Méthodes pour l'archéologie linguistique : datation par combinaison d'indices temporels

Anne Garcia-Fernandez[1] Anne-Laure Ligozat[1, 2] Marco Dinarelli[1] Delphine Bernhard[1]
(1) LIMSI-CNRS, BP133, 91403 Orsay cedex
(2) ENSIIE, 1 square de la résistance, 91000 Évry
{annegf,annlor,marcod,bernhard}@limsi.fr

Résumé. Dans cet article, nous présentons notre participation au défi fouille de texte (DEFT) 2011 à la tâche de datation d'un document. Notre approche est fondée sur une combinaison de plusieurs sous-systèmes, certains supervisés, d'autres non supervisés, et utilise plusieurs ressources externes comme Wikipédia, les Google Books n-grams ainsi que des connaissances sur les réformes orthographiques du français. Notre meilleur système obtient un score de 0,378 sur les portions de 300 mots et de 0,452 sur les portions de 500 mots, ce qui représente 37% de décennies correctes et 10% d'années correctes au premier rang sur les portions de 300 mots, et 42% de décennies correctes et 14% d'années correctes au premier rang sur les portions de 500 mots.

Abstract. In this article, we present a method for automatically determining the publication dates of documents, which was evaluated on a French newspaper corpus in the context of the DEFT 2011 evaluation campaign. Our approach is based on a combination of different sub-systems, both supervised and unsupervised, and uses several external resources, e.g. Wikipedia, Google Books n-grams, and etymological background knowledge about the French language. Our best system obtains a score of 0.378 on 300 words portions and 0.452 on 500 words portions. This represents 37% of correct decades and 10% of correct years at first rank on 300 words portions, and 42% of correct decades and 14% of correct years at first rank of 500 words portions.

Mots-clés : Analyse diachronique, classification de documents, apprentissage supervisé.

Keywords: Diachronic analysis, document classification supervised learning.

1 Introduction

En 2011, le DÉfi Fouille de Texte, DEFT, a proposé deux tâches. La première, à laquelle nous avons participé, consiste à identifier l'année d'extraits d'articles de journaux. Cette tâche s'inscrit dans la continuité de l'édition 2010 qui proposait notamment d'identifier la décennie d'un extrait de document. Pour déterminer automatiquement l'année d'un texte, nous avons choisi d'utiliser différentes méthodes et ressources, puis de les combiner. Dans cet article, nous présentons les méthodes que nous avons utilisées et les résultats que nous avons obtenus.

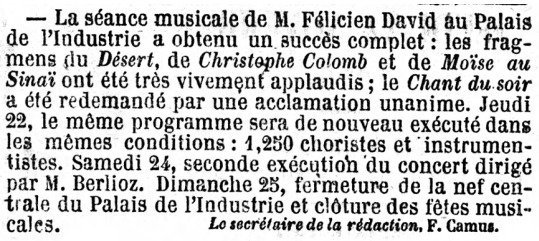
Une première section présente le corpus et les pré-traitements que nous avons effectués. La section suivante décrit de façon générale notre approche puis nous présentons les deux types d'approches que nous avons mis en œuvre. Dans la section 4, nous présentons des approches à base de ressources externes et indépendantes du corpus que nous nommons *méthodes chronologiques*. Dans la section 5, nous exposons les approches par apprentissage dites *méthodes de similarité temporelle*. Nous présentons les résultats en terme de F-mesure (telle que définie par DEFT et présentée dans (Grouin *et al.*, 2011)) et en terme de pourcentage d'années et de décennies identifiées correctement.

2 Corpus et prétraitements

Le corpus, décrit dans (Grouin *et al.*, 2011), est composé d'article de journaux issus de la base de donnée Gallica [1]. Il s'agit d'extraits d'articles publiés entre 1801 et 1944 de 300 ou 500 mots. Ces documents sont issus de la numérisation de journaux papiers (figure 2) et contiennent, comme nous pouvons le voir dans la figure 1, de nombreuses erreurs provenant du processus de reconnaissance optique des caractères (OCR).

La séance musicale de M. Félicien David au Palais de l'Industrie a obtenu un succès complet les fragmens du Désert, de Christophe Colomb et de Moïse au Sinaï ont été très vivemçnt applaudis ; le Chant du soir a été redemandé par une acclamation unanime. Jeudi 22, le même programme sera de nouveau exécuté dans les mêmes conditions : l,2S0 choristes et'instrumentistes. Samedi 24, seconde exécution du concert dirigé par M. Berlioz. Dimanche 2S, fermeture de la nef centrale du Palais de l'Industrie et clôture des fêtes musicales. Lotecfètairedela rédaction, F. Carani.

FIGURE 1: Version électronique d'un extrait de journal datant de 1855



— La séance musicale de M. Félicien David au Palais de l'Industrie a obtenu un succès complet : les fragmens du *Désert*, de *Christophe Colomb* et de *Moïse au Sinaï* ont été très vivement applaudis ; le *Chant du soir* a été redemandé par une acclamation unanime. Jeudi 22, le même programme sera de nouveau exécuté dans les mêmes conditions : 1,250 choristes et instrumentistes. Samedi 24, seconde exécution du concert dirigé par M. Berlioz. Dimanche 25, fermeture de la nef centrale du Palais de l'Industrie et clôture des fêtes musicales. *Le secrétaire de la rédaction*, F. Camus.

FIGURE 2: Image du même extrait de journal datant de 1855

2.1 Découpage du corpus d'entraînement

Le corpus de développement fourni par DEFT contient 3 596 portions de documents. Nous avons divisé ce corpus en deux parties : un ensemble d'entraînement (TRN) constitué de 2 396 portions et un ensemble de validation (DEV) constitué de 1 200 portions.

2.2 Lemmatisation

Afin de réduire la taille du vocabulaire des corpus, nous avons remplacé les mots par leurs lemmes, en appliquant le TreeTagger (Schmid, 1994). Ceci nous a permis de passer, pour le corpus TRN, d'un vocabulaire de 74 000 mots à un vocabulaire de 52 000 mots.

1. http://gallica.bnf.fr/

3 Description générale de l'approche

Notre approche est fondée sur l'utilisation de deux types de méthodes. Les méthodes dites chronologiques s'appuient sur des ressources externes et indépendantes du corpus d'entrainement fourni. Les méthodes de similarité temporelle sont quant à elle des approches par apprentissage fondées sur l'utilisation de la similarité cosinus et de modèles SVM.

4 Méthodes chronologiques

Les méthodes chronologiques ont pour objectif de déterminer des périodes de temps qui sont les plus probables pour chaque portion. Elle ne permettent pas d'estimer l'année précise de publication d'une portion.

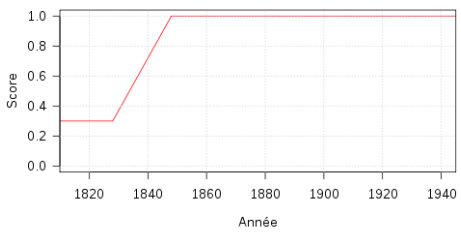
4.1 Dates de naissance de personnes

La présence d'un nom de personne peut être un indice de la date d'un texte (Albert *et al.*, 2010), puisque le document est nécessairement postérieur à l'année de naissance de cette personne. Afin d'utiliser cette information, nous souhaitions reconnaître les noms de personnes dans nos corpus, puis aller chercher leurs dates de naissance dans Wikipédia.

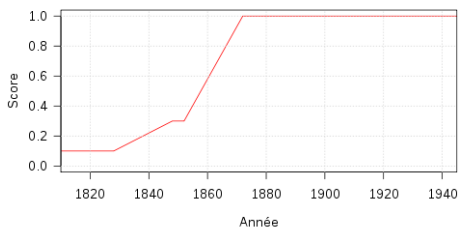
Nous avons dans un premier temps essayé d'appliquer un système de reconnaissance d'entités nommées au corpus d'entraînement (TRN), mais les résultats n'étaient pas suffisamment fiables. Nous avons donc employé une stratégie différente : nous avons collecté de façon automatique les années de naissance de personnes nées entre 1781 et 1944 en utilisant les catégories Wikipédia «Naissance_en_AAAA», qui regroupent des personnes étant nées à une année donnée et présentes dans Wikipédia. Nous avons ainsi recueilli environ 99 000 noms de personnes associés à leurs années de naissance, à partir desquelles nous en avons sélectionné 96 000 non ambiguës (par exemple, deux «Albert Kahn» avaient été trouvés), puisque nous n'avons pas de moyen simple de savoir de quelle personne il s'agit précisément.

Nous avons ensuite appliqué WMatch, un moteur d'expressions régulières [2] permettant notamment une annotation rapide de textes (Rosset *et al.*, 2008; Galibert, 2009), à chaque portion, afin de détecter la présence de ces noms de personnes dans nos corpus. Pour le corpus TRN, 529 noms de personnes ont été trouvés (concernant 375 portions), dont 16 (3%) correspondaient en réalité à des homonymes ou des annotations erronées : par exemple, Wikipédia a une entrée pour la romancière Colette, qui est par ailleurs un prénom, ce qui donnait lieu à des ambiguïtés.

Un score a ensuite été donné à chaque portion de chaque année possible, en fonction de la présence de noms de personnes. La figure 3 montre les scores obtenus dans le cas de la présence de deux noms de personnes, Jules Verne, né en 1828, et Antoni Gaudì, né en 1852.



(a) avec Jules Verne, né en 1828



(b) avec Jules Verne, né en 1828 et Antoni Gaudí, né en 1852

FIGURE 3: Scores en fonction de la présence de noms de personnes

Nous partons du principe qu'une portion citant une personne n'a pas pu être publiée avant la naissance de cette

2. Disponible sur demande.

personne. Le score de cette portion pour toutes les années précédant l'année de naissance de la personne a été fixé à 0,3. Le score augmente au fur et à mesure jusqu'à atteindre 1 vingt ans après l'année de naissance de la personne. Si plusieurs personnes sont citées dans un même extrait, les scores pour chaque personne et pour chaque année sont multipliés entre eux.

4.2 Néologismes et archaïsmes

Les néologismes correspondent à des mots nouvellement créés, tandis que les archaïsmes sont des mots qui ont cessé d'être utilisés à un certain moment et qui ne sont donc plus d'usage. Les deux phénomènes constituent des indices utiles pour déterminer la date de publication d'un document : s'il contient un néologisme, une probabilité très faible peut être attribuée aux années précédant la date d'apparition du néologisme, l'inverse étant vrai pour les archaïsmes. Cependant, les dates d'apparition et de disparition des mots ne sont pas des informations aisément accessibles. Nous avons donc développé une méthode pour extraire automatiquement des néologismes et des archaïsmes à partir des données de Google Books pour le français [3].

4.2.1 Acquisition automatique de néologismes et d'archaïsmes

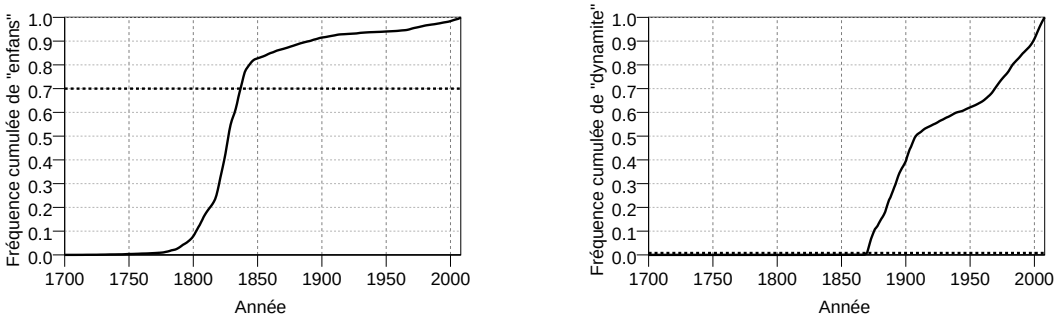
L'acquisition des dates d'apparition ou de disparition des mots n'est pas une tâche triviale. En effet, les métadonnées associées à Google Books ne sont pas toujours précises (Nunberg, 2009). Il n'est donc pas possible d'utiliser un critère simple comme l'extraction de la première année d'occurrence du mot pour identifier les néologismes.

Nous avons donc appliqué la méthode suivante, qui se fonde sur la distribution des fréquences cumulées :

1. recueil de la distribution des occurrences du mot entre les années 1700 et 2008 [4] ;
2. lissage de la distribution avec une fenêtre de 3 années ;
3. calcul de la distribution des fréquences cumulées et extraction de la date d'apparition/disparition, correspondant à l'année à laquelle la fréquence cumulée dépasse un certain seuil.

Nous avons défini les meilleurs seuils en utilisant 32 néologismes (par exemple «photographie» ou «télévision») et 21 archaïsmes (anciennes orthographes, voir section 4.3). Les seuils ainsi obtenus sont 0,008 pour les néologismes et 0,7 pour les archaïsmes. De plus, nous n'avons conservé que les néologismes ayant un nombre d'occurrences moyen par année supérieur à 10 et les archaïsmes ayant un nombre d'occurrences moyen supérieur à 5 pour les années considérées. Nous avons ainsi extrait 34 396 néologismes et 53 392 archaïsmes avec leur date d'apparition/disparition.

La figure 4 présente deux courbes de fréquences cumulées : l'une pour un archaïsme (l'orthographe ancienne «enfans»), et l'autre pour le néologisme «dynamite» (inventée en 1867).



(a) Archaïsme

(b) Néologisme

FIGURE 4: Fréquences cumulées

3. Disponibles à l'adresse suivante : `http://ngrams.googlelabs.com/datasets`

4. La première année disponible dans les Google Books n-grams est en réalité 1536, mais étant données les années qui nous intéressent ici, nous avons considéré que 1700 était un seuil adéquat. En outre, les données pour les 16e et 17e siècle sont trop peu nombreuses pour constituer des données fiables pour ce type de traitement.

Le seuil correspond aux lignes horizontales en pointillés. Ces courbes ont des profils très différents : les archaïsmes sont caractérisés par une fonction logistique, qui atteint un plateau bien avant la fin de la période considérée ; les néologismes correspondent quant à eux à des courbes exponentielles.

Nous avons calculé le taux d'erreur sur le corpus d'entraînement : pour 90% des archaïsmes trouvés dans le corpus, la date de la portion est bien antérieure à la date de disparition, et, si l'on suppose que le mot peut encore être utilisé jusqu'à 20 ans après la date de disparition calculée, la date de la portion est antérieure à la date de disparition plus 20 ans pour 97% des archaïsmes. Pour les néologismes, la date de la portion est postérieure à la date d'apparition dans 97% des cas, et à la date d'apparition moins 20 ans dans 99,8% des cas.

4.2.2 Score attribué par les néologismes et archaïsmes

Les listes de néologismes et archaïsmes établies ont été utilisées pour attribuer un score à chaque année pour chaque portion. Pour les néologismes, un score élevé a été attribué aux années postérieures à la date d'apparition, et un score faible pour les années la précédant. La formule utilisée pour les néologismes est la suivante, avec p la portion de texte, w un mot, y une année dans la période considérée 1801-1944 et $année(w)$ la date d'apparition extraite pour un néologisme :

$$score_{néo}(p,y) = \frac{\sum_{w \in t} score_{néo}(w,y)}{|p|} \text{ avec :}$$

$$score_{néo}(w,y) = \begin{cases} 1 & \text{si } w \notin \text{néologismes} \\ 1 & \text{si } w \in \text{néologismes et } y \geq année(w) \\ 0,2 & \text{si } w \in \text{néologismes et } (année(w) - y) > 20 \\ 0,2 + 0,04 \cdot (20 + y - année(w)) & \text{sinon} \end{cases}$$

Une formule équivalente est utilisée pour les archaïsmes : dans ce cas les années suivant la date de disparition d'un mot ont un score faible.

4.3 Réformes orthographiques

Pendant la période 1801-1944, le français a connu deux réformes orthographiques majeures : la première en 1835 et la seconde en 1878. Le principal changement introduit par la première concerne certaines conjugaisons finissant par «oi», qui deviennent «ai» (par exemple pour le verbe «avoir», «avois» devient «avais»). La seconde réforme concerne principalement les noms finissant par «ant» ou «ent» dont le pluriel devient «ants»/«ents» au lieu de «ans»/«ens» (par exemple, «enfans» devient «enfants»).

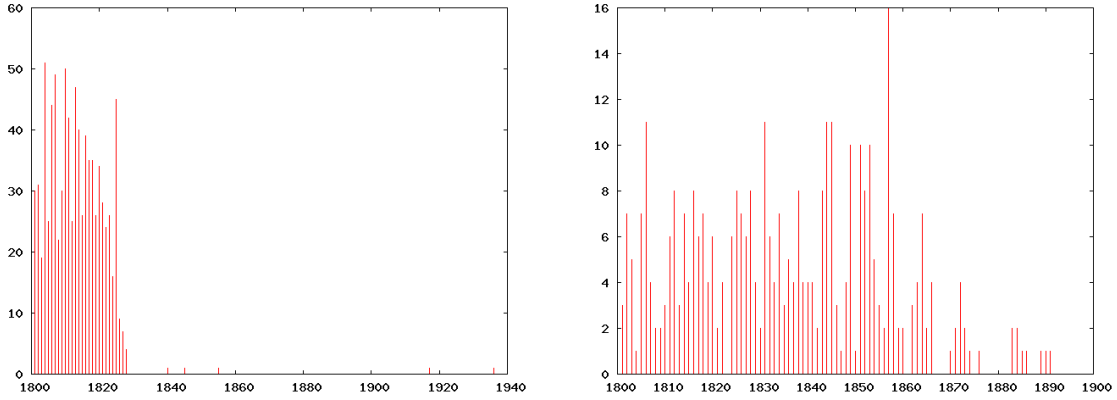
À la suite de (Albert *et al.*, 2010), nous avons utilisé ces informations. La figure 5 montre la distribution de chaque type de mots dans le corpus d'entraînement TRN pour chaque année. Les mots ayant été modifiés par la première réforme sont bien présents principalement avant 1828, et ceux modifiés par la deuxième réforme sont présents uniquement avant 1891.

4.3.1 Scores attribués grâce aux réformes orthographiques

Nous avons utilisé les informations fournies par les réformes de la même façon que (Albert *et al.*, 2010) : nous avons attribué un score à chaque année pour chaque portion de texte. Afin de détecter les anciennes orthographes dans nos textes, nous avons utilisé la méthode suivante :

- recueil des mots inconnus avec hunspell [5]. Puis pour chaque mot inconnu :
- si le mot finit par «ois/oit/oient», remplacer «o» par «a» ;
  - si le mot ainsi formé est dans le dictionnaire, incrémenter le compteur n_{28}, qui correspond au nombre de mots dont l'orthographe a été modifiée par la première réforme ;
- si le mot finit par «ans/ens», insérer «t» avant «s» ;
  - si le nouveau mot est dans le dictionnaire, incrémenter le compteur n_{91}, qui correspond au nombre de mots dont l'orthographe a été modifiée par la deuxième réforme ;

5. Hunspell a été utilisé avec le dictionnaire DELA pour le français (Blandine & Silberzstein, 1990)



(a) Mots terminant par «oi»

(b) Mots terminant par «a/ens»

FIGURE 5: Distribution des mots modifiés par les réformes orthographiques dans le corpus d'entraînement (TRN)

Une fonction permet de déterminer un score pour chaque année y, à partir des compteurs n_{28} et n_{91} selon les formules suivantes :

$score_{ortho}(p, y) = score_{28}(p, y) \cdot score_{91}(p, y)$ avec :

$$score_r(p,y) = \begin{cases} f_r(p,y) & si\ y > r \\ 1 & si\ y \leq r \end{cases} \quad , \quad f_{28}(p,y) = \begin{cases} 1 & si\ n_{28}(p) = 0 \\ 0,15 & si\ n_{28}(p) = 1 \\ 0 & si\ n_{28}(p) > 1 \end{cases} \quad et \quad f_{91}(y) = \begin{cases} 1 & if\ n_{91}(p) = 0 \\ 0 & if\ n_{91}(p) > 0 \end{cases}$$

Ainsi, si $n_{28} = 1$ et $n_{91} = 1$ pour une portion de texte, le score pour les années antérieurs à 1828 est de 1, puis de 0,15 pour les années comprises entre 1828 et 1892, ce qui correspond au taux d'erreur pour ce critère dans notre corpus d'entraînement, et de 0 pour les années postérieures à 1891, puisque la présence d'une orthographe modifiée par la deuxième réforme est un indicateur très fiable d'une date de publication antérieure à 1891.

5 Méthodes de similarité temporelle

Les méthodes de similarité temporelle calculent des similarités entre chaque portion de texte et un corpus de référence. Elles permettent de dater précisément une portion, mais sont sujettes aux erreurs, qui devraient être en partie corrigées par les méthodes chronologiques. Notre intuition est que les informations apportées par les deux types de méthodes sont complémentaires.

5.1 Similarité cosinus

5.1.1 Corpus de documents comme référence

Le corpus d'entraînement fournit des exemples de textes pour chaque année dans la période 1801-1944. Ces textes peuvent être utilisés comme référence pour obtenir des statistiques temporelles. Nous avons donc regroupé toutes les portions d'une même année du corpus d'entraînement et utilisé ces groupes de portions comme référence pour les années correspondantes. Chaque groupe et chaque portion testée ont été indexés par le tf·idf, selon la formule suivante pour le mot i et le document j :

$$tf \cdot idf_{i,j} = \frac{n_{i,j}}{\sum_k n_{k,j}} \cdot \log \frac{|Y|}{|\{y_j : w_i \in y_j + smoothing\}|}$$

avec $|Y|$ le nombre d'années dans le corpus d'entraînement, y_j le groupe de portions de texte pour l'année j, $n_{i,j}$ le nombre d'occurrences d'un mot w_i dans le document j. $smoothing$ est un lissage appliqué pour tenir compte

des mots du corpus d'évaluation qui n'étaient pas dans le corpus d'entraînement.

Pour une portion de texte du corpus d'évaluation, nous avons calculé les similarités entre la portion et chaque groupe représentant une année grâce à une mesure cosinus. Nous avons essayé de calculer cette similarité pour des n-grams de mots, n allant de 1 à 5 ; cependant, pour n>2, la taille du corpus d'entraînement ne permet pas d'obtenir de bons résultats. Nous avons utilisé pour ces n-grams la version lemmatisée des corpus.

Puisque le corpus est composé de documents numérisés automatiquement, de nombreuses erreurs sont présentes dans les textes, ce qui pose problème pour le tf.idf : les tf et df sont plus petits que ce qu'ils devraient être pour des mots réels, puisque des erreurs de numérisation empêchent l'identification de certaines occurrences, tandis que des mots erronés ont des idf élevés, puisque nécessairement peu présents dans le corpus. Afin de prendre en compte cette particularité, nous avons également calculé la similarité en utilisant des n-grams de caractères (comme par exemple (Naji *et al.*, 2011) pour la recherche d'information dans des corpus numérisés). Ainsi, par exemple, pour le texte «sympathie1» qui contient une erreur de numérisation, les n-grams de caractères (pour n<9) correspondront en grande partie à ceux du mot «sympathie» malgré l'erreur de numérisation.

5.1.2 Google n-grams comme référence

Le corpus d'entraînement étant relativement petit, nous avons également tenté d'utiliser les Google n-grams comme données d'entraînement. Pour des raisons de temps de calcul, nous avons utilisé uniquement les n-grams ayant un contenu alphanumérique et ayant plus de 10 occurrences pour une année donnée. Les données ainsi créées ont été utilisées à la place du corpus d'entraînement. La formule du tf.idf est alors légèrement modifiée pour le corpus d'entraînement, puisque $n_{i,j}$ devient le nombre d'occurrences du n-gram w_i pour l'année j et y_j correspond aux données des Google n-gram pour l'année j.

5.2 Système SVM

Les SVM sont des algorithmes d'apprentissage automatique largement utilisés en TAL, appartenant à la classe des classifieurs à marge maximale (Vapnik, 1998). Nous avons utilisé l'implémentation `svm-light` [6] (Joachims, 1999). Nous avons utilisé deux fonctions noyaux dans les SVM : la `fonction à base radiale` et le `noyau polynomial` toutes deux disponibles dans le logiciel `svm-light`. Étant donnée la faible quantité de données disponibles pour chaque année (25 portions par année, sauf pour 1815, pour laquelle nous disposons de 21 portions), l'approche d'entraînement `un-contre-tous` a été utilisée, et non pas `un-contre-un`. Le système SVM est ainsi composé de 144 modèles binaires, un pour chaque année entre 1801 et 1944. Dans chaque modèle, les instances positives sont celles extraites des portions appartenant à l'année à détecter, et les instances négatives sont les autres. Chaque modèle reconnaît ainsi les portions appartenant à l'année qu'il décrit. Au moment de la classification, chaque portion est évaluée par les 144 modèles et celui donnant le meilleur score est sélectionné.

Les paramètres et jeux d'attributs ont été réglés sur les corpus d'entraînement (TRN) et de validation (DEV) décrits plus hauts. Nous n'avons pas paramétré tous les paramètres ni tous les types d'attributs, ce qui aurait requis un trop grand nombre d'expériences, mais utilisé notre expérience pour en régler certains. Pour les autres nous avons utilisé les valeurs par défaut. Le paramètre de compromis C a été fixé à 1. Dans la plupart des tâches, sa valeur optimale est entre 1 et 10, 1 donnant toujours de bons résultats. Le paramètre de coût, qui affecte le poids des erreurs faites sur les instances positives et négatives, a été fixé au ratio entre le nombre d'instances négatives et positives, comme suggéré dans (Morik *et al.*, 1999). En ce qui concerne les fonctions noyau, nous avons testé les `fonctions à base radiale` et le `noyau polynomial`. Cette dernière était plus efficace sur l'ensemble DEV et a donc été conservée. Les valeurs par défaut ont été utilisées pour les paramètres du noyau polynomial (1 pour c et 3 pour le degré polynomial d).

Pour les attributs, nous avons effectué plusieurs expériences et conservé le jeu donnant le meilleur résultat sur DEV. Nous avons tout d'abord essayé les configurations courantes dans des tâches de classification de textes. Par exemple, nous avons supprimé les mots vides et remplacé les mots par leurs lemmes. Cette configuration dégradait les performances. En revanche, en gardant les mots vides et en utilisant à la fois les mots et leurs lemmes, nous obtenions de meilleurs résultats qu'avec les mots seuls. Cette configuration a donc été choisie comme système

6. Disponible à l'addresse `http://svmlight.joachims.org/`

SVM de base. Nous avons également testé l'utilisation de n-grams pour n allant de 1 à 4 et les 2-grams ont donné les meilleurs résultats.

À partir de cette configuration, nous avons intégré les informations founies par les systèmes décrits dans la section 4, c'est-à-dire les années de naissance des personnes, les néologismes et les archaïsmes, et les réformes orthographiques. Chacun de ces systèmes a fourni des informations pour le SVM sous la forme de vecteurs $attribut : année$, où $attribut$ est un nom de personne dans le cas des dates de naissance, un néologisme ou un archaïsme ou un mot qui a été affecté par l'une des réformes orthographiques. Cette forme a posé problème pour le SVM, même en représentant les années dans l'intervalle 1..144 au lieu de 1801..1944 : les performances étaient moins bonnes en utilisant cette représentation. Nous avons donc changé le mode de représentation, et représenté l'information fournie par les méthodes chronologiques par une information binaire (0 si l'attribut est absent, 1 si il est présent) : un attribut code l'information elle-même, c'est-à-dire par exemple la présence d'un néologisme particulier, et un autre attribut code l'année associée à cette information. Cette représentation a permis d'améliorer nos résultats.

Puisque dans nos expériences préliminaires le comportement des systèmes était identique sur les portions de 500 mots et sur les portions de 300 mots, nous avons mené nos expériences uniquement sur celles de 300, et avons appliqué la meilleure configuration observée sur celles de 300 pour les portions de 500.

6 Combinaison des scores

Nous avons ensuite effectué une combinaison des scores définis par les méthodes décrites précédemment. Les différentes méthodes fournissent en effet des informations complémentaires : par exemple, les archaïsmes indiquent une limite haute à la date de publication, alors que la similarité cosinus va donner des années probables de publication. Pour la combinaison des scores, nous avons utilisé deux stratégies : la multiplication de tous les scores et la régression linéaire.

6.1 Multiplication des scores

La méthode la plus simple de combinaison était la multiplication des scores, qui sont tous entre 0 et 1, 0 indiquant une probabilité nulle pour une année donnée, et 1 indiquant une probabilité maximale. Le score final est donc la multiplication des scores précédents :

$$score_{multiplication}(p, y) = \prod_k score_k(p, y)$$

avec $score_k(p, y)$ le score du système k pour la portion p et l'année y.

6.2 Régression linéaire

Dans ce cas, les scores des différents systèmes ne sont pas multipliés mais additionnés avec des coefficients de pondération, selon la formule suivante :

$$score_{régression}(p, y) = \sum_k \alpha_k \cdot score_k(p, y) + \varepsilon$$

avec α_k coefficient pour le système k, $score_k(p, y)$ le score donné par le système k à la portion p pour l'année y et ε le terme d'erreur.

Les coefficients ont été calculés sur le corpus d'entraînement en utilisant la fonction R `lm()`. Le processus de régression linéaire trouve le meilleur modèle (déterminé par les coefficients α pour prédire une valeur numérique à partir de plusieurs indices ici, les scores des systèmes). Dans notre cas, la valeur numérique à prédire dépend de la distance $dist$ entre une année et l'année réelle de publication de la portion : la valeur est $1 - dist/143$.

Dans la phase de développement, nous avons réglé les valeurs de α et ε sur le corpus d'entraînement TRN et testé la combinaison sur le corpus DEV. Comme le cosinus et les SVM avaient besoin d'une phase d'entraînement,

nous n'avons pas inclus les scores de ces systèmes dans notre modèle de régression. Nous avons donc calculé un score de régression à partir des scores donnés par les néologismes, archaïsmes, années de naissance et réformes orthographiques. Les scores du cosinus et des SVM ont ensuite été multipliés par ce score de régression. Pour la phrase d'évaluation, nous avons réglé les paramètres sur le corpus de développement complet.

6.3 Pondération

La régression linéaire permet de pondérer les scores obtenus par les méthodes chronologiques (néologismes, archaïsmes, dates de naissance et réformes orthographiques) mais ne permet pas de pondérer les résultats fournis par les méthodes de similarité temporelle (similarité cosinus et SVM). Nous avons donc utilisé une pondération basée sur la formule suivante :

$$score_{ponderé}(p, y) = \beta \cdot score_{régression}(p, y) + (1 - \beta) \cdot score_{cosinus} \cdot score_{SVM}$$

avec β un coefficient, déterminé à partir du corpus d'apprentissage et donnant alors les meilleurs résultats, de 0,015.

7 Résultats

7.1 Score

Nous avons évalué nos systèmes en utilisant le score proposé par DEFT, qui prend en compte la distance entre l'année prédite et l'année réelle de publication.

Étant donnée une portion de texte a_i dont l'année de publication de référence est $d_r\left(a_i\right)$, un système donne une estimation de l'année $d_p\left(a_i\right)$. Le système reçoit alors un score qui dépend de la distance entre l'année prédite et l'année de référence. Le score est basé sur une fonction gaussienne et est moyenné sur les N portions de test. La formule précise est la suivante :

$$S = \frac{1}{N} \sum_{i=1}^{N} e^{-\frac{\pi}{10^2}\left(d_p(a_i) - d_r(a_i)\right)^2} \tag{1}$$

Nous présenterons d'abord les résultats du cosinus et des SVM, puis deux des combinaisons. Les systèmes utilisés pour les données d'évaluation ont été entraînés sur le corpus de développement complet (TRN + DEV).

7.2 Résultats pour les méthodes de similarité temporelle

7.2.1 Similarité cosinus

Les résultats de la similarité cosinus sont présentés dans les tableaux 1 et 2 (seuls les meilleurs systèmes sont présentés). Nous pouvons voir que les meilleurs résultats sont obtenus en utilisant les n-grams de 5 caractères, ce qui était attendu du fait du bruit dans nos données. Les 1-grams de mots sont meilleurs sur les portions de 300 mots que les 2-grams, mais c'est le contraire sur les portions de 500 mots, ce qui peut s'expliquer par le fait que les 2-grams sont meilleurs avec plus de données d'entraînement.

Pour le cosinus utilisant les Google n-grams, le corpus a été utilisé dans sa version non lemmatisée, puisque les Google n-grams contiennent des mots fléchis. Les meilleurs résultats sont obtenus avec les 2-grams, mais sont inférieurs à ceux obtenus avec le corpus d'entraînement. Ceci est étonnant car les Google n-grams représentent un corpus bien plus grand, mais peut peut-être s'expliquer par la différence de nature des documents, puisque notre corpus est composé uniquement d'extraits de journaux. En outre, la datation des Google Books n'est pas toujours fiable (Nunberg, 2009).

	Corpus d'entraînement (DEV)		Corpus d'évaluation	
	300 mots	**500 mots**	**300 mots**	**500 mots**
1-grams de mots	0,260	0,299	0,267	0,321
2-grams de mots	0,209	0,319	0,263	0,327
5-grams de caractères	0,287	0,327	0,311	0,363

TABLE 1: Résultats obtenus par la méthode fondée sur le cosinus

	Corpus d'entraînement (DEV)		Corpus d'évaluation	
	300 mots	**500 mots**	**300 mots**	**500 mots**
1-grams de mots	0,210	0,221	0,200	0,216
2-grams de mots	0,238	0,295	0,241	0,264

TABLE 2: Résultats obtenus par la méthode fondée sur le cosinus avec les Google n-grams

7.2.2 Système SVM

Les résultats obtenus avec le système fondé sur les SVM sont donnés dans les tableaux 3 et 4. Comme nous pouvons le voir dans le tableau 3, l'ajout incrémental d'attributs encodant l'information donnée par les méthodes chronologiques améliore les résultats.

	Corpus d'entraînement (DEV)
	300 mots
Baseline (2-grams mots + lemmes)	0,228
+néologismes	0,234
+réformes orthographiques	0,242
+années de naissance	0,243

TABLE 3: Résultats additifs du système SVM sur les portions de 300 mots avec différents types d'attributs

7.2.3 Combinaison des scores

Le tableau 5 présente les résultats obtenus sur les corpus d'entraînement et de test pour les meilleurs systèmes.

La combinaison des systèmes améliore nettement les scores des systèmes individuels. Les résultats sur les portions de 500 mots sont meilleurs que ceux sur les portions de 300 mots, ce qui était attendu puisque les portions de 500 mots présentent plus d'indices de leur date de publication. Globalement, la combinaison par multiplication obtient de meilleurs scores que la régression linéaire. La figure 6 montre les résultats en termes d'année et de décennie correcte au premier rang. Nos systèmes obtiennent environ 35% de décennie correcte au premier rang pour les portions de 300 mots, et 40% pour celles de 500 mots. Pour les années, la régression linéaire détecte la bonne année pour 10% des portions de 300 mots et 14% des portions de 500 mots. Ces résultats sont bien au-dessus du hasard (7% pour les décennies et 0,7% pour les années), et sont meilleurs pour les décennies que ceux du challenge DEFT 2010 (Grouin *et al.*, 2010).

7.2.4 Soumissions à DEFT 2011

Nous avons soumis trois résultats lors de la campagne DEFT 2011. Concernant les portions de 300 mots, nous avons soumis les résultats provenant de la combinaison par multiplication, par régression linéaire et par pondération, systèmes ayant donné les meilleurs résultats sur le corpus d'entraînement. Concernant les portions de 500 mots, nous avons, lors des 3 jours de tests, eu une difficulté d'adaptation du système de combinaison par pondération et n'avons donc pas soumis les résultats de ce système. Nous avons ainsi soumis, pour les portions de

Corpus d'entraînement (DEV)		Corpus d'évaluation	
300 mots	**500 mots**	**300 mots**	**500 mots**
0,243	0,293	0,272	0,330

TABLE 4: Résultats du système SVM

	Entraînement (DEV)		Évaluation	
	300 mots	**500 mots**	**300 mots**	**500 mots**
multiplication	0,343	0,401	0,378	0,452
régression	0,356	0,390	0,374	0,428
pondération	0,319	0,425	0,358	0,384

TABLE 5: Résultats obtenus avec la combinaison des scores

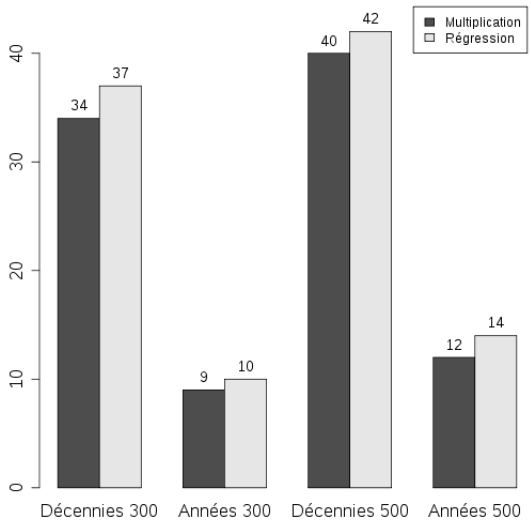


FIGURE 6: Pourcentage de décennies et d'années correctes au premier rang

500 mots, les résultats provenant de la combinaison par multiplication, par régression linéaire ainsi que le résultat fournis par la similarité cosinus calculée sur des 5-grams de caractères.

8 Conclusion

Nous avons présenté une approche pour la datation automatique de documents historiques. Celle-ci est fondée sur différentes méthodes et cherche à tirer avantage de chacune d'elles afin d'estimer au mieux l'année de publication d'extraits de journaux. Notre meilleur système a obtenu une F-mesure de 0,452 sur des portions de 500 mots et 0,378 sur des portions de 300 mots. Les résultats montrent l'importance d'utiliser des données externes qui prennent en compte l'évolution diachronique d'une langue associées à des techniques de fouille de texte par apprentissage.

Cette tâche reste un défi puisque notre meilleur système permet d'estimer correctement pour des portions de 500 mots *seulement* 40% de décennies et 14% d'années. La difficulté d'une telle tâche vient d'une part de la qualité des documents, puisqu'ils sont numérisés (mais ceci fait aussi de la tâche une application proche d'un besoin réel), d'autre part de la faible quantité de données de référence à notre disposition.

Cette tâche nous a permis de confronter des techniques *classiques* utilisées en traitement automatique des langues à des données bruitées par une numérisation (qui rendent très difficile la tâche d'un détecteur d'entités nommées habituellement performant, par exemple) et ainsi de développer des techniques adaptées comme l'utilisation de n-grams de caractères. Il serait bien sûr intéressant de préalablement corriger orthographiquement les corpus.

Remerciements

Ce travail a été partiellement financé par le projet Quæro (financement Oseo, agence française pour l'innovation et la recherche) et le projet DOXA du pôle de compétitivité CAP-DIGITAL.

Références

ALBERT P., BADIN F., DELORME M., DEVOS N., PAPAZOGLOU S. & SIMARD J. (2010). Décennie d'un article de journal par analyse statistique et lexicale. In *Atelier DEFT, Actes TALN 2010*.

BLANDINE C. & SILBERZSTEIN M. (1990). Dictionnaires électroniques du français. *Langue française*, **87**.

GALIBERT O. (2009). *Approches et méthodologies pour la réponse automatique à des questions adaptées à un cadre interactif en domaine ouvert*. PhD thesis, Thèse de doctorat en informatique, Université Paris-Sud 11, Orsay, France.

GROUIN C., FOREST D., PAROUBEK P. & ZWEIGENBAUM P. (2011). Présentation et résultats du défi fouille de texte DEFT2011. In *Atelier DEFT, Actes TALN 2011*.

GROUIN C., FOREST D., SYLVA L. D., PAROUBEK P. & ZWEIGENBAUM P. (2010). Présentation et résultats du défi fouille de texte DEFT2010 : Où et quand un article de presse a-t-il été écrit ? In *Atelier DEFT, Actes TALN 2010*.

JOACHIMS T. (1999). Making large-scale SVM learning practical. In B. SCHÖLKOPF, C. BURGES & A. SMOLA, Eds., *Advances in Kernel Methods - Support Vector Learning* : MIT Press, Cambridge, MA, USA.

MORIK K., BROCKHAUSEN P. & JOACHIMS T. (1999). Combining statistical learning with a knowledge-based approach - a case study in intensive care monitoring. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML'99, p. 268–277.

NAJI N., SAVOY J. & DOLAMIC L. (2011). Recherche d'information dans un corpus bruité (OCR). In *CORIA 2011*.

NUNBERG G. (2009). Google's Book Search : A Disaster for Scholars.

ROSSET S., GALIBERT O., BERNARD G., BILINSKI E. & ADDA G. (2008). The LIMSI participation to the QAst track. In *Working Notes of CLEF 2008 Workshop*.

SCHMID H. (1994). Probabilistic part-of-speech tagging using decision trees. In *International Conference on New Methods in Language Processing*, p. 44–49.

VAPNIK V. N. (1998). *Statistical Learning Theory*. John Wiley and Sons.