Le changement linguistique au XVIIe s. : nouvelles approches scriptométriques

Simon Gabay, Rachel Bawden, Philippe Gambette, Jonathan Poinhos, Eleni Kogkitsidou, Benoît Sagot

Le changement linguistique au XVII^e s. : nouvelles approches scriptométriques

Simon Gabay[1], *Rachel* Bawden[2], *Philippe* Gambette[3], *Jonathan* Poinhos[3], *Eleni* Kogkitsidou[3] et *Benoît* Sagot[2]

[1]Université de Genève (Suisse)
[2]Inria (France)
[3]LIGM, Université Gustave Eiffel, CNRS, Marne-la-Vallée (France)

Résumé. La fin du XVII^e s. reste un angle mort de la recherche sur les systèmes graphiques, en dépit de l'importance de cette période pour l'histoire du français, qui s'est alors vu imposer une orthographe continuant encore aujourd'hui de régir, peu ou prou, son écriture. En privilégiant une approche pratique, sur corpus, plutôt que théorique, nous nous proposons de poser les bases d'une étude scriptométrique computationnelle de la langue classique, s'intéressant à son évolution au cours du Grand Siècle. Pour ce faire nous mesurons et évaluons la distance entre la langue classique et la langue contemporaine au moyen de deux méthodes de normalisation automatique, une avec des règles, et une autre avec un réseau de neurones.

Abstract. Linguistic change in 17th c. France: new scriptometric approaches The end of the 17th c. remains a blind spot of the research on the spelling system, despite its importance for French at this period, during which a strict norm, still (more or less) in place, was created and imposed. Focusing on a practical rather than a theoretical approach, we propose to lay the foundation for a computational scriptometric study of early modern French and analyse the evolution of the spelling system over the 17th c. To do so, we measure and evaluate the distance between the early modern and the contemporary versions of the language, thanks to two automatic normalisers: one rule-based and another one neural-based.

Les aspects grapho-phonétiques du français classique restent paradoxalement très mal connus, en dépit de l'importance de la question graphématique au XVII^e s. La richesse des débats théoriques sur l'orthographe à cette époque a jusqu'à présent concentré l'essentiel de la recherche (par ex. dans Catach 2001 ou Biedermann-Pasques 1992), et les cahiers de Mezeray (Académie française 1863) ou les *Remarques* de Vaugelas (Vaugelas 1647, 2009) restent encore parmi les principales sources utilisées, plutôt que des relevés statistiques sur de vastes corpus.

À la décharge des linguistes, les sources utilisables sont excessivement rares. L'absence d'éditions philologiquement rigoureuses comme celle de Charles Sorel par A. Spica (Sorel 2014) ou de

textes privés par G. Ernst (2019) a évidemment joué un rôle crucial, en privant non seulement les lecteurs du plaisir de lire des textes dans une langue proche des documents d'époque, mais aussi les diachroniciens de sources primaires pour l'étude des systèmes graphiques à l'époque moderne. L'étude de ces derniers est cependant devenu un enjeu majeur de la recherche, pour des raisons évidemment linguistiques, mais aussi littéraires, et même informatiques.

Si les divers dialectes et autres *scriptae* peuplant l'ancien et le moyen français ont été abondamment décrits (par ex. dans Dees 1985), tout comme l'« orthographie » de la Renaissance (pour reprendre le terme de Baddeley 1993), la lente imposition d'une norme orthographique, pourtant un phénomène majeur dans l'histoire d'une langue aussi prescriptive que le français, reste un angle mort de la linguistique diachronique. Comment le français que nous connaissons, peu ou prou, aujourd'hui a-t-il supplanté ses diverses réalisations modernes ? Doit-on attribuer un rôle important à des *Fachsprachen* (« langues spécialisées »), les pratiques linguistiques de certains groupes professionnels, comme les imprimeurs, ayant pu être une des sources de la nouvelle norme écrite ? Des imprimés francophones provenant des quatre coins de l'Europe (Pays-Bas, Suisse, Allemagne, Angleterre. . .), des interférences avec d'autres variations diatopiques expliquent-elles certains choix ?[i]

Ce problème de la variation graphique n'est cependant plus l'apanage des seuls linguistes, et rencontre les préoccupations d'autres chercheurs, comme les spécialistes d'informatique. Nombre de bibliothèques possédant des fonds patrimoniaux cherchent désormais à offrir des numérisations augmentées d'informations facilitant la lecture (normalisation linguistique) ou l'exploration des données (recherche de mots en dépit de la variation graphique). Les outils nécessaires à la réalisation de tels services, conçus par les spécialistes de traitement automatique des langues (TAL), doivent néanmoins s'appuyer sur une connaissance minimale de la langue afin d'entraîner, évaluer et améliorer des outils complexes à même de gérer une langue graphiquement instable (Gabay, Bartz et Deguin 2020).

Le pari que nous faisons est, qu'à rebours, ces approches computationnelles peuvent offrir de nouvelles perspectives à la linguistique diachronique. Profitant d'un travail sur la normalisation automatique du français classique, *i.e.* son alignement sur le système graphique du français contemporain *modulo* quelques exceptions pour des raisons métriques[ii], nous nous proposons d'évaluer plusieurs méthodes pour mesurer la distance graphématique entre la langue classique et la langue contemporaine, et d'analyser cette distance pour comprendre l'évolution du français.

1 L'analyse scriptométrique de la langue

Avec l'apparition de l'informatique, l'étude du changement linguistique peut désormais proposer des observations sur les lieux comme le rythme du changement de manière plus concrète qu'avec l'étude des traités de grammaire, et plus précise qu'avec les traditionnels relevés artisanaux. De nombreux travaux abordent plus ou moins directement ce problème, qu'ils soient méthodologiques (en réfléchissant à une méthode optimale afin de calculer la distance entre deux états de langue) ou pratiques (en appliquant ces méthodes de calculs à différents types de données).

1.1 Calculer la « distance » entre deux mots

Les études les plus anciennes datent du début des années soixante-dix avec les travaux précurseurs de Jean Séguy, l'inventeur décédé trop jeune du terme « dialectométrie » (Séguy 1973) et pionnier des calculs de distance sur des grands ensembles de données (Séguy 1971), puis d'Hans Goebl (notamment sa thèse, cf. Goebl 1982). Du fait de l'utilisation d'atlas linguistiques plutôt que de corpus textuels, mais aussi de la nature diatopique des dialectes, l'essentiel des résultats est publié sous la forme de cartes (Goebl 2011) produites à partir de calculs de distance entre une forme A et une forme B (par ex. [drum'ei] vs [drum'i] pour « Dormir » en franco-provençal)[iii].

Le calcul de cette distance a été sujet à plusieurs évolutions, une des principales améliorations étant due à Br. Kessler (1995) qui, le premier, abandonne la distance de Hamming (1950) au profit de celle de Levenshtein (1966) pour évaluer quantitativement la similarité entre deux points d'enquête (cf. tab. 1) – les deux distances calculant le nombre minimal de caractères à modifier pour passer d'une chaîne de caractères à une autre, mais la première n'autorisant que les substitutions (*etoit→étoit*), la seconde ajoutant aux substitutions les insertions (*enfans→enfants*) et les suppressions (*debvoir→devoir*).

Tableau 1. Calcul de la distance de Hamming puis de la distance de Levenshtein entre *estoit* et *étoit*.

	e	s	t	o	i	t	
Distance de Hamming							
	é	t	o	i	t		
d(estoit, étoit)	1 +	1 +	1 +	1 +	1 +	1 =	6
Distance de Levenshtein							
	é		t	o	i	t	
Lev(estoit, étoit)	1 +	1 +	0 +	0 +	0 +	0 =	2

De nombreux raffinements ont été amenés par la suite à cette distance de Levenshtein, comme l'utilisation de sa version de Damerau-Levenshtein (Damerau 1964), plus à même de gérer le cas des transpositions trouvées par exemple dans les cas de métathèses (*formage* pour *fromage*) où la permutation des deux lettres ne compte que pour un changement (*or→ro*) et non deux (*o→r+ r→o*). Il serait cependant vain et fastidieux de résumer ici les recherches sur ces calculs par les écoles de Salzbourg (Goebl 2012) et de Groningue (Nerbonne et Heeringa 2010), qui ont par ailleurs largement synthétisé leurs travaux pour des publics plus ou moins versés dans les mathématiques.

1.2 Une approche sur corpus textuel

Ces études ne se sont que très peu penchées sur le changement linguistique du fait de leur focalisation sur les atlas, sauf dans les rares cas où les données le permettaient (Goebl 2006). Depuis désormais une décennie, des approches sur corpus sont néanmoins apparues, avec les thèses de J. Grieve, qui propose une étude synchronique de l'anglais états-unien (Grieve 2009), et celle de Cl. Vachon, qui s'intéresse au français de la Renaissance (Vachon 2010). Sur la base d'un corpus composé avec un extrême soin, cette dernière a clairement dégagé, au niveau graphématique, un changement progressif à partir de 1550 culminant à la fin du siècle, où le système du français moderne se dessine clairement, puis un reflux au début du XVII^{e} s., probablement lié à celui de la Réforme (Vachon 2010, p. 253).

Un des principaux enjeux techniques pour la réalisation de telles études repose sur la constitution de documentations de grande ampleur, afin d'asseoir clairement les résultats. Malheureusement, de tels corpus de français classique n'existent pas pour deux raisons. D'une part, comme en a amèrement fait l'expérience Cl. Vachon (2010, p. 32, n. 31), les éditeurs de textes ont pris l'habitude de normaliser la langue de cette époque (Duval 2015 ; Gabay 2014), ce qui rend particulièrement compliqué, voire impossible son étude. D'autre part, les quelques corpus qui ont été constitués, comme celui de Cl. Vachon, mais aussi d'autres comme celui du Réseau Corpus Français Préclassique et Classique (RCFC) (Amatuzzi *et al.* 2020) ne sont pas, ou pas intégralement, disponibles pour les chercheurs. Étant donné les quantités de données sans cesse grandissantes nécessaires aux études computationnelles, il est cependant évident que ces deux corpus, même librement accessibles, resteraient de toutes les manières insuffisants pour les approches les plus récentes proposées en TAL.

L'apparition de ces corpus pose, à rebours, de nouvelles questions sur nos capacités à extraire de l'information graphématique sur la base des méthodes préalablement exposées. En effet, l'utilisation de points d'enquête dans des atlas simplifie considérablement le travail de comparaison, qui se fait d'emblée mot à mot (prononciation d'une localité A vs prononciation d'une

localité B). Le passage à des données en contexte, donc non préparées comme dans des romans ou des chartes, fait immédiatement perdre le point de comparaison que peut être une attestation dans une autre localité, mais aussi la transcription phonétique comme pivot pour la comparaison – raison pour laquelle nous ne parlons pas ici de « dialectométrie » mais de « scriptométrie », l'objectif n'étant pas d'étudier des dialectes mais le polymorphisme scripturaire et son évolution. Confrontée à ce problème, Cl. Vachon propose une méthode semi-automatique d'extraction des particularités graphématiques, fondée sur une comparaison des formes originales et normalisées (*seureté*<*securitas* vs *sureté*) dont la liste a été établie à la main. Bien que rigoureuse, cette méthode se révèle extrêmement chronophage, et reste donc inapplicable pour des recherches sur des corpus plus vastes.

2 Normaliser la langue

Depuis plusieurs années, la recherche s'est intéressée au détournement de la traduction automatique à des fins de normalisation des états de langue anciens (Bollmann 2018) — les deux tâches étant en effet équivalentes, la seule différence étant que la cible n'est plus une autre langue, mais un autre état de langue, plus récent. Aussi simple que paraisse la normalisation, ce recours à des outils complexes s'explique par les difficultés rencontrées dans certains cas (par ex. *quoique* → *quoi que* vs. *quoique*). Des premières études sur le français classique (Gabay et Barrault 2020 ; Gabay, Riguet et Barrault 2019) ayant démontré la pertinence de ce choix, il nous a paru qu'elle serait à même de répondre aux besoins des lecteurs en quête de textes facilement lisibles, mais aussi des linguistes qui cherchent à établir ce point de comparaison nécessaire à l'analyse scriptométrique : il suffirait de comparer une version normalisée automatiquement et sa version originale pour analyser la forme de cette dernière.

2.1 Jeux de données : les corpus FreEM

Une telle approche nécessite *a minima* un corpus destiné à construire des outils de normalisation qui associe des exemples en français classique à leur équivalent normalisé. C'est donc un corpus « bilingue » parallèle. Notre corpus parallèle FreEM$_{\text{norm}}$(pour *FREnch Early Modern*), régulièrement augmenté depuis les premiers essais mentionnés *supra*, est un corpus constitué de phrases ou de segments de phrases[iv] d'environ 658 000 tokens. Il propose à chaque fois une version diplomatique et une version normalisée du texte, la normalisation étant respectueuse du mètre comme expliqué *supra* (cf. exp. 1). Construit sur la base des textes disponibles, ce corpus parallèle est imparfaitement distribué, même si une attention a été portée au fait que l'ensemble du siècle, des formes (vers vs. prose) et des genres littéraires soient représentés (cf. tab. 2).

Phrase d'origine : Achevez , Seigneur , voſtre ambaſſade
Phrase normalisée : Achevez , Seigneur , votre ambassade (1)

Un second corpus, FreEM$_{\text{max}}$, vient en complément et a pour objectif de rassembler le maximum de textes disponibles écrits entre 1500 (retenu comme date de fin du moyen français) et 1800. Contrairement à FreEM$_{\text{norm}}$, ce corpus est « monolingue » (chaque texte n'y est disponible que dans une seule version, normalisée ou non). Pour chaque texte, les métadonnées (notamment l'auteur, le titre, la date, le genre, éventuellement le sous-genre, le type de transcription) sont harmonisées afin d'en faciliter l'exploitation. Les sources sont extrêmement diverses : ont été notamment mobilisés wikisource, des collègues et des corpus en ligne. Certaines sources n'étant pas distribuées avec des licences permettant leur redistribution, il est malheureusement impossible de publier FreEM$_{\text{max}}$dans son intégralité, à la différence de FreEM$_{\text{norm}}$.

Tableau 2. Distribution du corpus FREEM$_{norm}$ par décennie.

Décennie	Textes	Tokens	Auteurs
1600	3	42 032	Sponde, Ellain, D'Urfé
1610	1	3 417	Coutme de Normandie
1620	3	113 698	Gournay, Viau, Guez de B.
1630	3	32 771	Gomberville, Tr. L'Hermite
1640	4	38 471	Sales, Mendez Pinto, Pascal, Descartes
1650	3	35 596	Scudéry, Chapelain
1660	8	138 106	Molière, Pascal, La Fontaine, Racine, Bussy-R.
1670	8	111 597	Villedieu, Mercure Galant, Sales, Cyrano de B., Racine, La Fayette
1680	5	58 179	Pradon, Papin, Bossuet, Campistron, La Bruyère
1690	4	84 277	Campistron, Molière, Perrault, Pradon

2.2 Principes de traduction automatique neuronale

La normalisation automatique de la langue peut être produite de différentes manières. On trouve ainsi des méthodes utilisant (par ordre chronologique d'apparition et de manière non exhaustive) une simple table de substitution, des règles, des modèles statistiques ou encore des réseaux de neurones. Pour la présente étude, nous avons retenu la dernière de ces approches. Les réseaux de neurones constituent l'état de l'art actuel en la matière, et sont les mieux à même de gérer les cas les plus complexes (cf. Bollmann 2019 pour une évaluation détaillée des différentes approches).

En traitement automatique des langues, l'architecture standard pour réaliser des tâches qui consistent à transformer une séquence (dans notre cas une phrase, ou une sous-phrase) de texte en une autre séquence de texte (on parle de tâche *sequence-to-sequence*, en abrégé *seq2seq*) est le modèle « encodeur-décodeur », qui utilise un réseau de neurones (cf. fig. 1). La terminologie vient de l'idée, d'un point de vue crytpographique, que le texte d'origine peut être vu comme une version broyée/chiffrée du texte que l'on souhaite produire, et la tâche consiste à retrouver les bonnes transformations de ce texte pour le décoder. Le rôle de l'encodeur est de faire une représentation des mots du texte d'origine qui soit adaptée pour retrouver comment bien décoder. Le décodeur utilise donc ces représentations du texte original pour prédire le texte à retrouver (dans notre cas, sa version normalisée) mot après mot de gauche à droite, en utilisant toujours les informations sur les mots du contexte gauche (c'est-à-dire ceux qui ont déjà été prédits) ainsi, naturellement, que les représentations du texte d'origine produites par l'encodeur.

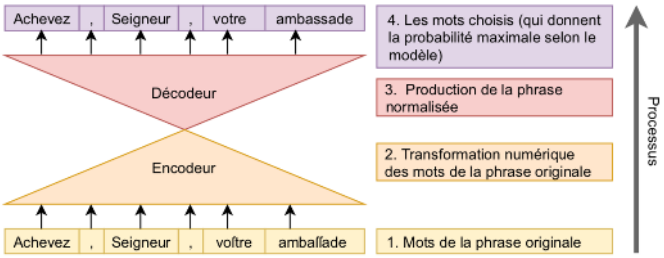

Fig. 1. Le modèle encodeur-décodeur pour les tâches de transformation de séquences de texte, appliqué sur un exemple de normalisation d'une phrase du français moderne.

La transformation appliquée par le modèle ne s'applique pas sur les formes dites « de surface » du texte avec des règles discrètes et bien interprétables (par exemple en insérant, supprimant ou échangeant les caractères des mots). Il s'agit d'une transformation numérique complexe des mots originaux, où les mots d'origine sont représentés par des vecteurs à grande dimension (plusieurs centaines de dimensions), que l'on appelle les plongements lexicaux (*word embeddings*). Ceci permet au modèle de faire des observations bien plus fines et abstraites que ce qui serait possible

en étudiant les transformations directement sur la graphie seule. Le réseau de neurones est une grande fonction mathématique qui, comme n'importe quelle fonction mathématique, prend en entrée certaines valeurs (ici les représentations vectorielles des mots d'origine) et les transforme. Ce qu'il produit en sortie dans le cas de la transformation de texte est, à chaque étape de décodage, une distribution de probabilités sur le vocabulaire de la langue de sortie du modèle. Chaque mot du français contemporain est donc associé à sa probabilité d'être le bon mot à chaque étape du décodage. La prédiction finale du modèle est celle qui donne la probabilité maximale de la séquence complète des mots prédits.

La bonne performance du modèle dépend de la proximité entre d'un côté l'opération que l'on souhaite modéliser (dans notre cas, la normalisation) et d'un autre côté la transformation mathématique réalisée par le modèle. Les valeurs qui sont utilisées pour réaliser cette transformation (les « paramètres » du modèle) doivent être calculées au cours d'un processus appelé entraînement du modèle. Dans notre cas, cet entraînement, ou apprentissage, est dit supervisé, ce qui veut dire qu'il sera réalisé en présentant successivement au modèle des exemples d'entrée (dans notre cas, des phrases en français (pré)classique) associés à la sortie correspondante attendue, ou « vérité terrain » (dans notre cas, leur équivalent normalisé). L'ensemble de ces exemples constitue ce que l'on appelle le corpus d'entraînement du modèle. En l'espèce, nous avons utilisé une sous-partie du corpus parallèle FreEM$_{norm}$ appelée « jeu d'entraînement », d'autres sous-parties étant réservées pour étudier le comportement du modèle (jeu dit « de développement ») et pour en évaluer les performances (jeu « de test »)[v]. Au cours de l'apprentissage, les paramètres du modèle, qui sont initialisés aléatoirement, sont mis à jour successivement, exemple après exemple, de façon à rapprocher ce que produit le modèle de la vérité terrain. Ces mises à jour successives s'appuient sur ce qu'on appelle une fonction de coût (*loss function*), qui permet de quantifier la distance entre la prédiction d'un modèle pour une entrée donnée et la vérité terrain qu'il aurait dû idéalement produire (dans notre cas, la version normalisée attendue telle que fournie par FreEM$_{norm}$). La méthode mathématique permettant de modifier les paramètres du modèle grâce à l'application de la fonction de coût sur chacun des exemples du corpus d'entraînement s'appelle rétropropagation (*backpropagation*).

Comme décrit ci-dessus, la capacité des réseaux de neurones à appliquer des transformations complexes ouvre des possibilités nouvelles par rapport aux approches qui se limitent à ne regarder que la forme de surface des mots. Le modèle peut ainsi apprendre toute sorte de transformation attestée de façon suffisante dans les données d'entraînement, telles que des transformations de nature graphique, lexicale, syntaxique, stylistique, etc. Au final, c'est le modèle qui induit des connaissances lui-même à partir des données. En revanche, ce que le modèle apprend est difficile à interpréter directement. Pour cela, l'approche la plus simple consiste à analyser le comportement du modèle sur un certain nombre d'exemples choisis, préférablement des exemples qui ne font pas partie du corpus d'entraînement.

2.3 Premiers résultats

L'évaluation du modèle de normalisation est effectuée sur le jeu de test du corpus FreEM$_{norm}$, qui ne comprend que des exemples non inclus dans les données d'entraînement. Plusieurs métriques d'évaluation sont utilisées à partir des prédictions du modèle et des normalisations de référence pour estimer la qualité des normalisations produites par le modèle. Traditionnellement, en traduction automatique, des méthodes à base de comparaisons de séquences de mots ou de caractères sont utilisées pour juger du degré de similarité entre une prédiction et une référence, les plus fréquemment utilisées étant Bleu (Papineni *et al.* 2002), qui se calcule sur les séquences de mots, et ChrF (Popovi 2015), qui se calcule sur les séquences de caractères. Ces métriques sont fortement critiquées en traduction automatique pour leur manque de capacité à récompenser de bonnes prédictions qui diffèrent de la référence (à cause de synonymes, de paraphrases, etc.) et à pénaliser certaines erreurs graves, comme la présence ou non de la négation, qui impacte très peu de mots.

Ces métriques sont en fait plus adaptées pour la tâche de normalisation, pour laquelle aucune variation lexicale n'est possible (l'anglais *owl* peut se traduire en « chouette » ou en « hibou », mais « ambaſſade » ne peut se normaliser qu'en « ambassade »). Une autre caractéristique de notre tâche de normalisation est son caractère monotone : contrairement à ce qui se passe généralement en traduction automatique, la normalisation ne change pas l'ordre des mots. Ceci nous permet de définir une troisième métrique, l'exactitude lexicale, encore plus adaptée, qui consiste à calculer le taux de mots correctement prédits parmi l'ensemble des mots de la référence, après l'avoir alignée avec le (ou les) mot(s) correspondants dans le texte prédit par le modèle au moyen d'une matrice de Levenshtein comme celle décrite plus loin dans le tab. 4.

Pour fournir une meilleure idée du niveau de performance de notre modèle, nous comparons les résultats d'évaluation contre les résultats de quatre autres systèmes :

1. Fonction d'identité : aucune transformation n'est appliquée sur le texte d'origine. Cette *baseline*[vi] permet de se rendre compte du taux de performance obtenu lorsqu'aucune normalisation n'est effectuée.
2. Méthode à base de règles : cette *baseline* un peu moins simpliste que la précédente consiste à appliquer une cinquantaine d'expressions régulières, développées en quelques dizaines de minutes par l'examen statistique manuel du jeu d'entraînement de FreEM$_{norm}$.
3. Méthode à base de règles + lexique : cette dernière *baseline* est une extension de la précédente. Elle consiste à appliquer à la sortie de la précédente *baseline* un filtre qui s'appuie sur le lexique Le*fff* (Sagot 2010) comme suit : chacun des tokens de la normalisation produite par l'application des règles est recherché dans le lexique en tolérant certaines variations simples (rajout de diacritiques, remplacement d'un « s » post-vocalique par un accent circonflexe, etc.) ; si cette recherche renvoie exactement un seul résultat, alors le token est remplacé par ce résultat (par exemple « estre » ne correspond qu'à « être » et sera donc remplacé par lui) ; si cette recherche ne renvoie aucun résultat ou en renvoie plusieurs, le token est laissé inchangé.
4. Outil ABA : méthode statistique décrite ci-dessous dans la sec. 3.1.

Les résultats, présentés dans le tab. 3, montrent que le modèle neuronal dépasse significativement les autres modèles selon les trois métriques. Notons que les scores de la première *baseline* (la fonction d'identité) atteint déjà des scores élevés, ce qui indique qu'un grand nombre de mots doivent être laissés intacts par le processus de normalisation.

Modèle	Exactitude lexicale	BLEU	ChrF
Fonction d'identité	0,724	40,246	0,738
Règles	0,887	72,474	0,899
Règles + Le*fff*	0,905	76,899	0,917
ABA	0,947	87,700	0,958
Notre modèle neuronal	**0,962**	**91,762**	**0,968**

Tableau 3. Résultats de normalisation sur le jeu de test de FreEM$_{norm}$ avec les trois métriques : Bleu (scores allant de 0 à 100), ChrF (de 0 à 1) et exactitude lexicale (de 0 à 1). Les meilleurs scores selon chaque métrique sont indiqués en gras.

3 Vers l'analyse scriptométrique de masse

La normalisation automatique permet de produire le point de comparaison nécessaire à l'analyse scriptométrique, ce qui n'est, à nouveau, pas une tâche simple, et pose deux problèmes. D'une part,

Source : Digne de paroître à ſes yeux ? *Référence* : Digne de paraître à ses yeux ? *Prédiction* : Digne de paraître à ses yeux ?
Source : Le veuë sur vos yeux, l'oreille à vostre voix *Référence* : Le vue sur vos yeux, l'oreille à votre voix, *Prédiction* : Le vue sur vos yeux, l'oreille à votre voix,
Source : REſtes infortunez du plus beau ſang du monde, *Référence* : REstes infortunés du plus beau sang du monde, *Prédiction* : **Rîtes** infortunés du plus beau sang du monde,
Source : Ie ne ſçay ce que ie ferois comme Amant, reprit fierement Stenius, mais ie ſçay bien que comme Amy d'Horace, ie ne vous dois rien dire où il ait intereſt : *Référence* : Je ne sais ce que je ferais comme Amant, reprit fièrement Sténius, mais je sais bien que comme Ami d'Horace, je ne vous dois rien dire où il ait intérêt : *Prédiction* : Je ne sais ce que je ferais comme Amant, reprit fièrement **Stenius**, mais je sais bien que comme Ami d'Horace, je ne vous dois rien dire où il ait intérêt :

Fig. 2. Quelques exemples de normalisation par le modèle neuronal. Les erreurs sont signalées en gras.

une première analyse de grande ampleur sur le Grand Siècle dépend de corpus non normalisés de grande taille, corpus que nous n'avons pas. D'autre part, la variation graphique au fil du temps est concomitante avec d'autres évolutions, notamment lexicales et thématiques, susceptibles de brouiller l'analyse. Ainsi, nous verrons qu'il est relativement aisé de prédire la décennie au cours de laquelle un texte a été écrit à partir de sa version normalisée : une telle prédiction, par définition, ne s'appuie pas sur l'état du système graphique, mais sur des indices notamment lexicaux. Il convient donc idéalement d'isoler au mieux l'évolution graphique en elle-même.

Nous avons étudié l'évolution du système graphique au cours du XVII^e^ siècle par deux approches complémentaires qui s'appuient sur deux outils de normalisation mentionnés ci-dessus, à savoir d'une part l'outil ABA et d'autre part notre propre modèle neuronal. Dans le premier cas, nous utilisons avant tout notre corpus parallèle FrEEM$_{norm}$, alors que dans le deuxième cas nous tirons également parti de FrEEM$_{max}$. Nous décrivons successivement les principaux enseignements obtenus par ces deux approches.

3.1 Une approche fondée sur l'alignement de corpus parallèles

Il nous faut identifier précisément les portions de mots qui diffèrent entre la version originale et la version normalisée, et regrouper les différences similaires, par exemple ayant une même origine historico-linguistique, ou le même type d'opérations en termes d'ajout, suppression ou modification de caractères. Pour ce faire, nous appliquons dans un premier temps l'outil ABA (Poinhos 2020) sur le corpus FrEEM$_{norm}$, qui permet de donner des premiers résultats sur un corpus de petite taille dont la fiabilité est garantie (contrairement à un corpus normalisé automatiquement, qui contiendrait inévitablement des erreurs). Chaque segment de phrase de FrEEM$_{norm}$ est découpé en mots, nettoyé de sa ponctuation, puis la version originale et modernisée sont alignées au niveau des mots à l'aide de l'algorithme de Needleman-Wunsch (Needleman et Wunsch 1970), en utilisant la distance de Levenshtein entre chaque paire de mots d'un même segment de phrase (Levenshtein 1966) en version originale et normalisée[vii].

Dans un deuxième temps, pour chacune des paires de mots alignés, la version originale et la version normalisée sont alignées au niveau du caractère, toujours en utilisant l'algorithme de Needleman-Wunsch, mais en utilisant une matrice de substitution spécifique pour permettre non seulement à des lettres identiques d'être alignées, mais aussi des lettres considérées comme

proches en français (pré)classique et en français contemporain (présence/absence de diacritique, de ligatures. . .). Par exemple[viii], alors que les lettres identiques bénéficient d'un score de substitution de 4, des lettres différant seulement par l'accentuation ou la cédille bénéficient d'un score de 2, tout comme *ſ* et *s* ou *s* et *ß* par exemple. D'autres couples de lettres bénéficient d'un score de 1, comme *u* et *v*, *s* et *z* ou encore *n* et *m*. À l'inverse, un score de -1 est attribué aux couples de lettres distinctes ne faisant pas l'objet de telles exceptions, ainsi qu'à la suppression ou l'insertion d'un caractère. Ces scores ont été fixés expérimentalement suite à des tests sur les résultats d'alignement obtenus sur la version de mars 2020 du corpus FrEEM$_{norm}$, qui contenait alors une trentaine de textes.

Cette exécution de l'algorithme de Needleman-Wunsch pour obtenir un alignement au niveau des caractères est illustrée dans la matrice du tab. 4, où chaque nombre représente le score de similarité du meilleur alignement trouvé entre le préfixe d'*Apoſtre* et d'*Apôtre* jusqu'à cette case. Il est précédé d'une flèche indiquant de quelle case venir pour obtenir ce meilleur alignement. Par exemple, pour obtenir le meilleur alignement entre *Apof* et *Apô*, il faut considérer le meilleur alignement entre *Apo* et *Apô* (qui a un score de 10) puis faire une insertion de *f*, qui a un score de -1, ce qui fournit au total un score de 9. Si on avait préféré considérer d'abord le meilleur alignement entre *Apof* et *Ap*, qui a un score de 6, puis faire une suppression du *ô*, qui a un score de -1, on aurait obtenu un alignement avec un score de 5, donc inférieur à l'optimal. En cas d'insertion ou de suppression lors de cette étape d'alignement, on utilise le caractère ¤ afin d'obtenir deux mots de même longueur à la fois en version originale et normalisée. Ainsi, à l'issue de cette seconde étape d'alignement, le mot « Apoſtre » en version originale est mis en correspondance avec « apô¤tre » en version normalisée pour obtenir un alignement caractère par caractère.

Tableau 4. Matrice de similarité des préfixes pour la version originale et normalisée d'*Apoſtre*. Les flèches indiquent la case précédente sur le chemin optimal pour calculer la similarité entre deux préfixes, l'un du mot sur la première ligne, l'autre du mot de la première colonne. Sur ce chemin optimal, le vert indique l'égalité, le rouge la substitution et le bleu la suppression.

	A	p	o	ſ	t	r	e
A	↘ 4	→ 3	→ 2	→ 1	→ 0	→ -1	→ -2
p	↓ 3	↘ 8	→ 7	→ 6	→ 5	→ 4	→ 3
ô	↓ 2	↓ 7	↘ 10	→ 9	→ 8	→ 7	→ 6
t	↓ 1	↓ 6	↓ 9	↓ 8	↘ 13	→ 12	→ 11
r	↓ 0	↓ 5	↓ 8	↓ 7	↓ 12	↘ 17	→ 16
e	↓ -1	↓ 4	↓ 7	↓ 6	↓ 11	↓ 16	↘ 21

Enfin, pour chaque mot du corpus, ses versions originale et normalisée sont analysées, caractère par caractère, pour détecter, en cas de caractères différents à la même position, la règle de normalisation qui s'applique, ou signaler qu'aucune règle n'a été identifiée le cas échéant. 72 règles ont été définies en s'appuyant sur la bibliographie et sur les différences observées dans le corpus. Par exemple[ix], la règle *lettre ramiste* est détectée si un *i*, un *j*, un *u* ou un *v* est présent dans le mot en version originale associé respectivement à un *j*, un *i*, un *v* ou un *u* en version normalisée. Il est ainsi possible de produire une première description de l'évolution des graphies dans le corpus FrEEM$_{norm}$ (cf. fig. 3).

En complément de cet outil d'analyse par comparaison entre version originale et normalisée d'un texte, un outil de normalisation est dérivé de l'analyse, d'une part en appliquant un dictionnaire de remplacement de mots appris sur un corpus d'entraînement et d'autre part en appliquant des règles de remplacements de caractères, prises en compte seulement si le résultat de leur application sur un mot en orthographe originale produit un mot trouvé dans un dictionnaire de français actuel.

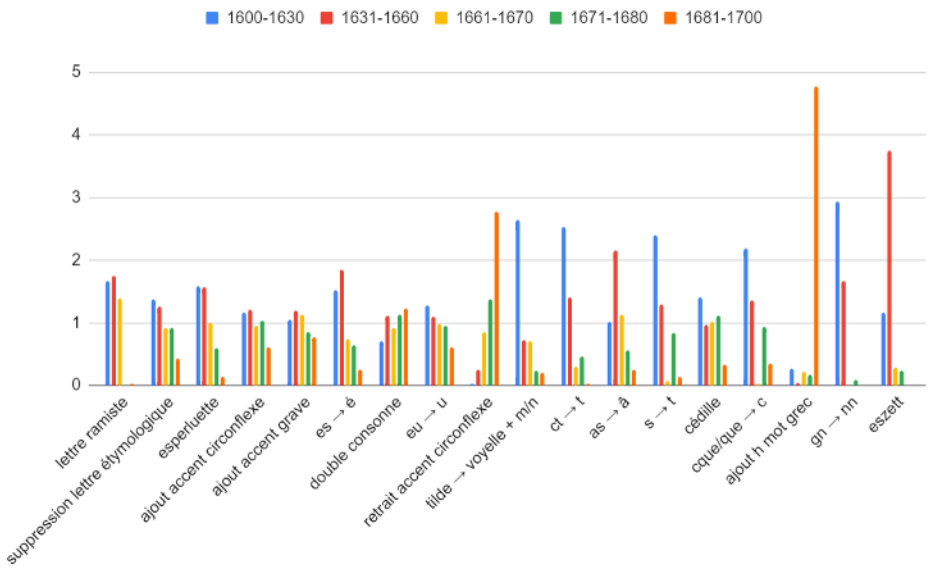

Fig. 3. Évolutions significatives au cours du XVII^e siècle. Une valeur de 1 correspond à la valeur moyenne de la fréquence de la règle (nombre d'occurrences divisé par nombre total de mots de l'ensemble du corpus). Une valeur de 1.5 sur la période 1600-1630 signifie que la règle est détectée en moyenne 50% de plus sur cette période que sur la totalité du 17e siècle, dans le corpus FrEEM$_{norm}$.

3.2 Une approche fondée sur notre modèle neuronal de normalisation

Parallèlement à cette première approche, il est aussi possible d'exploiter le normaliseur automatique neuronal décrit ci-dessus. Ce dernier ne fonctionnant pas avec les formes « de surface » mais avec leur représentation abstraite sous forme de vecteurs à grande dimension, la méthode présentée *supra* s'avère impraticable. Il convient donc de trouver une solution pour contourner cet écueil.

3.2.1 Validation de la sensibilité du modèle de normalisation au changement du système graphique

Avant d'étudier l'évolution du système graphique utilisé au cours du XVII^e siècle, il est d'abord utile de vérifier qu'un tel modèle automatique puisse y être sensible. Nous testons cette sensibilité en entraînant un modèle de normalisation à prédire, à partir d'une phrase d'origine (la « source »), non seulement sa version normalisée mais aussi sa décennie de rédaction (la « cible »). Nous choisissons une modélisation simple pour prédire cette information supplémentaire, qui consiste à prédire la décennie sous forme d'un « pseudo-mot » (ou plus précisément un « pseudo-token ») au début de la version normalisée (Kobus, Crego et Senellart 2017 ; Sennrich, Haddow et Birch 2016 ; Yamagishi *et al.* 2016) comme s'il en faisait partie (cf. ex 2)[x,xi]. Ensuite, nous extrayons ce premier « pseudo-mot » de la séquence produite et nous le comparons à la vraie décennie afin d'évaluer la performance du modèle. Si le modèle réussit à prédire les bonnes décennies, nous pouvons conclure qu'il a pu exploiter les informations des textes d'origine et est donc sensible à l'évolution dans le temps.

Source (phrase originale) : Achevez, Seigneur, voſtre ambaſſade
Cible (décennie+phrase normalisée) : <dec=1660> Achevez, Seigneur, votre ambassade (2)

Cependant, les informations dans les phrases d'origine sont de deux natures différentes : il y

a le système graphique utilisé (que nous souhaitons que le modèle capture) mais également, entre autres, le contenu lexical, qui diffère selon les textes, les genres et les périodes — rien n'empêche d'envisager, par exemple, que certains mots soient associés plus fortement à certaines décennies. Afin d'évaluer s'il y a suffisamment d'indices dans le système graphique lui-même, il est important de pouvoir soustraire l'apport des autres indices potentiels, et notamment du contenu lexical, à la prédiction. Pour cela, nous réalisons une expérience de contrôle qui consiste à reproduire la même tâche mais à l'envers : à partir d'un texte déjà normalisé en français contemporain (la « source »), notre nouveau modèle de contrôle apprend à prédire le texte non normalisé et la décennie de rédaction du texte (la « cible ») : c'est un modèle de « dénormalisation » (cf. ex 3). Dans cette expérience, l'information graphique d'origine étant absente, puisque l'on fournit en entrée le texte normalisé, le modèle ne peut s'appuyer que sur le contenu des textes, et notamment les informations lexicales, pour prédire la décennie de rédaction. La comparaison entre la performance de cette expérience de contrôle et le modèle de normalisation décrit ci-dessus qui, rappelons-le, prédit la version normalisée d'un texte mais également sa décennie de rédaction à partir du texte d'origine, permet donc d'observer si l'information liée au système graphique permet de faire de meilleures prédictions. Si c'est le cas, c'est que le système graphique en soi évolue dans le temps d'une façon que notre modèle de normalisation a réussi à capturer.

Source (phrase normalisée) : Achevez , Seigneur , votre ambassade
Cible (décennie+phrase originale) : <dec=1660> Achevez , Seigneur , voſtre ambaſſade (3)

Nous évaluons chaque modèle sur sa capacité à prédire la bonne décennie de rédaction pour les 2 443 phrases du jeu de développement du corpus FreEM$_{norm}$. Les résultats (cf. fig. 4) montrent de bonnes performances pour les deux modèles (respectivement 57% et 49% de prédictions correctes) et que la plupart des prédictions se situent autour de la diagonale (c'est-à-dire qu'elles ne s'éloignent pas beaucoup de la vérité terrain), même si les deux modèles sur-prédisent les décennies 1660 et 1670. Il est important de remarquer que le modèle de normalisation, qui a accès aux informations lexicales et graphiques, a une meilleure performance que le modèle de contrôle, qui n'a accès qu'aux informations lexicales. Ceci montre que, même si les informations lexicales sont un indicateur fort pour prédire la date de rédaction d'un texte, les informations relevant du système graphique sont utiles et exploitables par le modèle, qui est donc sensible à son évolution.

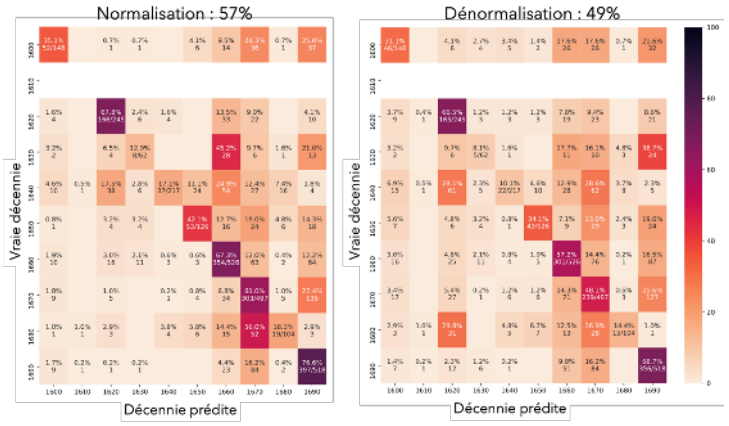

Fig. 4. Résultats de la prédiction de la décennie de rédaction à partir de la phrase d'origine (à gauche) et de la phrase normalisée (à droite, expérience de contrôle) sous la forme de matrice de confusion.

3.2.2 Extraction des informations concernant l'évolution dans le système graphique

Comment extraire les connaissances du modèle maintenant que nous savons qu'il est sensible à l'évolution du système graphique ? C'est une problématique de recherche active en traitement

automatique des langues neuronal, car le fonctionnement de ces modèles est difficile à interpréter. Nous choisissons d'extraire les connaissances apprises par le modèle en analysant les transformations automatiques appliquées d'une façon similaire à la méthode décrite dans la sec. 3.1, c'est-à-dire en alignant le texte donné au modèle avec le texte transformé qu'il produit. Comme pour l'expérience de contrôle décrite précédemment, nous entraînons un modèle de dénormalisation qui part d'un texte écrit avec le système graphique du français contemporain et produit un texte en français (pré)classique. Cette fois-ci, nous conditionnons la dénormalisation sur une des décennies du XVII^e^ s., afin de pouvoir comparer les transformations à différents moment du siècle, ce qui implique d'entraîner et d'appliquer le modèle sur les données préparées comme dans l'exemple 4, où la décennie est ajoutée comme pseudo-mot supplémentaire au début de la phrase à dénormaliser.

Source (décennie+phrase normalisée) : <dec=1660> Achevez, Seigneur, votre ambassade
Cible (phrase originale) : Achevez, Seigneur, voſtre ambaſſade (4)

Avec une telle approche, il est possible d'appliquer le modèle sur une même phrase en conditionnant sur chacune des dix décennies pour en produire dix versions artificielles, dont le système graphique reflète ce que le modèle a appris à produire pour chacune des dix décennies (cf. ex. 5). Ainsi, il devrait être possible d'analyser l'évolution de la graphie utilisée au cours du siècle en utilisant un corpus strictement comparable pour chaque décennie (puisqu'il s'agit du même texte de départ), contrairement à un corpus réel, nécessairement déséquilibré.

Phrase artificielle 1 : <dec=1610> Acheués, Seigneur, voſtre ambaſſade
Phrase artificielle 4 : <dec=1640> Acheuez, Seigneur, voſtre ambaſſade (5)
Phrase artificielle 6 : <dec=1660> Achevez, Seigneur, vôtre ambaſſade

Le texte de départ que nous transformons pour chacune des décennies n'est, cette fois, plus le corpus FrEEM$_{norm}$, mais le sous-ensemble du corpus FrEEM$_{max}$ constitué de textes normalisés, soit 43 405 phrases. Une fois le texte dénormalisé artificiellement pour chaque décennie, nous alignons au niveau du mot le texte d'origine et les textes transformés en appliquant l'algorithme de Levenshtein au niveau des caractères. Cet alignement au niveau des mots permet d'extraire des paires en français contemporain et leur version dénormalisée. Nous comptons le nombre d'occurrences des transformations où le mot a changé de graphie et nous comparons les différents mots changés pour voir s'il y a des généralisations saillantes (cf. fig. 5).

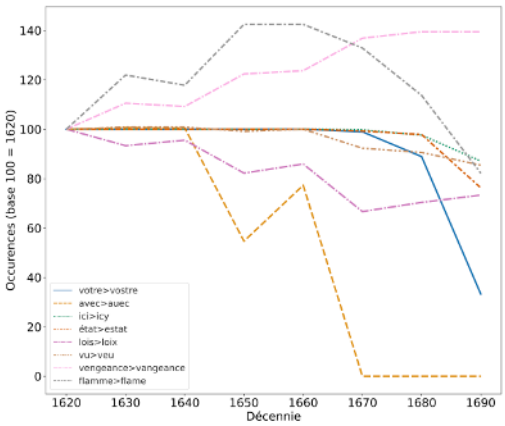

Fig. 5. L'évolution de certaines correspondances graphiques extraites du modèle.

4 Conclusion

Ces premières expériences permettent de valider une approche automatique de l'analyse des graphies sur des corpus de grande ampleur. Il est désormais possible, à partir de n'importe quel texte en français (pré)classique, d'analyser automatiquement son système graphique et, avec un ensemble de textes, de dégager quelques premières observations quant à l'évolution du système graphique au XVII^e^ s. Cette évolution semble, à première vue, dessiner une courbe différente de

celle obtenue par Cl. Vachon pour le siècle précédent (Vachon 2010, p. 253) : sans reflux aucun, l'évolution serait progressive tout au long du Grand Siècle (cf. notamment la fig. 3).

Le gain est donc double, voire même triple : nous sommes non seulement (a) capable de normaliser automatiquement la langue, mais aussi (b) d'analyser le changement linguistique sur la (très) longue durée (c) au moyen d'une méthodologie innovante, qui tire pour la première fois pleinement partie de l'outil informatique. De simple outil d'accélération de relevés linguistiques, il devient rouage central de notre analyse diachronique. Cette dernière se démarque donc des précédents travaux en linguistique de corpus, d'avec lesquels elle diverge par nature, et non simplement par degré, en ouvrant la porte à une approche véritablement computationnelle de l'évolution de la langue (plutôt que de son changement, cf. Marchello-Nizia 2011) dans toute sa profondeur historique.

Simple preuve de concept, ce papier se veut inaugural d'un travail de fond sur l'évolution de la langue, que nous nous promettons donc de continuer. Outre l'amélioration des corpus, de nombreux problèmes restent en suspens, comme celui de l'angle mort laissé par des graphies archaïsantes (par ex. *corps*<*corpus*) passées dans la langue contemporaine et dont les lettres en surcharge restent ignorées par une approche contrastive comme la nôtre. Elle reste néanmoins assez robuste pour dès à présent ouvrir de nouvelles perspectives de recherche autour de ce champ délaissé qu'est l'analyse des systèmes graphiques à l'époque (pré)classique.

Références bibliographiques

Académie française (1863). Cahiers de remarques sur l'orthographe françoise pour estre examinez par chacun de Messieurs de l'Academie, avec des observations de Bossuet, Pellisson, etc. éd. Charles Joseph Marty-Laveaux. Jules Gay. Disp. à l'adr. : <https://books.google.ch/books?id=u5Y5AQAAIAAJ>.

Amatuzzi, A. *et al.* (2020). Changement linguistique et périodisation du français (pré)classique : deux études de cas à partir des corpus du RCFC. *Journal of French Language Studies* 30.3. Publisher : Cambridge University Press, p. 301-326.

Baddeley, S. (1993). L'Ortographie française au temps de la Réforme. Genève : Droz.

Biedermann-Pasques, L. (1992). Les Grands Courants orthographiques au XVIIe siècle et la formation de l'orthographe moderne, Impacts matériels, interférences phoniques, théories et pratiques (1606–1736). Tübingen : Max Niemeyer Verlag.

Bollmann, M. (2018). Normalization of Historical Texts with Neural Network Models. Thèse de doct. Bochum : Ruhr-Universität Bochum. Disp. à l'adr. : <https://hss-opus.ub.ruhr-uni-bochum.de/opus4/frontdoor/index/index/docId/6213>.

— (2019). A Large-Scale Comparison of Historical Text Normalization Systems. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*. NAACL-HLT 2019. Minneapolis, Minnesota : Association for Computational Linguistics, p. 3885-3898.

Catach, N. (2001). Histoire de l'orthographe française. Sous la dir. de R. Honvault. Ed. posthume. 9. Paris, Genève : H. Champion.

Damerau, F. J. (1964). A technique for computer detection and correction of spelling errors. *Communications of the ACM* 7.3, p. 171-176.

Dees, A. (1985). Dialectes et scriptae à l'époque de l'ancien français. *Revue de Linguistique Romane* 49, p. 87-117.

Duval, F. (2015). Les éditions de textes du XVIIe siècle. *Manuel de la philologie de l'édition*. Sous la dir. de D. Trotter. Berlin, Boston : De Gruyter, p. 369-394.

Ernst, G., éd. (2019). Textes français privés des XVIIe et XVIIIe siècles. 2 t. Berlin : De Gruyter.

Gabay, S. (2014). Pourquoi moderniser l'orthographe ? Principes d'ecdotique et littérature du XVIIe siècle. *Vox Romanica* 73.1, p. 27-42.

Gabay, S. et L. Barrault (2020). Traduction automatique pour la normalisation du français du XVIIe siècle. *Proceedings of TALN 2020*. Nancy, France. Disp. à l'adr. : <https://hal.archives-ouvertes.fr/hal-02596669>.

Gabay, S., A. Bartz et Y. Deguin (2020). CORPUS17 : a philological corpus for 17th c. French. *Proceedings of the 2nd International Digital Tools & Uses Congress (DTUC '20)*. Hammamet, Tunisia.

Gabay, S., M. Riguet et L. Barrault (2019). A Workflow For On The Fly Normalisation Of 17th c. French. *Digital Humanities 2019 Conference Abstracts*. DH2019. Utrecht, The Netherlands : Alliance of Digital Humanities Organizations (ADHO). Disp. à l'adr. : <https://hal.archives-ouvertes.fr/hal-02276150>.

Goebl, H. (1982). Dialektometrie : Prinzipien und Methoden des Einsatzes der Numerischen Taxonomie im Bereich der Dialektgeographie. Wien : Verlag der Österreichischen Akademie der Wissenschaften.

— (2006). Sur le changement macrolinguistique survenu entre 1300 et 1900 dans le domaine d'oïl. Une étude diachronique d'inspiration dialetométrique. *Linguistica* 46, p. 3-43. Disp. à l'adr. : <https://revije.ff.uni-lj.si/linguistica/article/view/4186>.

— (2011). 22. Dialectometry and quantitative mapping. *Language Mapping*. T. 2. Berlin, Boston : De Gruyter Mouton, p. 433-457. Disp. à l'adr. : <https://doi.org/10.1515/9783110219166.1.433>.

— (2012). Introduction aux problèmes et méthodes de l'« École dialectométrique de Salzbourg » (avec des exemples gallo-, italo- et ibéroromans). *Proceedings of the International Symposium on Limits and Areas in Dialectology*. Lisbonne, Portugal : Centro de Linguística da Universidade de Lisboa, p. 117-166.

Grieve, J. (2009). A corpus-based regional dialect survey of grammatical variation in written Standard American English. Thèse de doct. Flagstaff, AZ : Northern Arizona University.

Hamming, R. W. (avr. 1950). Error detecting and error correcting codes. *The Bell System Technical Journal* 29.2, p. 147-160.

Kessler, B. (1995). Computational dialectology in Irish Gaelic. *Proceedings of the Seventh Conference of the European Chapter of the Association for Computational Linguistics*. Dublin, Irlande. Disp. à l'adr. : <https://aclanthology.org/E95-1009>.

Kobus, C., J. Crego et J. Senellart (2017). Domain Control for Neural Machine Translation. *Proceedings of Recent Advances in Natural Language Processing*. Varna, Bulgarie, p. 372-378.

Levenshtein, V. (1966). Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics doklady* 10.8, p. 707-710.

Marchello-Nizia, C. (2011). Écrire une nouvelle grammaire historique du français à la lumière de l'histoire des descriptions de la langue. *Vers une histoire générale de la grammaire française ?* Paris : Champion, p. 45-60. Disp. à l'adr. : <https://bcl.cnrs.fr/rubrique137>.

Needleman, S. B. et C. D. Wunsch (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology* 48.3, p. 443-453.

Nerbonne, J. et W. Heeringa (2010). 31. Measuring dialect differences. *Theories and Methods : An International Handbook of Linguistic Variation*. T. 1. De Gruyter Mouton, p. 550-567.

Papineni, K. *et al.* (2002). Bleu : a Method for Automatic Evaluation of Machine Translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphie, Pennsylvanie, États-Unis, p. 311-318. Disp. à l'adr. : <https://aclanthology.org/P02-1040>.

Pellat, J.-C. (1998). Les mots graphiques dans des manuscrits et des imprimés du XVIIe siècle. *Langue française* 119.1, p. 88-104. Disp. à l'adr. : <http://www.persee.fr/doc/lfr_0023-8368_1998_num_119_1_6261>.

Poinhos, J. (2020). ABA (Alignment-Based Approach). Version 1. Disp. à l'adr. : <https://github.com/johnseazer/aba>.

Popovi, M. (2015). chrF : character n-gram F-score for automatic MT evaluation. *Proceedings of the Tenth Workshop on Statistical Machine Translation*. Lisbonne, Portugal : Association for Computational Linguistics, p. 392-395. Disp. à l'adr. : <https://aclanthology.org/W15-3049>.

Sagot, B. (2010). The Le*fff*, a freely available and large-coverage morphological and syntactic lexicon for French. *Proceedings of the 7th international conference on Language Resources and Evaluation (LREC 2010)*. La Valette, Malte. Disp. à l'adr. : <https://hal.inria.fr/inria-00521242>.

Séguy, J. (1971). La relation entre la distance spatiale et la distance lexicale. *Revue de Linguistique Romane* 35, p. 335-57.

— (1973). La dialectométrie dans l'Atlas linguistique de la Gascogne. *Revue de linguistique romane* 37, p. 1-24.

Sennrich, R., B. Haddow et A. Birch (2016). Controlling Politeness in Neural Machine Translation via Side Constraints. *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*. San Diego, Californie, États-Unis, p. 35-40.

Sorel, C. (2014). L'anti-roman, ou, L'histoire du berger Lysis, accompagnée de ses remarques : seconde édition du "Berger extravagant" revue et augmentée par l'auteur. ed. Anne-Élisabeth Spica. 2 t. 115. Paris : Honoré Champion éditeur.

Vachon, C. H. (2010). Le Changement linguistique au XVIe siècle : une étude basée sur des textes littéraires français. Strasbourg, France : ELiPhi, Éditions de linguistique et de philologie.

Vaugelas, C. F. d. (1647). Remarques sur la langue françoise, utiles à ceux qui veulent bien parler et bien escrire. Paris : Vve J. Camusat et P. Le Petit.

— (2009). Remarques sur la langue françoise. éd. Marzys, Zygmunt. Genève : Droz.

Yamagishi, H. *et al.* (2016). Controlling the voice of a sentence in Japanese-to-English neural machine translation. *Proceedings of the 3rd Workshop on Asian Translation (WAT2016)*, p. 203-210.

---

[i] On retrouve des hypothèses similaires dans les inspirants travaux de J.-Ch. Pellat, comme Pellat 1998.

[ii] Les normalisations *avecque* → *avec* ou *jusques* → *jusque* ne sont pas toujours souhaitables car elles font perdre une syllabe.

[iii] Cf. *ALAVAL - Atlas linguistique audiovisuel du francoprovençal valaisan*, `http://alaval.unine.ch/atlas?carte=41310`.

[iv] Étant donnée la longueur de la phrase classique, il a paru opportun de réduire cette dernière à des segments constitués automatiquement à partir de la ponctuation lorsque cela était possible. Une phrase contenant un point-virgule ou deux-points est ainsi découpée en segments.

[v] L'évaluation d'un modèle ne saurait se faire sur des exemples vus par le modèle au cours de son entraînement. Dans ce cas en effet, il suffirait au modèle d'apprendre « par cœur » l'ensemble des données d'entraînement pour obtenir un score parfait, sans avoir fait la moindre généralisation pertinente lui permettant de bien se comporter sur d'autres exemples. Les modèles sont donc toujours évalués sur des données non vues au cours de l'entraînement.

[vi] Une *baseline* (on trouve parfois le calque « ligne de base ») est un modèle intentionnellement trop simple mais destiné à servir de point de comparaison.

[vii] Quelques subtilités sont apportées à cet ajustement, comme *et* et *&* qui sont considérés comme équivalents

[viii] Ces scores spécifiques sont tous listés dans la fonction `init_submat_chars` du fichier `strings.py` d'ABA.

[ix] Ces règles sont toutes définies dans la fonction `find_diffs` du fichier `modern.py` d'ABA.

[x] Pour entraîner un tel modèle, nous préparons les données d'entraînement de sorte que chaque phrase cible soit préfixée d'un pseudo-token qui indique la vraie décennie de rédaction du texte, extraite des métadonnées. Lorsque le modèle est appliqué sur une nouvelle phrase, le modèle fera une prédiction de la décennie ainsi que de la normalisation de la phrase d'origine.

[xi] Des expériences préliminaires nous ont montré que prédire la décennie comme premier mot de la version normalisée produisait de meilleurs résultats que de la prédire comme dernier mot, mais également que de prédire uniquement la décennie (sans prédire la normalisation de la phrase d'origine).