette tâche pour laquelle les données représentent une matrice creuse. Il permet donc d'éviter le problème de dimension des données et facilite l'interprétation des résultats par rapport aux modèles non linéaires (Lauer and Bloch, 2006). Entraîné sur le corpus multilingue avec les paramètres de base de *Scikit-Learn*, il obtient une performance satisfaisante avec une précision de 0.73. Le modèle a une bonne capacité à identifier correctement les instances de la GUE-NGL qui possède la meilleure précision (0.82), mais aussi la meilleure F-mesure (0.80). La classe PPE-DE, elle, a le meilleur rappel avec un score de 0.81 indiquant que le modèle reconnaît un grand pourcentage des instances appartenant réellement à cette classe.

3.1.3 Random Forest (RF)

De par son efficacité à traiter des données de hautes dimensions (Xu et al., 2012) et les problèmes a plus de deux classes (Breiman, 2001), Random Forest est une autre méthode de classification adaptée à la tâche effectuée. Comme son nom l'indique, cette méthode repose sur une succession d'arbres de décisions. Comme précédemment, le premier essai avec le corpus multilingue complet s'avère concluant. Avec une F-mesure de 0.77, le modèle ainsi entrainé performe relativement bien même si la profondeur d'arbres et leur nombre n'est pas limité ce qui peut rapidement allonger le temps d'éxécution du modèle.

Les modèles Random Forest et SVM réalisent donc des performances satisfaisantes sur le corpus multilingue mais il est intéressant d'entraîner des modèles différents pour chaque langue, réduisant ainsi légèrement la dimension des données et le coût d'entraînement.

3.2 Modèles par langues

La seconde seconde piste, plus prometteuse, est celle de l'entraînement de modèles distincts parallèles pour chaques langues.

3.2.1 Naive Bayes (NB)

Un premier entraînement des modèles Naive Bayes sur chaque corpus de langue est réalisé avec les paramètres par défauts de *Scikit-Learn*. Le classifieur performe faiblement avec une F-mesure de 0.48 en moyenne soit le même score que pour le modèle NB précédent.
Pour optimiser les résultats et fine tuner les modèles, une étape de test de valeurs des paramètres du modèle est nécéssaire. Grâce à *GridSearchCV* de *scikit-learn*, une recherche exhaustive des paramètres permettant une optimisation de la F-mesure est réalisée. La méthode est peu coûteuse pour ce classifieur et facilite l'exploration des paramètres.
Les meilleurs paramètres trouvés par cette méthode sont les mêmes pour les trois modèles et entraînent des résultats différents mais proches. Les modèles italien et français obtiennent des résultats très similaires avec une exactitude de 0.63 tandis que le modèle de l'anglais obtient une exactitude de 0.59. Les classes GUE-NGL et PPE-DE sont les mieux détectées par les modèles et ce pour l'ensemble des langues.

3.2.2 Support Vector Machine (SVM)

Le premier entraînement des modèles SVM est effectué en utilisant les paramètres par défaut de *Scikit-Learn*. Les modèles italien et français ont des résultats très similaires avec une exactitude d'environ 0.74 tandis que le modèle anglais obtient une exactitude de 0.72. Sans fine tuning, les résultats obtenus sont similaires au modèle SVM multilingue. Les paramètres du modèle sont ensuite ajustés pour optimiser le modèle de base. Avec *RandomizedSearchCV* de *Scikit-Learn*, un ensemble de valeurs possibles est établie pour chaque paramètres et des combinaisons sont testées de manière aléatoire. À la fin, la meilleure combinaison de paramètres pour l'optimisation de la F-mesure sont sélectionnés avec le paramètre *scoring*. Cependant, en exécutant le modèle avec les meilleurs paramètres obtenus, les résultats sont inférieurs à ceux précédemment obtenus, la performance passant de 72% à 69%.

3.2.3 Random Forest (RF)

En testant un modèle différent par corpus de langues avec les paramètres de base de *Scikit-Learn*, la précision obtenue pour chaque modèle est similaire à celle du modèle RF multilingue avec une F-mesure de 0.78. Ces modèles sont entre autre caractérisés par l'absence de restriction en ce qui concerne le nombre d'arbres ou la profondeur des arbres.
Le fine tuning du modèle est réalisé comme pour les modèles SVM à l'aide de *RandomizedSearchCV*. Le test des paramètres a conduit à une augmentation de la précision des modèles de 1% notamment grâce à l'augmentation du nombre d'arbres des modèles (de 100 à 800). D'autres essais de paramètres ne permettent pas d'amélioration significatives de la performance (± 0,001 de variation de précision pour les modèles).
Les 3 modèles ont des performances similaires avec un F-mesure égale à 79% pour l'anglais et l'italien et 78% pour le français. Les interventions GUE-NGL sont les mieux classées pour chaque modèles monolingues RF même si elle ne sont pas les plus nombreuses. De nombreux documents sont faussement associés à la classe la plus représentée PPE-DE notamment ceux de la classe PSE, la seconde classe la plus représentée.

4 Résultats

Pour la baseline ainsi que le modèle par langues, les modèles Random Forest s'avèrent être les modèle les plus performants des trois, suivi des SVM linéaires, puis des Naive Bayes Mutlinomial (voir Table 1).
Il n'y a pas de différence dans le classement des classifieurs entre la baseline (modèle multilingue) et les modèles par langues. Bien que les classifieurs SVM linéaires et Random Forest n'aient montré qu'une amélioration d'environ 0.01 (sur l'italien et le français pour SVM et sur l'italien et l'anglais pour RF) en termes d'exactitude, ils demeurent les classifieurs présentant les meilleurs résultats, tant sur le corpus multilingue que sur les corpus individuels par langues. Par ailleurs, il

est important de souligner que malgré ces moindres résultats, le classifieur Naive Bayes a montré la meilleure amélioration grâce à une recherche exhaustive (GridSearch), faisant de lui le modèle ayant bénéficié de la meilleure optimisation des paramètres (voir Table 1).

5 Conclusion

Ce travail présente donc deux types de modèles (multilingues et monolingues) pour une tâche d'identification du parti politique d'un intervenant parlementaire à partir de son intervention. Les résultats des modèles de baseline multilingues Random Forest et SVM se révèlent étonamment hauts avec des performances environnant les 75%. Les résultats des modèles monolingues se révelent eux étonamment proches des modèles multilingues et ce même après le fine-tuning. Seuls les modèles multilingues Naive Bayes semblent par ailleurs bénéficier d'une augmentation significative de leur performance après optimisation des paramètres (+ 0.12 de précision). Globalement, les modèles Random Forest sont les plus perfomants sur cette tâche et surpassent largement les résultats obtenus par annotation humaine et par les équipes du DEFT'09. La tâche demandée est en effet une tâche complexe dont l'annotation humaine pour l'identification du parti politique de l'orateur est difficile comme en témoigne des mesures de rappel et de précision compris entre 0.23 et 0.47. Par ailleurs, le fichier résultat soumis par Forest et al. (2009) présente des modèles dont les performances sont jugées insuffisantes (F-mesure d'environ 0.33).

Pour optimiser les performances des modèles plusieurs pistes d'amélioration peuvent être envisagées. Une opération de prétraitement supplémentaire comme une lemmatisation ou stemming sur les données est une piste pouvant conduire à l'amélioration de la performance des modèles et à la réduction des dimensions des données. De plus, un fine-tunning plus exhaustif des paramètres des classifieurs serait intéressant pour obtenir des résultats plus concluants. En effet, la puissance de nos machines limite notamment les opérations possibles et donc la qualité des résultats obtenus. Nous n'avons ainsi pas eu l'occasion d'optimiser au maximum les modèles SVM, l'exécution du code d'optimisation des paramètres ayant pris plus de huit heures lors de la première tentative.

Finalement, au vu des capacités actuelles des modèles neuronaux et du théorème d'approximation universelle, il serait intéressant d'explorer l'utilisation d'un modèle neuronal pour effectuer cette tâche de classification.

References

Yves Bestgen and Guy Lories. 2009. Un niveau de base pour la tâche 1 (corpus français et anglais) de deft'09. *Actes DEFT'09 « DEfi Fouille de Textes »*.

Leo Breiman. 2001. Random forest. *Machine Learning*, 45:5–32.

Jindong Chen, Pengjia Yuan, Xiaoji Zhou1, and Xijin Tang. 2016. Performance comparison of tf*idf, lda and paragraph vector for document classification. *KSS 2016: Knowledge and Systems Sciences*, pages 225–235.

Michael Courtney, Michael Breen, Iain McMenamin, and Gemma McNulty. 2020. Automatic translation, context, and supervised learning in comparative politics. *Journal of information technology politics*.

Dominic Forest, Astrid van Hoeydonck, Danny Létourneau, and Martin Bélange. 2009. Impacts de la variation du nombre de traits discriminants sur la catégorisation des documents. *Actes DEFT'09 « DEfi Fouille de Textes »*.

Fabien Lauer and Gérard Bloch. 2006. Méthodes svm pour l'identification. Hal-00110344.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Tom Nicholls and Pepper D. Culpepper. 2022. Better political text classification using large language models. *Information, Redistribution and Financial Regulation conference*.

Baoxun Xu, Xiufeng Guo, Yunming Ye, and Jiefeng Cheng. 2012. An improved random forest classifier for text categorization. *Journal of Computers*, 12:2913–2920.

Modèle	**Type de valeurs**	**ELDR**	**GUE-NGL**	**PPE-DE**	**PSE**	**Verts/ALE**	**Accuracy**
Multilingue MNB	Précision	1	0.87	0.44	0.55	0.83	0.48
	Rappel	0.0	0.26	0.94	0.38	0.03	
Multilingue SVM	Précision	0.78	0.82	0.72	0.69	0.76	0.73
	Rappel	0.61	0.77	0.81	0.71	0.63	
Multilingue RF	Précision	0.99	0.95	0.66	0.81	1	**0.78**
	Rappel	0.63	0.74	0.94	0.72	0.62	
MNB anglais	Précision	0.56	0.65	0.68	0.55	0.49	0.59
	Rappel	0.59	0.64	0.58	0.59	0.59	
SVM anglais	Précision	0.79	0.81	0.71	0.68	0.74	0.72
	Rappel	0.60	0.76	0.80	0.70	0.62	
RF anglais	Précision	0.99	0.96	0.66	0.87	1	**0.79**
	Rappel	0.65	0.76	0.96	0.71	0.62	
MNB français	Précision	0.60	0.71	0.69	0.58	0.55	0.63
	Rappel	0.61	0.64	0.65	0.63	0.60	
SVM français	Précision	0.81	0.82	0.72	0.70	0.77	0.74
	Rappel	0.61	0.79	0.81	0.0.72	0.64	
RF français	Précision	0.99	0.94	0.66	0.86	1	**0.78**
	Rappel	0.63	0.75	0.96	0.70	0.62	
MNB italien	Précision	0.65	0.71	0.68	0.58	0.57	0.63
	Rappel	0.59	0.63	0.65	0.65	0.60	
SVM italien	Précision	0.79	0.83	0.73	0.71	0.77	0.75
	Rappel	0.62	0.77	0.82	0.73	0.65	
RF italien	Précision	0.99	0.94	0.66	0.87	1	**0.79**
	Rappel	0.63	0.76	0.96	0.71	0.62	

Table 1: Tableau comparatif des performances des modèles multilingues et monolingues.