UFR Sciences et Techniques
Département d'Informatique

Segmentation discursive des courriels selon une approche supervisée "paresseuse"

Soufian Salim

encadré par Nicolas Hernandez et tutoré par Christian Viard-Gaudin

Soumis dans le cadre de l'obtention du diplôme
Master en Informatique
Juillet 2014

Résumé

Dans le cadre de l'analyse de discussions asynchrones en ligne, multi-modales et multi-domaines, nous proposons une stratégie novatrice pour l'identification de segments d'actes de langage. Le processus décrit vise à soutenir l'analyse de messages en termes d'intention communicative. Notre objectif est de développer un système d'étiquetage de séquences permettant de détecter les frontières entre segments. L'originalité de l'approche proposée vient du fait que nous exploitons les efforts cognitifs effectués par des humains pour la tache de mise en forme de messages de réponse pour éviter d'avoir à effectuer un laborieux travail d'annotation manuelle. Nous décrivons notre approche, proposons un nouveau corpus de courriers électroniques et rapportons l'évaluation des modèles de segmentation ainsi construits.

Abstract

In the context of multi-domain and multimodal online asynchronous discussion analysis, we propose an innovative strategy for the annotation of speech act (SA) segments. The process aims at supporting the analysis of messages in terms of SA. Our objective is to train a sequence labelling system to detect the segment boundaries. The originality of the proposed approach is to avoid manually annotating the training data and instead exploit the human computational efforts dedicated to message reply formatting when the writer replies to a message by inserting his response just after the quoted text appropriate to his intervention. We describe the approach, propose a new electronic mail corpus and report the evaluation of segmentation models we built.

Remerciements

Je voudrais remercier :

– Nicolas Hernandez (LINA)

– Christian Viard-Gaudin (IRCCyN)

– Nathalie Camelin (LIUM)

– Les membres de l'équipe TALN au LINA

– Les enseignants et étudiants du master ATAL

Table des matières

Liste des tableaux

Table des figures

Chapitre 1

Introduction

Dans ce chapitre, nous présentons tout d'abord le contexte dans lequel ce travail a été effectué. Nous exposons ensuite la motivation et les objectifs du travail, avant d'en détailler les contributions.

1.1 Contexte

Ce travail a été effectué dans le cadre d'un stage obligatoire de cinq mois nécessaire à l'obtention du diplôme de Master en Informatique, spécialité ATAL [1], à l'Université de Nantes. Le stage a été effectué au LINA [2], au sein de l'équipe TALN [3]. L'encadrement a été assuré par Nicolas Hernandez et le tutorat par Christian Viard-Gaudin.

Ce stage fait également office de préambule à une thèse potentielle dans le domaine de la communication médiée par les machines (CMC), qui s'inscrirait dans le cadre du projet ODISAE (*Optimizing Digital Interaction with a Social and Automated Environment*) pour lequel le LINA a été sélectionné comme laboratoire partenaire, et auquel participent les entreprises suivantes : EPTICA (coordinateur), Jamespot, Kwaga, Cantoche, TokyWoky, Aproged et CDT10.

1. Apprentissage et Traitement Automatique de la Langue (`http://www.dpt-info.univ-nantes.fr/1326208903095/0/fiche___pagelibre/`)
2. Laboratoire Informatique de Nantes Atlantique (`http://www.lina.univ-nantes.fr/`)
3. Traitement Automatique du Langage Naturel (`http://www.lina.univ-nantes.fr/?-TALN-.html`)

1.2 Motivation et objectifs

Le traitement automatique de conversations asynchrones en ligne (comme les fils de discussions de forums, ou les conversations par courriels) est d'une importance capitale pour les communautés qui cherchent à améliorer les systèmes de question-réponse, à analyser les opinions et les intentions des utilisateurs, à détecter les messages contenant des requêtes urgentes, à identifier les problèmes non-résolus, etc. En particulier, l'analyse des conversations portant des demandes d'information (e.g. dépannage et assistance) constitue un enjeu scientifique et industriel particulièrement important.

C'est pour améliorer tous ces systèmes que nous cherchons à segmenter rhétoriquement (c'est à dire en fonction de l'*intention discursive* du locuteur ; le moyen qu'il met en œuvre pour agir sur son environnement et ses interlocuteurs par ses mots) les messages de discussions en ligne, et plus particulièrement les courriels. Nous supposons qu'en segmentant un courriel en fragments structurellement autonomes et rhétoriquement homogènes, et donc potentiellement plus pertinents que la phrase ou le paragraphe, son analyse sera facilitée.

Dans le cadre de ce travail, nous nous intéressons plus particulièrement à la segmentation de courriels de langue anglaise tirés de listes de diffusions [4] réservées à l'assistance aux utilisateurs.

1.3 Contributions

Ce travail développe une technique permettant d'exploiter les efforts cognitifs effectués par des humains attelés à une tâche de mise en forme de messages de réponse pour entraîner un segmenteur discursif capable d'identifier des fragments de messages autonomes et homogènes.

Nous proposons également un système de segmentation visant à soutenir l'analyse de messages en termes d'actes du langage, et rapportons l'évaluation de différents modèles construits à partir

4. Une liste de diffusion, ou liste de distribution (*mailing list* en anglais), est un système permettant d'envoyer à une adresse unique un message qui sera ensuite distribué à tous les abonnés de la liste. Les listes de diffusion peuvent être utilisées de manière unilatérale (comme pour l'envoi de *newsletters* par exemple), ou autoriser les abonnés à envoyer des messages (on parle alors parfois de listes de discussion).

de plusieurs ensembles de traits.

De plus, nous proposons à la communauté un nouveau corpus pour l'analyse de discussions asynchrones en ligne, qui a l'avantage d'être vaste, moderne, multimodal et multilingue.

Chapitre 2

Concepts et étude bibliographique

Dans ce chapitre, nous décrivons certains travaux existants que l'on peut rapprocher au notre et introduisons certains concepts utiles à notre approche. Tout d'abord, nous abordons le concept d'acte du langage. Cette notion nous sera utile puisque comme le verrons plus tard, une propriété majeure des segments que nous cherchons à isoler est d'être porteur de son propre acte du langage. Ensuite, nous nous intéresserons aux travaux portant sur la segmentation de texte, avant de considérer différents corpus que nous pourrions utiliser dans le cadre de nos expériences. Enfin, nous listons différentes approches collaboratives existantes pour l'obtention de corpus annotés, et indiquons en quoi elles diffèrent de la notre.

2.1 Actes du langage

La théorie des actes du langage [Austin, 1975] propose de décrire les énonciations en termes des fonctions communicatives portées par chacun d'eux (e.g. question, réponse, remerciement...). Ainsi, dans la plupart des travaux, c'est en termes d'actes du langage [1] que les interactions entre participants d'une conversation sont modélisées. Austin considère les énonciations comme des actions effectuées par le locuteur ; on trouve ici l'idée selon laquelle tout acte d'énonciation

1. Aussi connu dans certains travaux sous le nom d'actes du dialogue (*dialog act*), ou d'actes du discours (*speech act*).

Acte	Description ou exemples	Référence
Verdictif	acquitter, condamner, décréter...	[Austin, 1975]
Exercitif	dégrader, commander, ordonner, pardonner, léguer...	
Promissif	promettre, faire vœu de, garantir, parier, jurer de...	
Comportatif	s'excuser, remercier, déplorer, critiquer...	
Expositif	affirmer, nier, postuler, remarquer...	
Assertif	affirmation d'un état de fait	[Searle, 1976]
Directif	tentative de pousser un interlocuteur à faire quelque chose	
Promissif	engagement de la part du locuteur	
Expressif	expression d'un état psychologique	
Déclaratif	déclaration ayant un impact direct	

TABLE 2.1: Taxonomies fondatrices pour la catégorisation des actes du langage.

serait la réalisation d'un acte social. Les verbes qui spécifient ces actions sont appelés *verbes performatifs*, comme quand on dit "Je vous confère le titre de capitaine". Mais les actes du langage ne sont pas constitués uniquement de ces types de verbes. [Searle, 1976] propose cinq classes d'actes du dialogue : les assertifs (assertion, affirmation, etc.), les directifs (ordre, demande, conseil, etc.), les promissifs (promesse, offre, invitation, etc.), les expressifs (félicitation, remerciement, etc.) et les déclaratifs (déclaration de guerre, nomination, baptême, etc.).

Les travaux existants concernant les actes du langage s'intéressent dans leur large majorité à classifier les énoncés selon telle ou telle taxonomie, dont il existe un très grand nombre [Traum, 2000]. Le tableau 2.1 détaille deux taxonomies fondatrices : celle de Austin et celle de Searle. Le tableau 2.2 présente quelques taxonomies récemment employées dans le cadre de l'analyse de courriels. L'usage d'algorithmes de classification supervisée [Tavafi *et al.* , 2013] représente l'approche dominante pour déterminer l'acte porté par une phrase ou un message.

2.2 Segmentation de messages

Jusqu'à présent, assez peu de travaux adressent le problème de la segmentation de courrier électronique. [Lampert *et al.* , 2009a] propose de segmenter les courriels en zones prototypiques telles que la contribution de l'auteur, les citations de messages originaux, la signature, ou encore la formule d'ouverture ou de fermeture. Pour ce faire, il utilise un système basé sur les SVM

Acte	Corpus ou type de corpus	Référence
Divulgation Édification Conseil Confirmation Question Reconnaissance Interprétation Réflexion	multi-domaines	[Lampert *et al.* , 2006]
Question-requête Question ouverte Engagement à la 1ère personne Expression à la 1ère personne Autres énoncés à la 1ère personne Autres	messagerie d'entreprise	[De Felice *et al.* , 2013]
Acceptation Reconnaissance/appréciation Motivateur d'action Mécanisme de politesse Question rhétorique Question ouverte Question à choix multiple Question en "wh*" Question binaire Rejet de réponse Affirmation Réponse incertaine	BC3	[Ulrich *et al.* , 2008a]

TABLE 2.2: Exemples de taxonomies des actes du langage spécifiques à l'analyse de courriels.

(machines à vecteurs de support, ou *Support Vector Machines*[2]) et atteint un précision de 87% pour une segmentation en neuf zones. Notre travail contraste en ce que nous nous concentrons sur la segmentation de la contribution de l'auteur (ce que nous appelons le "nouveau contenu").

[Joty *et al.* , 2013] identifie des groupes de phrases thématiquement proches au travers de multiple messages d'un fil de discussion, sans distinguer courriels et messages de forums. Notre problème diffère, d'une part parce que nous cherchons en premier lieu à effectuer une segmentation rhétorique et non thématique, et en second lieu en ce que nous ne nous intéressons qu'à la cohésion entre phrases consécutives, et non entre phrases distantes.

En ce qui concerne la segmentation de textes d'une manière générale, la plupart des travaux portant sur le sujet ne considèrent que l'aspect thématique des segments. Dans le domaine, il est important de mentionner notamment l'algorithme *TextTiling*, basé sur la notion de rupture lexicale [Hearst, 1997]. Il s'agit de l'un des algorithmes les plus communément utilisés pour la segmentation automatique de texte. Si l'algorithme détecte une rupture dans la cohésion lexicale du texte (entre deux blocs consécutifs), il place une frontière pour indiquer un changement thématique. Bien que *TextTiling* soit capable de fonctionner correctement à l'échelle d'un courriel, il ne répond pas directement à notre problème puisque, comme nous l'avons dit, nous cherchons à effectuer une segmentation rhétorique et non thématique.

Nous sommes au courant des travaux récents portant sur la segmentation de texte linéaire, tels que [Kazantseva & Szpakowicz, 2011], qui tente de résoudre le problème en modélisant le texte sous la forme d'un graphe de phrases et y appliquant des méthodes de regroupement ou de découpage. Cependant, en raison de la petite taille des messages (et par conséquent du modeste volume de matériau lexical que nous avons à disposition), il ne nous est généralement pas possible d'exploiter ce genre de méthode.

2. `fr.wikipedia.org/wiki/Machine_à_vecteurs_de_support`

2.3 Corpus de courriers électroniques

La plupart des travaux portant sur les actes du langage et les messages évitent d'annoter eux-mêmes leurs corpus et préfèrent faire appel à des corpus distribués dans la communauté scientifique. Cependant, et notamment en raison de problématiques dues au respect de la vie privée, peu de conversations sont disponibles publiquement.

Le corpus Enron [3] contient plus de 600 000 courriels envoyés par 158 employés de la compagnie Enron [Klimt & Yang, 2004b]. En 2010, EDRM a publié une version étendue de ce corpus, contenant plus de 1,7 millions de messages [4].

Le W3C [5] est le résultat de la récupération de 50 000 fils de conversation tirés du *World Wide Web Consortium*. Le corpus est constitué de courrier de type "entreprise". Les messages extraits de la liste de diffusion de w3c.org est constituée d'environ 200 000 documents. Il est utilisé par [Joty *et al.* , 2011] pour la modélisation non supervisée d'actes du dialogue dans les courriels. [Tavafi *et al.* , 2013] l'utilise en tant que jeu de données non annotées pour une tâche de classification semi-supervisée des actes du langage.

Le *British Columbia Conversation Corpus* (BC3) est utilisé par [Tavafi *et al.* , 2013] comme jeu de données annotées pour une tâche de classification semi-supervisée des actes du langage. Il contient 40 fils de discussion tirés du corpus W3C. Chaque fil a été annoté par trois annotateurs différents. Les métadonnées produites comportent notamment des résumés (par extraction) et des actes de discours (*Propose*, *Request*, *Commit* et *Meeting*) [Ulrich *et al.* , 2008a].

Bien que ce dernier corpus présente certains avantages évidents, ils sont tous constitués de courriels d'entreprise et aucun n'est directement pertinent dans le contexte de l'amélioration de systèmes d'assistance aux utilisateurs. Nous obtiendrons donc nos données autrement, comme nous le montrerons plus tard.

3. `http://www.cs.cmu.edu/~./enron/`
4. EDMR Enron Email Data Set v2 Now Available : `http://www.edrm.net/archives/6462`
5. `http://research.microsoft.com/enus/um/people/nickcr/w3csummary.html`

2.4 Exploitation d'efforts humains

Nous pouvons assimiler notre approche au genre des approches collaboratives pour obtenir des corpus annotés, tels que le *Game With A Purpose* (GWAP) [von Ahn, 2006] - technique qui consiste à faire faire par des humains, au travers une interface ludique, les étapes d'un processus trop complexes pour être effectuées par une machine - ou le *crowdsourcing* payé [Fort *et al.* , 2011] (comme la plate-forme Mechanical Turk de Amazon par exemple). Dans la taxonomie développée par [Wang *et al.* , 2013] pour catégoriser ce type d'approches, la notre pourrait être assimilée au genre *Wisdom of the Crowds* (WotC) où les motivations sont l'altruisme ou le prestige pour collaborer à la construction d'une ressource publique, prédire l'issue de certains événements, etc.

Une différence majeure entre notre travail et ces approches est que nous n'avons pas initié le processus d'étiquetage et par conséquent nous n'avons pas défini de directives d'annotation, ce qui est toujours une tâche problématique : nous nous sommes contenté de détourner *a posteriori* le résultat d'une tâche existante effectuée dans un contexte distinct. Ici, la motivation ne serait pas le prestige, mais la volonté de se plier à une étiquette ainsi que le désir d'envoyer un message clairement formaté.

Chapitre 3

Étiquetage automatique de corpus

Dans ce chapitre, nous présentons notre hypothèse ainsi que les étapes détaillées de notre approche pour l'étiquetage automatique de corpus.

3.1 Hypothèses

Premièrement, nous supposons qu'un message peut être divisé en segments du discours subséquents et consécutifs, chacun porteur de son propre acte du langage. On considère la phrase comme l'unité élémentaire dont sont constitués ces segments.

Deuxièmement, nous partons du postulat que, lorsqu'un internaute répond à un courriel et qu'il en reprend certains passages dans son message, il effectue des opérations cognitives pour identifier des fragments de texte autonomes : la partie citée consiste en une unité d'information homogène. Par conséquent, ces opérations peuvent être interprétées comme des opérations d'annotation. Les suppositions que l'on peut faire sur le type d'annotation dont il s'agit dépendent de l'opération qui a été effectuée. Ainsi, par exemple, la suppression ou la reprise de texte originel peut donner des indices sur la pertinence du contenu : du texte rejeté est probablement moins pertinent que du texte réutilisé.

En effet, comme recommandé par la *Netiquette*[1], lorsque l'on répond à un message (que ce soit un courriel ou un message de forum), on devrait soit résumer le message d'origine en haut de son message, soit inclure (ou "citer") juste assez de texte du message original pour lui donner du contexte, de manière à s'assurer que les lecteurs comprennent de quoi il s'agit quand ils commencent à lire la réponse. Nous utilisons cet effort à notre profit, en particulier quand le rédacteur répond à un message en insérant sa réponse ou son commentaire juste après le texte cité approprié pour son intervention. Lorsqu'un internaute répond à plusieurs points d'un courriel en insérant du nouveau contenu entre différents blocs de citations (i.e. des groupes de lignes citées consécutives, facilement reconnues grâce au chevron qui les préfixent), il s'agit alors du style de format "interfolié", celui qui nous intéresse.

La figure 3.1 montre un exemple de message original et une de ses réponses. On voit clairement que la réponse ne reprend que quatre phrases du message source, à savoir $S2$, $S3$, $S4$ et $S5$, qui correspondent respectivement aux phrases $R2$, $R3$, $R4$ et $R6$ dans la réponse. $R2$, $R3$ et $R4$ constituent un premier bloc de citation, et $R2$ constitue le second.

3.2 Schéma d'annotations

Comme nous l'avons vu, nous supposons que lorsqu'une personne ajoute du nouveau contenu entre deux blocs de texte cité, il effectue un découpage du message originel. On peut supposer que la première phrase d'une partie citée comporte des instructions pour ouvrir un nouveau segment de discours tandis que la dernière phrase comporte des instructions pour achever le segment. Par conséquent, nous pouvons effectuer certaines suppositions par rapport au rôle joué par ces phrases dans la structure informationnelle du message d'origine. Une phrase dans un segment peut jouer l'un des rôles suivants : *starting and ending* (SE), si elle constitue un segment à elle seule, *starting* (S), si elle débute un segment, *inside* (I), si elle n'est ni en début ni en fin de segment, et *ending* (E), si elle termine un segment.

Ce schéma est similaire au schéma *BIO* à la différence qu'il est appliqué au niveau de la phrase

1. Ensemble de recommandations pour les communications sur Internet. Le document officiel définissant les règles de la nétiquette est la RFC 1855 (`http://tools.ietf.org/html/rfc1855`).

[Hi !]^{S1}

[I got my ubuntu cds today and i'm really impressed.]^{S2} [My friends like them and my teachers too (i'm a student).]^{S3} [It's really funny to see, how people like ubuntu and start feeling geek and blaming microsoft when they use it.]^{S4}

[Unfortunately everyone wants an ubuntu cd, so can i download the cd covers anywhere or an 'official document' which i can attach to self-burned cds ?]^{S5}

[I searched the entire web site but found nothing.]^{S6} [Thanks in advance.]^{S7}

[John]^{S8}

Message source.

[On Sun, 04 Dec 2005, John Doe <john@doe.com>wrote :]^{R1}

>[I got my ubuntu cds today and i'm really impressed.]^{R2} [My friends like them and
>my teachers too (i'm a student).]^{R3} [It's really funny to see, how people like ubuntu
>and start feeling geek and blaming microsoft when they use it.]^{R4}

[Rock !]^{R5}

>[Unfortunately everyone wants an ubuntu cd, so can i download the cd covers
>anywhere or an 'official document' which i can attach to self-burned cds ?]^{R6}

[We don't have any for the warty release, but we will have them for hoary, because quite a few people have asked. :-)]^{R7}

[Bob.]^{R8}

Message de réponse.

FIGURE 3.1: Un message originel (ou "message source") et sa réponse (tirés de l'archive de courriers électroniques *ubuntu-users*). Les différentes phrases ont été clairement indiquées.

Source	Réponse	Étiquette
S1		
	R1	
S2	*>R2*	`Start`
S3	*>R3*	`Inside`
S4	*>R4*	`End`
	R5	
S5	*>R6*	`Start&End`
S6		
S7		
S8		
	R7	
	R8	

FIGURE 3.2: Alignement des phrases tirées des messages montrés dans la figure 3.1, ainsi que les étiquettes inférées de la reprise de texte du message source. Les étiquettes sont associées aux phrases d'origine.

et non au niveau du token [Ratinov & Roth, 2009].

La figure 3.2 illustre ce schéma en montrant comment les phrases de la figure 3.1 peuvent être alignées et comment les étiquettes peuvent en être inférées.

3.3 Procédure de génération des données annotées

Avant de pouvoir prédire les étiquettes des phrases du message originel, il est nécessaire d'identifier celles qui ont été réutilisées dans un message de réponse. La seule identification des lignes citées dans le message de réponse est insuffisante pour diverses raisons. Premièrement, le segmenteur est supposé fonctionner sur des données non-bruitées (i.e. les nouveaux contenus dans les messages) alors qu'un texte cité est une version altérée du texte originel. En effet, certains clients de messagerie électronique ne respectent pas toujours les standards et ne sont pas forcément toujours compatibles entre eux [2]. En particulier, l'absence de certaines métadonnées peut causer un ré-encodage erroné des blocs de citation à chaque échange. De plus, les programmes clients peuvent intégrer leurs propres mécanismes pour citer les précédents messages, ou encore

2. Les *Request for Comments* (RFC) sont des règles et protocoles proposés par les groupes de travail participant à l'*Internet Standardization* (`https://tools.ietf.org/html`). Certains RFC sont consacrés aux formats des courriels et aux spécifications d'encodage (voir RFC 2822 et 5335 pour commencer). Il y a eu de nombreuses propositions, parfois mises à jour et donc parfois rendues caduques, ce qui peut expliquer certains problèmes de compatibilité.

tronquer les lignes trop longues [3]. Deuxièmement, accéder aux messages originels peut permettre de prendre en compte certains traits contextuels (comme la disposition visuelle par exemple). Troisièmement, pour aller plus loin, le contexte originel du texte extrait contient également de l'information sur la segmentation d'un message. Par exemple, une phrase du message originel, qui ne serait pas présente dans la réponse, mais qui suit une phrase alignée, peut être considérée comme débutant un nouveau segment. Pour ces trois raisons, en plus d'identifier les lignes citées, nous devons suivre une procédure d'alignement pour obtenir leurs versions d'origine.

La procédure de génération des données annotées suit les étapes suivantes :

1. Les messages postés dans le style interfolié sont identifiés
2. Pour chaque paire message source / réponse :
   (a) Les deux messages sont tokenisés au niveau de la phrase et du mot (voir sous-section 3.3.1 pour le détail des techniques employées pour la tokenisation)
   (b) Les lignes citées présentes dans la réponse sont identifiées
   (c) Les phrases qui font partie du texte cité dans le message de réponse sont identifiées
   (d) Les phrases du message d'origine sont alignées avec le texte cité dans la réponse (voir sous-section 3.3.1 pour le détail de la procédure d'alignement)
   (e) Les phrases alignées sont étiquetées (voir sous-section 3.3.3 pour le détail de l'algorithme d'étiquetage)
   (f) La séquence de phrases alignées est ajoutée au jeu de données

Les paires de messages sources et leurs réponses sont constituées à partir des champs *In-Reply-To* de leurs entêtes [4].

3. Fonctionnalité utilisée pour rendre le texte lisible sans avoir à invoquer la barre de défilement horizontale. Les phrases sont généralement découpées en segments d'environ 80 caractères.
4. Les champs des entêtes de courriels sont définis par le RFC 5322. Le champ *In-Reply-To* contient l'identifiant (*Message-ID*) du message parent (c'est à dire le message auquel celui-ci fait réponse).

3.3.1 Tokenisation

Nous n'utilisons pas de segmenteur en phrases ou de tokeniseur connu parce qu'il n'en existe pas de disponible spécialisé pour les courriels. En effet ces documents ont des caractéristiques spécifiques qu'il faut considérer comparativement aux segmenteurs de texte écrits en anglais canonique. L'approche employée pour la tokenisation des messages suit globalement la stratégie du système de tokenisation en mots qui vient avec le *TreeTagger* [Schmid, 1994], c'est à dire qu'il se concentre sur les marques de segmentation et l'analyse récursive des marques de ponctuation qui "collent" les débuts et fins de mots et phrases.

Pour chaque courriel :

(a) Reconnaissance des marques de segmentation (phrases)

(b) Correction des segmentations abusives

(c) Pour chaque phrase obtenue :

i. Reconnaissance des marques de segmentation (mots)

ii. Correction des segmentations abusives

Pour chaque sous temps (reconnaissance et correction), les règles utilisées sont ordonnées des plus sûres au moins sûres.

3.3.2 Alignement

Pour trouver les alignements entre deux messages donnés, nous utilisons un algorithme d'alignement de chaînes basé sur la programmation dynamique (DP) [Sankoff & Kruskal, 1983].

Dans le contexte de la reconnaissance de la parole, cet algorithme est aussi connu sous le nom de *NIST align/scoring algorithm*. En effet il est largement utilisé pour évaluer les systèmes de reconnaissance de la parole en comparant leurs sorties au texte de référence. Il est utilisé en particulier pour calculer deux taux d'erreur : le *Word Error Rate* (WER) et le *Sentence Error rate* (SER).

L'algorithme fonctionne en cherchant à minimiser globalement la distance de Levenshtein [5] [Levenshtein, 1966] en attribuant aux mots corrects, aux insertions, aux suppressions et aux substitutions des poids de respectivement 0, 3, 3 et 4. L'algorithme est de complexité $O(MN)$.

L'Université de Carnegie Mellon fournit une implémentation de cet algorithme dans son kit de reconnaissance de la parole [6].

3.3.3 Étiquetage

L'étiquetage d'une phrase alignée (phrase du message source réutilisée dans la réponse) se fait suivant un simple algorithme à base de règles :

Pour chaque phrase source alignée :

(a) si la phrase est entourée par du nouveau contenu dans la réponse, l'étiquette est `Start&End`

(b) sinon si la phrase est précédée par du nouveau contenu, l'étiquette est `Start`

(c) sinon si la phrase est suivie par du nouveau contenu, l'étiquette est `End`

(d) sinon, l'étiquette est `Inside`

5. Article Wikipédia sur la distance de Levenshtein : `http://fr.wikipedia.org/wiki/Distance_de_Levenshtein`

6. Sphinx 4, *edu.cmu.sphinx.util.NISTAlign*, `http://cmusphinx.sourceforge.net`

Chapitre 4

Segmentation de courriels

Dans ce chapitre, nous présentons notre approche pour la segmentation de courriels ainsi que les traits utilisés pour l'entraînement du classifieur.

4.1 Étiquetage de séquences

En traitement automatique du langage naturel, l'approche supervisée reste généralement la méthode la plus stable et la plus efficace pour résoudre les problèmes de classification. Notre but est donc d'entraîner un système à détecter les frontières des segments, c'est à dire de déterminer, via une approche de classification, et pour chaque phrase, si elle débute un nouveau segment ou non.

Plus spécifiquement, nous choisissons de traiter le problème comme une tache d'étiquetage de séquences dont l'objectif est d'attribuer globalement le meilleur ensemble d'étiquettes pour la séquence entière d'un seul coup [1]. Dans cette perspective, chaque courriel est traité comme une séquence de phrases. L'idée sous-jacente est que l'étiquette la plus pertinente pour une phrase est dépendante des traits et de l'étiquette des phrases proches.

Notre segmenteur est basé sur un classifieur utilisant les champs aléatoires de Markov, tel

1. Un exemple classique de tâche accomplie de cette manière est l'étiquetage morpho-syntaxique, qui cherche à identifier la nature grammaticale de chaque terme d'une phrase ou d'un document.

qu'implémenté dans le programme d'étiquetage de séquences *Wapiti* [Lavergne *et al.* , 2010]. Nous fixons la taille de la fenêtre à 5, c'est à dire que l'algorithme prend en compte non seulement les traits de la phrase qu'il cherche à étiqueter mais également ceux des deux phrases précédentes et des deux phrases suivantes.

Entraîner le classifieur à reconnaître les différents labels du schéma d'annotation précédemment déterminé peut être problématique. En effet, il présente certains inconvénients qui peut nuire à l'efficacité du classifieur. En particulier, les phrases étiquetées *SE* partageront, par définition, d'importantes caractéristiques avec les phrases étiquetées *S* et *E*. Nous choisissons donc de transformer ces annotations en un schéma binaire et nous contentons de différencier les phrases qui débutent un nouveau segment (*True*), ou "phrases-frontières", de celles qui ne débutent pas un nouveau segment (*False*). Le processus de conversion est trivial, et peut facilement être inversé.

Procédure de conversion :

Pour chaque phrase :

(a) si la phrase est étiquetée *SE* ou *S*, l'étiquette devient *True*

(b) sinon, elle devient *False*

Procédure inverse, pour retrouver les étiquettes d'origine :

Pour chaque phrase :

(a) si l'étiquette de la phrase courante est *True* :

  (i) si la phrase suivante est étiquetée *True*, elle devient *SE*

  (ii) sinon, elle devient *S*

(b) sinon :

  (i) si la phrase suivante est étiquetée *True*, elle devient *E*

  (ii) sinon, elle devient *I*

4.2 Ensembles de traits

On distingue cinq ensembles de traits : les n-grammes, les traits basés sur la théorie de la structure de l'information, les traits thématiques, les traits stylistiques et les traits sémantiques (dans le cadre des expériences, les deux derniers ensembles sont regroupés sous l'appellation "traits divers"). Tous les traits sont indépendants du domaine et presque tous les traits sont indépendants du langage, à l'exception des traits sémantiques, qui peuvent néanmoins être facilement traduits.

Pour construire les traits du segmenteur, nous utilisons l'étiqueteur de Stanford pour l'étiquetage morpho-syntaxique [Toutanova *et al.* , 2003], et la base de données lexicale *WordNet* pour la lemmatisation [Miller, 1995].

4.2.1 n-grammes

On sélectionne, de manière insensible à la casse, les 1000 [2] bigrammes et trigrammes apparaissant dans le plus grand nombre de phrases du corpus (ou *document frequency*). Puisque la probabilité d'avoir de multiples occurrences d'un même n-gramme dans une phrase est extrêmement faible, nous ne conservons pas le nombre d'occurrences mais une valeur booléenne pour ne considérer que la présence ou l'absence du n-gramme.

4.2.2 Traits basés sur la théorie de la structure de l'information

Cet ensemble de traits est inspiré de la théorie de la structure de l'information, qui décrit l'information portée par une phrase en fonction de la façon dont elle est reliée à son contexte [Kruijff-Korbayová & Kruijff, 1996]. La théorie affirme l'importance de constructions syntaxiques particulières et de l'ordre des mots dans la phrase. En effet pour des langages comme l'anglais ou le français, le début de la phrase est une position importante pour structurer l'information au niveau du discours, tandis que la fin de la phrase peut comporter de

2. Valeur estimée empiriquement.

l'information utile pour annoncer ce qui vient ensuite.

On s'intéresse aux trois premiers et trois derniers tokens significatifs de la phrase. Un token est considéré comme significatif si sa fréquence est supérieure à $1/2\ 000$ [3]. Si une phrase contient moins de six tokens significatifs, le même token peut se retrouver dans les deux triplets. Si la phrase contient moins de trois tokens significatifs, les valeurs manquantes sont remplacées par une valeur spéciale "bouche-trou". Nous définissons trois traits individuels pour chacun des trois unigrammes, les deux bigrammes et le trigramme qui se trouvent dans chacun de ces triplets. Les traits sont les suivants : la forme de surface de chaque token (sensible à la casse), leur forme lemmatisée (insensible à la casse) et leur étiquette morpho-syntaxique. Ces traits sont illustrés par la figure 4.1.

4.2.3 Trait thématique

Le seul trait que nous prenons en compte pour la reconnaissance des variations thématiques est la sortie de l'algorithme *TextTiling* [Hearst, 1997], dont nous avions détaillé le fonctionnement lors de notre étude bibliographique. En raison de la taille relativement courte des courriels, nous définissons la taille d'un bloc comme égale à trois fois la taille moyenne d'une phrase dans notre corpus. Nous définissons la "taille-étape" (la distance parcouru par la fenêtre glissante à chaque étape) comme égale à la taille moyenne d'une phrase du corpus.

4.2.4 Traits divers

Cet ensemble inclut les traits stylistiques et sémantiques. Il contient 24 traits, plusieurs ayant été empruntés à des travaux dans le domaine de la classification d'actes de langage [Qadir & Riloff, 2011] et de la segmentation de courriels [Lampert *et al.* , 2009b].

Les traits stylistiques capturent l'information portant sur la structure visuelle et la composition du message :

3. Cette valeur a été déterminée empiriquement par rapport à nos données. Un travail supplémentaire devra être effectué pour la généraliser.

- la position de la phrase dans le courriel
- la taille moyenne des tokens
- le nombre total de tokens
- le nombre total de caractères
- la proportion de majuscules
- la proportion de caractères alphabétiques
- la proportion de caractères numériques
- le nombre de chevrons
- si la phrase finit sur ou contient un point d'interrogation, une virgule ou un point-virgule
- si la phrase contient des caractères de ponctuation parmi ses trois premiers tokens (pour reconnaître les salutations [Qadir & Riloff, 2011]).

Les traits sémantiques cherchent à identifier certains mots et formules particuliers :

- si la phrase commence par un mot interrogatif de type "wh" (*"who"*, *"when"*, *"where"*, *"what"*, *"which"*, *"what"*, *"how"*)
- si la phrase contient un mot interrogatif de type "wh"
- si la phrase commence par une forme interrogative (e.g. "*is it*", "*are there*"...)
- si la phrase contient une forme interrogative
- si la phrase contient un modal (*"can"*, *"may"*, *"must"*, *"shall"*, *"will"*, *"might"*, *"should"*, *"would"*, *"could"*, et leurs formes négatives)
- si la phrase contient une formule de planification (e.g. "*I will*", "*we are going to*"...)
- si la phrase contient des indices de la première personne (e.g. "*we*", "*my*"...)
- si la phrase contient des indices de la deuxième personne
- si la phrase contient des indices de la troisième personne
- le premier pronom personnel trouvé dans la phrase
- la première forme verbale rencontrée, telle qu'étiquetée par l'étiqueteur de Stanford, c'est à dire un élément du *Penn Treebank tag set*[4] (e.g. le trait *"VBZ"* indique un verbe au présent et à la troisième personne du singulier).

4. Liste alphabétique des étiquettes morpho-syntaxiques utilisées par le *Penn Treebank Project* : `http://www.ling.upenn.edu/courses/Fall_2003/ling001/penn_treebank_pos.html`

Formes de surface	**Lemmes**	**Étiquettes**
Many	many	JJ
thanks	thanks	NNS
to	to	TO
your	your	PRP
suggestions	suggestion	DD
.	.	.
Many thanks	many thanks	JJ NNS
thanks to .	thanks to .	NNS TO .
your suggestions	your suggestion	PRP DD
suggestions	suggestion	DD
Many thanks to	many thanks to	JJ NNS TO
your suggestions .	your suggestion .	PRP DD .

TABLE 4.1: Traits syntaxiques formés par la phrase "*Many thanks to all of you for the help you have offered, I have learned tremendously from all your suggestions*". Chaque cellule est un trait (36 au total).

Chapitre 5

Cadre expérimental

Dans ce chapitre, nous décrivons les corpus et protocoles d'évaluation employés pour effectuer nos expériences.

5.1 Corpus

Le travail s'inscrivant dans le cadre d'un projet portant sur le traitement de discussions multilingues et multimodales, principalement orientées autour des demandes d'informations techniques, nous n'avons pas retenu le corpus Enron (30 000 fils de discussion) [Klimt & Yang, 2004a] (qui vient d'un environnement business) ni le corpus W3C (malgré son caractère technique) ou le British Columbia Conversation Corpus (BC3) qui en est tiré [Ulrich *et al.* , 2008b].

Nous préférons employer l'archive de courriels *ubuntu-users*[1], qui contient les messages de la liste de diffusion d'assistance aux utilisateurs de la distribution Ubuntu, comme corpus principal. Il est gratuit, et distribué sous une licence non restrictive. Il continue de grandit perpétuellement, et est donc représentatif des pratiques de messagerie électronique à la fois en terme de contenu et de format. De plus, de nombreuses archives alternatives sont disponibles, dans un grand nombre de langues différentes, y compris certaines langues très pauvres en ressources. Ubuntu

1. Archives des listes de diffusion Ubuntu : `https://lists.ubuntu.com/archives/`

propose également un forum et une FAQ qui peuvent se révéler intéressantes dans le contexte d'études multimodales.

Nous utilisons une copie datant de décembre 2013. Le corpus contient un total de 272 380 messages (47 044 fils de conversation). 33 915 d'entre eux sont postés dans le style interfolié qui nous intéresse. Les messages sont faits de 418 858 phrases, elles mêmes constituées de 76 326 tokens uniques (5 139 123 au total). 87 950 de ces phrases (21%) ont été automatiquement étiquetées par notre système comme débutant un nouveau segment (soit *SE* soit *S*).

5.2 Protocole d'évaluation

Pour évaluer l'efficacité du segmenteur, nous effectuons une validation croisée à 10 échantillons sur le corpus Ubuntu, et comparons ses performances à deux systèmes de référence distincts. Les résultats sont mesurés par l'intermédiaire d'un ensemble de métriques utilisées en segmentation de texte et en recherche d'information (RI).

5.2.1 Systèmes de référence

Le premier système de référence, le système "régulier", est calculé en segmentant l'ensemble de test en segments réguliers de même taille que le segment moyen pour l'ensemble d'entraînement, arrondi au supérieur. Le second est l'algorithme *TextTiling* que nous avions décrit au chapitre 4. Bien qu'utilisée comme un trait dans l'approche proposée dans ce chapitre, ici c'est la sortie directe de l'algorithme qui est utilisée comme point de comparaison.

5.2.2 Métriques

La Précision (P) et le Rappel (R) sont fournis pour tous les résultats. P est la proportion de frontières identifiées par le classifieur qui sont bien de vraies frontières. R est la proportion

de vraies frontières qui ont été correctement identifiées par le classifieur. Nous fournissons également la F-mesure (F_1) qui représente leur pondération :

$$F_1 = 2 \cdot \frac{P \cdot R}{P + R}$$

Cependant, l'évaluation automatique de segmentations de textes au travers de ces seules métriques est problématique car les segments prédits sont rarement précisément alignés avec les segments de référence. De plus, bien qu'un segmenteur qui placerait des frontières juste à côté de leur emplacement correct est presque toujours plus approprié qu'un segmenteur qui manque par une marge bien plus grande, la Précision et le Rappel pénaliseraient les deux dans la même mesure. Par conséquent, pour pouvoir évaluer des degrés variés de succès et d'échec de manière plus subtile, nous proposons également un ensemble de métriques pertinentes dans le cadre de l'évaluation de segmentation de textes : P_k, *WindowDiff* et la distance de Hamming généralisée (*GHD*).

La mesure P_k prend en compte la distance qui se trouve entre les frontières prédites et celles qui auraient dû être trouvées [Beeferman *et al.* , 1999]. Elle évalue une probabilité d'erreur prenant en compte la probabilité pour deux phrases séparées par k phrases d'être localement dans les mêmes segments du document de référence et du document produit par le segmenteur, c'est à dire qu'aucune frontière ne les sépare dans les deux cas. La distance k est typiquement définie comme valant la moitié de la longueur moyenne des segments du document (c'est également de cette manière que nous la définissons pour évaluer nos résultats).

WindowDiff, inspiré de P_k, compare le nombre de frontières trouvées dans une fenêtre de taille k au nombre de frontières trouvées dans la même fenêtre de texte pour la segmentation de référence [Pevzner & Hearst, 2002].

La *GHD* est une extension de la distance de Hamming [2] qui donne un crédit partiel pour les échecs mineurs [Bookstein *et al.* , 2002].

2. Article Wikipédia sur la distance de Hamming : `http://fr.wikipedia.org/wiki/Distance_de_Hamming`

Chapitre 6

Expériences et résultats

Dans ce chapitre, nous décrivons les prétraitements opérés sur les données utilisées avant d'exposer les résultats obtenus pour différentes expériences.

6.1 Prétraitements

Pour réduire le bruit présent dans le corpus, nous filtrons les courriels indésirables en nous basant sur plusieurs critères, le premier d'entre eux étant l'encodage. Les messages qui ne sont pas encodés en UTF-8 sont retirés de la sélection. Le second critère est le type MIME : nous ne conservons que les messages en texte brut uniquement, et retirons ceux contenant du HTML ou d'autres contenus spéciaux.

De plus, nous choisissons de ne considérer que les réponses aux messages initiaux (les premiers messages d'une discussion). Ce choix est justifié par la supposition que nous faisons que le module d'alignement aura plus de difficultés à reconnaître correctement des phrases qui ont été transformées à plusieurs reprises en étant reprises par un certain nombre de réponses successives. En effet, ces réponses - qui contiendraient du texte cité d'autres messages - seraient plus probablement mal étiquetées par notre système d'annotation automatique.

Le dernier critère est la longueur. Le jeu de données étant construit à partir d'une liste de

diffusion pouvant couvrir des discussions très techniques, les utilisateurs peuvent parfois envoyer des messages contenant de nombreuses lignes de code copié-collé, des logs logiciels, des sorties de commandes bash, etc. Le nombre de ces messages est marginal, mais leur longueur peut être tellement disproportionné qu'ils peuvent tout de même avoir un impact négatif sur les performances du segmenteur. Par conséquent, nous excluons les messages de taille supérieure à la moyenne plus la déviation standard de la taille des messages.

Après filtrage, le jeu de données ne comporte plus que 6 821 des 33 915 messages (soit 20%).

6.2 Résultats et discussion

La table 6.1 montre l'ensemble des résultats obtenus par validation croisée. Les résultats calculés selon les métriques de segmentation se trouvent à gauche, et ceux calculés selon les métriques de recherche d'information à droite. Sont d'abord indiqués les scores des systèmes de référence, dans la section supérieure. Ensuite, dans la section intermédiaire, nous montrons les résultats des segmenteurs basés sur chaque ensemble de traits (avec A correspondant aux n-grammes, B aux traits basés sur la théorie de la structure de l'information, C sur *TextTiling* et D aux traits stylistiques et sémantiques). Enfin, dans la section inférieure, nous montrons les résultats des segmenteurs basés sur des combinaisons d'ensembles de traits.

6.2.1 Systèmes de référence (*baselines*)

La première section de la table 6.1 montre les résultats obtenus par les deux systèmes de référence. Sans surprise, *TextTiling* se montre beaucoup plus performant que l'approche basée sur une segmentation régulière, et ce selon toutes les métriques sauf le rappel.

6.2.2 Segmenteurs basés sur des ensembles de traits homogènes

La seconde section de la table 6.1 montre les résultats pour quatre différents segmenteurs, chacun entraîné avec un ensemble de traits distinct. La fonction ϕ est la fonction de classification, ses paramètres sont des traits, et sa sortie une prédiction. Tandis que tous les classifieurs battent sans problème le système de segmentation régulière, et rivalisent avec *TextTiling* en ce qui concerne les métriques de recherche d'information, seuls le segmenteur thématique et celui basé sur les n-grammes parviennent à le surpasser quand la performance est mesurée par les métriques de segmentation. En termes de scores de RI, le classifieur utilisant les n-grammes sort clairement du lot puisqu'il parvient à atteindre une précision exceptionnelle de 100%, bien que ce résultat soit mitigé par un maigre rappel (39%). Il est également intéressant de noter que le classifieur thématique, basé seulement sur la connaissance contextuelle des prédictions de *TextTiling*, surpasse la sortie brute de l'algorithme.

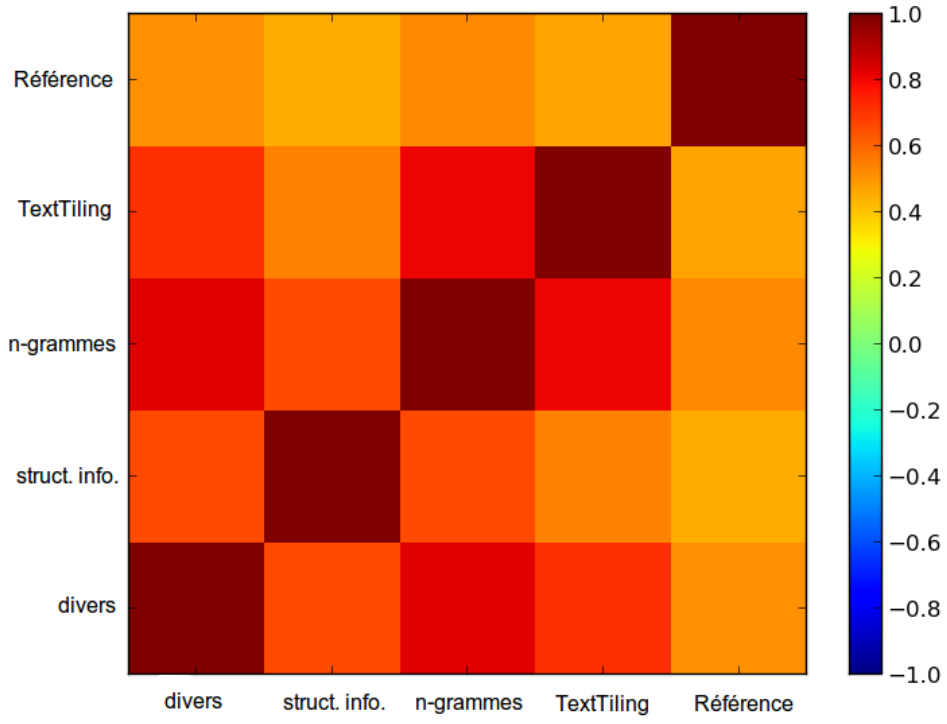

FIGURE 6.1: Matrice de corrélation pour les sorties des segmenteurs basés sur des ensembles de traits homogènes.

La figure 6.1 représente la matrice de corrélation entre les sorties de ces différents systèmes.

D'une manière générale, on observe des taux de corrélation modérés entre les différents segmenteurs, ce qui nous permet de supposer que les combiner pourrait améliorer les résultats obtenus.

6.2.3 Segmenteurs basés sur des combinaisons d'ensembles de traits

La dernière section de la table 6.1 montre les résultats pour quatre différents segmenteurs.

Le premier, $\phi(A + B + C + D)$, est un simple classifieur qui prend en compte tous les traits disponibles. Ses résultats sont exactement identiques à ceux du classifieur n-grammes, ce qui est probablement dû au fait que les autres traits sont noyés dans la masse de traits lexicaux (1000 n-grammes sont pris en compte).

Le second, $\phi(\phi(A)+\phi(B)+\phi(C)+\phi(D))$, utilise comme traits les sorties des quatre classifieurs entraînés sur des ensembles de traits différents décrits dans la sous-section précédente. Les résultats montrent que cette approche n'est pas significativement plus intéressante.

Le troisième, $\phi(A) \cup \phi(B + C + D)$, segmente selon l'union des frontières détectées par un classifieur entraîné sur les n-grammes et celles identifiées par un classifieur entraîné sur tous les autres traits. Cette idée est motivée par le fait que nous savons que toutes les frontières trouvées par le classifieur basé sur les n-grammes sont correctes ($P = 1$). Cette approche nous permet d'obtenir le meilleur rappel possible ($R = .69$), mais en contrepartie d'une bonne précision ($P = .58$). Le rappel n'est pas si élevé parce que, comme le montre la figure 6.1, le segmenteur basé sur les n-grammes est de base assez clairement corrélé avec les autres (le taux de corrélation varie entre .5 et .8 selon les systèmes).

Le dernier, $\phi(A) \cup \delta(\phi(B + C + D))$, tente d'améliorer le rappel du classifieur basé sur les n-grammes sans sacrifier trop de précision en se montrant plus sélectif en acceptant de nouvelles frontières. La fonction δ est la fonction de sélection, qui ignore les frontières prédites avec une faible confiance. Seules celles identifiées par le classifieur basé sur les n-grammes et celles identifiées avec un taux de confiance d'au moins 99% par un classifieur entraîné sur tous les autres traits sont prises en considération. Ce système obtient des performances supérieures

	Métriques de seg.			Métriques de RI		
	WD	P_k	GHD	P	R	F_1
Segmentation régulière	.59	.25	.60	.31	.49	.38
TextTiling	.41	.07	.38	.75	.44	.56
$\phi(A)$ avec A = n-grammes	.38	**.05**	.39	**1**	.39	.56
$\phi(B)$ avec B = structure info.	.43	.11	.38	.60	.68	**.64**
$\phi(C)$ avec C = TextTiling	.39	.05	.38	.94	.40	.56
$\phi(D)$ avec D = traits divers	.41	.09	.38	.69	.49	.57
$\phi(A+B+C+D)$	.38	**.05**	.39	**1**	.39	.56
$\phi(\phi(A)+\phi(B)+\phi(C)+\phi(D))$	.38	.06	.36	.81	.47	.59
$\phi(A) \cup \phi(B+C+D)$	.45	.12	.40	.58	**.69**	.63
$\phi(A) \cup \delta(\phi(B+C+D))$	**.36**	.06	**.34**	.80	.53	**.64**

TABLE 6.1: Résultats comparés entre les différents systèmes de référence et les segmenteurs testés. Tous les résultats présentent *WindowDiff* (WD), P_k et GHD en tant que taux d'erreur, par conséquent un score bas est désirable pour ces métriques. Ceci contraste avec les trois scores de RI, pour lesquels une maigre valeur représente une faible performance. Les meilleurs scores sont indiqués en gras.

à tous les autres à la fois en terme de scores de segmentation et de F_1, cependant il reste relativement conservateur et le ratio de segmentation (le nombre de frontières prédites divisé par le nombre de frontières réelles) reste significativement plus bas que voulu, à 0.67. Jouer avec le taux de confiance minimum (c) permet d'ajuster P de .58 ($c = 0$) à 1 ($c = 1$) et R de .39 ($c = 1$) à .69 ($c = 0$).

Chapitre 7

Conclusion

Dans ce chapitre, nous commençons par rappeler les apports de notre travail, avant de détailler comment nous pouvons l'améliorer dans le futur. Enfin, nous mentionnons les publications qui en ont été tirées.

7.1 Réalisations

La contribution principale de ce travail est une technique permettant d'exploiter les efforts cognitifs effectués par des humains attelés à une tâche de mise en forme de messages de réponse pour entraîner un segmenteur discursif.

Nous avons également développé un système de segmentation visant à soutenir l'analyse de messages en termes d'actes du langage et rapporté l'évaluation de différents modèles construits à partir d'ensembles de traits variés.

Enfin, nous avons proposé un nouveau corpus pour l'analyse de discussions asynchrones en ligne, qui a l'avantage d'être vaste, moderne, multimodal et multilingue.

7.2 Perspectives

Bien qu'il soit toujours possible de les améliorer, nos résultats indiquent que notre approche mérite un examen approfondi. Notre approche de segmentation reste relativement simple et peut facilement être étendue. Une manière de le faire serait de considérer les traits contextuels pour caractériser les phrases dans la structure originelle du message où elles ont été écrites.

Comme travaux futurs, nous prévoyons également de reproduire nos expériences sur un jeu de données constituées par l'ensemble des phrases des courriels, et pas seulement les phrases reprises dans les messages qui leur font réponse. Ce faisant, nous espérons corriger un biais dû fait que notre segmenteur n'est jusqu'à présent entraîné et testé que sur les parties des courriels typiquement reprises lors d'une conversation.

Enfin, il est possible de compléter nos expériences avec deux nouvelles approches pour l'évaluation. La première consistera à comparer la segmentation automatique avec celle effectuée par des annotateurs humains. Cette tâche reste difficile puisqu'il sera alors nécessaire de définir un protocole d'annotation, des lignes directrices et de construire de nouvelles ressources. La seconde évaluation que nous prévoyons d'effectuer est une évaluation extrinsèque. L'idée est de mesurer la contribution que peut apporter la segmentation d'un courriel au processus d'identification des actes du langage, c'est à dire de vérifier si la connaissance des frontières entre segments pourrait améliorer les systèmes de classification existants.

Pour aller dans une autre direction, nous pensons qu'il serait également possible d'exploiter notre approche pour estimer l'importance de tel ou tel segment, et peut-être lui trouver de nouvelles applications dans le cadre génération automatique de résumés par extraction de phrases.

7.3 Publications

Un article basé sur ce travail, titré *Exploiting the human computational effort dedicated to message reply formatting for training discursive email segmenters*, a été soumis et accepté à *The 8th Linguistic Annotation Workshop* (LAW 8), tenu en conjonction avec *The 25th International*

Conference on Computational Linguistics (COLING 2014). Cet article est joint à ce document en annexe.

Appendices

Annexe A

Exploiting the human computational effort dedicated to message reply formatting for training discursive email segmenters

Article accepté à *The 8th Linguistic Annotation Workshop* (LAW 8), tenu en conjonction avec *The 25th International Conference on Computational Linguistics* (COLING 2014).

Exploiting the human computational effort dedicated to message reply formatting for training discursive email segmenters

Abstract

In the context of multi-domain and multimodal online asynchronous discussion analysis, we propose an innovative strategy for manual annotation of dialog act (DA) segments. The process aims at supporting the analysis of messages in terms of DA. Our objective is to train a sequence labelling system to detect the segment boundaries. The originality of the proposed approach is to avoid manually annotating the training data and instead exploit the human computational efforts dedicated to message reply formatting when the writer replies to a message by inserting his response just after the quoted text appropriate to his intervention. We describe the approach, propose a new electronic mail corpus and report the evaluation of segmentation models we built.

1 Introduction

Automatic processing of online conversations (forum, emails) is a highly important issue for the industrial and the scientific communities which care to improve existing question/answering systems, identify emotions or intentions in customer requests or reviews, detect messages containing requests for action or unsolved severe problems...

In most works, conversation interactions between the participants are modeled in terms of dialogue acts (DA) (Austin, 1970). The DAs describe the communicative function conveyed by each text utterance (e.g. question, answer, greeting,...). In this paper, we address the problem of rhetorically segmenting the new content parts of messages in online asynchronous discussions. The process aims at supporting the analysis of messages in terms of DA. We pay special attention to the processing of electronic mails.

The main trend in automatic DA recognition consists in using supervised learning algorithms to predict the DA conveyed by a sentence or a message (Tavafi et al., 2013). The hypothesized message segmentation results from the global analysis of these individual predictions over each sentence. A first remark on this paradigm is that it is not realistic to use in the context of multi-domain and multimodal processing because it requires the building of training data which is a very substantial and time-consuming task. A second remark is that the model does not have a fine-grained representation of the message structure or the relations between messages. Considering such characteristics could drastically improve the systems to allow to focus on specific text parts or to filter out less relevant ones. Indeed, apart from the closing formula, a message may for example be made of several distinct information requests, the description of an unsuccessful procedure, the quote of third-party messages...

So far, few works address the problem of message segmentation. (Lampert et al., 2009a) propose to segment emails in prototypical zones such as the author's contribution, quotes of original messages, the signature, the opening and closing formulas. In comparison, we focus on the segmentation of the author's contribution (what we call the new content part). (Joty et al., 2013) identifies clusters of topically related sentences through the multiple messages of a thread, without distinguishing email and forum messages. Apart from the topical aspect, our problem differs because we are only interested in the cohesion between sentences in nearby fragments and not on distant sentences.

Despite the drawbacks mentioned above, a supervised approach remains the most efficient and reliable method to solve classification problems in Natural Language Processing. Our aim is to train a system to detect the segment boundaries, i.e. to determine, through a classification approach, if a given sentence starts, ends or continues a segment.

[Hi!]^{S1}

[I got my ubuntu cds today and i'm really impressed.]^{S2} [My friends like them and my teachers too (i'm a student).]^{S3}
[It's really funny to see, how people like ubuntu and start feeling geek and blaming microsoft when they use it.]^{S4}

[Unfortunately everyone wants an ubuntu cd, so can i download the cd covers anywhere or an 'official document' which i can attach to self-burned cds?]^{S5}

[I searched the entire web site but found nothing.]^{S6} [Thanks in advance.]^{S7}

[John]^{S8}

(a) Original message.

[On Sun, 04 Dec 2005, John Doe <john@doe.com> wrote:]^{R1}

> [I got my ubuntu cds today and i'm really impressed.]^{R2} [My
> friends like them and my teachers too (i'm a student).]^{R3}
> [It's really funny to see, how people like ubuntu and start feeling geek
> and blaming microsoft when they use it.]^{R4}

[Rock!]^{R5}

> [Unfortunately everyone wants an ubuntu cd, so can i download the cd
> covers anywhere or an 'official document' which i can attach to
> self-burned cds?]^{R6}

[We don't have any for the warty release, but we will have them for hoary, because quite a few people have asked. :-)]^{R7}

[Bob.]^{R8}

(b) Reply message.

Figure 1: An original message and its reply (*ubuntu-users* email archive). Sentences have been tagged to facilitate the discussion.

Original	Reply	Label
S1		
	R1	
S2	> *R2*	`Start`
S3	> *R3*	`Inside`
S4	> *R4*	`End`
	R5	
S5	> *R6*	`Start&End`
	R7	
	[...]	
S6		
[...]		

Figure 2: Alignment of the sentences from the original and reply messages shown in Figure 1 and labels inferred from the re-use of the original message text. Labels are associated to the original sentences.

The originality of the proposed approach is to avoid manually annotating the training data and instead to exploit the human computational efforts dedicated to a similar task in a different context of production (von Ahn, 2006). As recommended by the *Netiquette*[1], when replying to a message (email or forum post), the writer should "summarize the original message at the top of its reply, or include (or "quote") just enough text of the original to give a context, in order to make sure readers understand when they start to read the response[2]." As a corollary, the writer should "edit out all the irrelevant material." Our idea is to use this effort, in particular when the writer replies to a message by inserting his response or comment just after the quoted text appropriate to his intervention. This posting style is called *interleaved* or *inline replying*. The so built segmentation model should be usable for any posting styles by applying it only on new content parts. Figure 1a shows an example of an *original* message and, Figure 1b, one of its *reply*. We can see that the reply message re-uses only four selected sentences from the original message; namely $S2$, $S3$, $S4$ and $S5$ which respectively correspond to sentences $R2$, $R3$, $R4$ and $R6$ in the reply message. The author of the reply message deliberately discarded the remaining of the original message. The segment build up by sentences $S2$, $S3$, $S4$ and the one by the single sentence $S5$ can respectively be associated with two acts : a comment and a question.

In Section 2, we explain our approach for building an annotated corpus of segmented online messages at no cost. In Section 3, we describe the system and the features we use to model the segmentation. After presenting our experimental framework in Section 4, we report some evaluations for the segmentation task in Section 5. Finally, we discuss our approach in comparison to other works in Section 6.

[1]Set of guidelines for Network Etiquette (*Netiquette*) when using network communication or information services RFC1855.

[2]It is true that some email software clients do not conform to the recommendations of Netiquette and that some online participants are less sensitive to arguments about posting style (many writers reply above the original message). We assume that there are enough messages with inline replying available to build our training data.

2 Building annotated corpora of segmented online discussions at no cost

We present the assumptions and the detailed steps of our approach.

2.1 Annotation scheme

The basic idea is to interpret the operation performed by a discussion participant on the message he replies as an annotation operation. Assumptions about the kind of annotations depend on the operation that has been performed. Deletion or re-use of the original text material can give hints about the relevance of the content: discarded material is probably less relevant than re-used one.

We assume that by replying inside a message and by only including some specific parts, the participant performs some cognitive operations to identify homogeneous self-contained text segments. Consequently, we make some assumptions about the role played by the sentences in the original message information structure. A sentence in a segment plays one of the following roles: `starting and ending` (*SE*) a segment when there is only one sentence in the segment, `starting` (*S*) a segment if there are at least two sentences in the segment and it is the first one, `ending` (*E*) a segment if there are at least two sentences in the segment and it is the last one, `inside` (*I*) a segment in any other cases.

Figure 2 illustrates the scheme by showing how sentences from Figure 1 can be aligned and the labels inferred from it. It is similar to the *BIO* scheme except it is not at the token level but at the sentence level (Ratinov and Roth, 2009).

2.2 Annotation generation procedure

Before being able to predict labels of the original message sentences, it is necessary to identify those that are re-used in a reply message. Identification of the quoted lines in a reply message is not sufficient for various reasons. First, the segmenter is intended to work on non-noisy data (i.e. the new content parts in the messages) while a quoted message is an altered version of the original one. Indeed, some email software clients involved in the discussion are not always standards-compliant and totally compatible[3]. In particular, the quoted parts can be wrongly re-encoded at each exchange step due to the absence of dedicated header information. In addition, the client programs can integrate their own mechanisms for quoting the previous messages when including them as well as for wrapping too long lines[4]. Second, accessing the original message may allow taking some contextual features into consideration (like the visual layout for example). Third, to go further, the original context of the extracted text also conveys some segmentation information. For instance, a sentence from the original message, not present in the reply, but following an aligned sentence, can be considered as starting a segment.

So in addition to identifying the quoted lines, we deploy an alignment procedure to get the original version of the quoted text. In this paper, we do not consider the contextual features from the original message and focus only on sentences that have been aligned.

The generation procedure is intended to "automatically" annotate sentences from the original messages with segmentation information. The procedure follows the following steps:

1. Messages posted in the interleaved replying style are identified
2. For each pair of original and reply messages:
   (a) Both messages are tokenized at sentence and at word levels
   (b) Quoted lines in the reply message are identified
   (c) Sentences which are part of the quoted text in the reply message are identified
   (d) Sentences in the original message are aligned with quoted text in the reply message [5]
   (e) Aligned original sentences are labelled in terms of position in segment

[3] The *Request for Comments* (RFC) are guidelines and protocols proposed by working groups involved in the Internet Standardization `https://tools.ietf.org/html`, the message contents suffer from encoding and decoding problems. Some of the RFC are dedicated to email format and encoding specifications (See RFC 2822 and 5335 as starting points). There have been several propositions with updates and consequently obsoleted versions which may explain some alteration issues.

[4] Feature for making the text readable without any horizontal scrolling by splitting lines into pieces of about 80 characters.

[5] Section 2.3 details how alignment is performed.

(f) The sequence of labelled sentences is added to the training data

Messages with *inline replying* are recognized thanks to the presence of at least two consecutive quoted lines separated by new content lines. Pairs of original and reply messages are constituted based on the `in-reply-to` field present in the email headers. As declared in the RFC 3676[6], we consider as *quoted lines*, the lines beginning with the ">" (greater than) sign. Lines which are not quoted lines are considered to be *new content* lines. The word tokens are used to index the quoted lines and the sentences.

Labelling of aligned sentence (sentence from the original message re-used in the reply message) uses this simple rule-based algorithm:

For each aligned original sentence:
if the sentence is surrounded by new content in the reply message, the label is `Start&End`
else if the sentence is preceded by a new content, the label is `Start`
else if the sentence is followed by a new content, the label is `End`
else, the label is `Inside`

2.3 Alignment module

For finding alignments between two given text messages, we use a *dynamic programming (DP) string alignment algorithm* (Sankoff and Kruskal, 1983). In the context of speech recognition, the algorithm is also known as the *NIST align/scoring algorithm*. Indeed, it is widely used to evaluate the output of speech recognition systems by comparing the hypothesized text output by the speech recognizer to the correct, or reference text. The algorithm works by "performing a global minimization of a Levenshtein distance function which weights the cost of correct words, insertions, deletions and substitutions as 0, 75, 75 and 100 respectively. The computational complexity of DP is $O(MN)$."

The Carnegie Mellon University provides an implementation of the algorithm in its speech recognition toolkit[7]. We use an adaptation of it which allows working on lists of strings[8] rather than directly on strings (as sequences of characters).

3 Building the segmenter

Each email is processed as a sequence of sentences. We choose to define the segmentation problem as a sequence labelling task whose aim is to assign the globally best set of labels for the entire sequence at once. The underlying idea is that the choice of the optimal label for a given sentence is dependent on the choices of nearby sentences. Our email segmenter is built around a linear-chain Conditional Random Field (CRF), as implemented in the sequence labelling toolkit Wapiti (Lavergne et al., 2010).

Training the classifier to recognize the different labels of the previously defined annotation scheme can be problematic. It has indeed some disadvantages that can undermine the effectiveness of the classifier. In particular, sentences annotated *SE* will, by definition, share important characteristics with sentences bearing the annotation *S* and *E*. So we chose to transform these annotations into a binary scheme and merely differentiate sentences that starts a new segment (*True*), or "boundary sentences", from those that do not (*False*). The conversion process is trivial, and can easily be reversed[9].

We distinguish four sets of features: n-gram features, information structure based features, thematic features and miscellaneous features. All the features are domain-independant. Almost all features are language-independant as well, save for a few that can be easily translated. For our experiments, the CRF window size is set at 5, i.e. the classification algorithm takes into account features of the next and previous two sentences as well as the current one.

[6] `http://www.ietf.org/rfc/rfc3676.txt`

[7] Sphinx 4 `edu.cmu.sphinx.util.NISTAlign` `http://cmusphinx.sourceforge.net`

[8] `https://github.com/romanows/WordSequenceAligner`

[9] Sentences labelled with *SE* or *S* are turned into *True*, the other ones into *False*. To reverse the process, a *True* is turned into *SE* if the next sentence is also a boundary (i.e. a True) and into *S* otherwise. While a *False* is turned into *E* if the next sentence is a boundary (i.e. a True) and into *I* otherwise.

n-gram features We select the case-insensitive word bigrams and trigrams with the highest document frequency in the training data (empirically we select the top 1,000 n-grams), and check for their presence in each sentence. Since the probability of having multiple occurrences of the same n-gram in one sentence are extremely low, we do not record the number of occurrences but merely a boolean value.

Information structure based features This feature set is inspired by the information structure theory (Kruijff-Korbayová and Kruijff, 1996) which describes the information imparted by the sentence in terms of the way it is related to prior context. The theory relates these functions with particular syntactic constructions (e.g. topicalization) and word order constraints in the sentence.

We focus on the first and last three *significant* tokens in the sentence. A token is considered as significant if its occurrence frequency is higher than 1/2,000[10]. As features we use n-grams of the surface form, lemma and part-of-speech tag of each triplet (36 features).

Thematic feature The only feature we use to account for thematic shift recognition is the output of the TextTiling algorithm (Hearst, 1997). TextTiling is one of the most commonly used algorithms for automatic text segmentation. If the algorithm detects a rupture in the lexical cohesion of the text (between two consecutive blocks), it will place a boundary to indicate a thematic change. Due to the short size of the messages, we define a block size to equate the sum of three times the sentence average size in our corpus. We set the step-size (overlap size of the rolling window) to the average size of a sentence.

Miscellaneous features This feature set includes stylistic and semantic features. 24 features, several of them borrowed from related work in speech act classification (Qadir and Riloff, 2011) and email segmentation (Lampert et al., 2009b), are in the set: *Stylistic features* capture information about the visual structure and composition of the message: the position of the sentence in the email, the average length of a token, the total number of tokens and characters, the proportion of upper-case, alphabetic and numeric characters, the number of greater-than signs (“>”); whether the sentence ends with or contains a question mark, a colon or a semicolon; whether the sentence contains any punctuation within the first three tokens (this is meant to recognize greetings (Qadir and Riloff, 2011)).

Semantic features check for meaningful words and phrases: whether the sentence begins with or contains a “wh*” question word or a phrase suggesting an incoming interrogation (e.g. *“is it”*, *“are there”*); whether the sentence contains a modal; whether any plan phrases (e.g. *“i will”*, *“we are going to”*) are present; whether the sentence contains first person (e.g. *“we”*, *“my”*) second person or third person words; the first personal pronoun found in the sentence; the first verbal form found.

4 Experimental framework

We describe the data, the preprocessing and the evaluation protocol we use for our experiments.

4.1 Corpus

The current work takes place in a project dealing with multilingual and multimodal discussion processing, mainly in interrogative technical domains. For these reasons we did not consider the Enron Corpus (30,000 threads) (Klimt and Yang, 2004) (which is from a corporate environment), neither the W3C Corpus (despite its technical consistence) or its subset, the British Columbia Conversation Corpus (BC3) (Ulrich et al., 2008).

We rather use the *ubuntu-users* email archive[11] as our primary corpus. It offers a number of advantages. It is free, and distributed under an unrestrictive license. It increases continuously, and therefore is representative of modern emailing in both content and formatting. Additionally, many alternatives archives are available, in a number of different languages, including some very resource-poor languages. Ubuntu also offers a forum and a FAQ which are interesting in the context of multimodal studies.

We use a copy of December 2013. The corpus contains a total of 272,380 messages (47,044 threads). 33,915 of them are posted in the inline replying style that we are interested in. These messages are made

[10] This value was set up empirically on our data. More experimentation needs to be done to generalize it.

[11] Ubuntu mailing lists archives (See *ubuntu-users*): `https://lists.ubuntu.com/archives/`

of 418,858 sentences, themselves constituted of 76,326 unique tokens (5,139,123 total). 87,950 of these lines (21%) are automatically labelled by our system as the start of a new segment (either *SE* or *S*).

4.2 Evaluation protocol

In order to evaluate the efficiency of the segmenter, we perform a 10-fold cross-validation on the Ubuntu corpus, and compare its performance to two different baselines. The first one, the "regular" baseline, is computed by segmenting the test set into regular segments of the same length as the average training set segment length, rounded up. The second one is the TextTiling algorithm we described in section 3. While it is used as a feature in the proposed approach in the previous section, the direct output of the TextTiling algorithm is used for the baseline.

The results are measured with a panel of metrics used in text segmentation and Information Retrieval (IR). Precision (P) and Recall (R) are provided for all results. P is the percentage of boundaries identified by the classifier that are indeed true boundaries. R is the percentage of true boundaries that are identified by the classifier. We also provide the harmonic mean of precision and recall: $F_1 = 2 \cdot \frac{P \cdot R}{P+R}$

However, automatic evaluation of speech segmentation through these metrics is problematic as predicted segment boundaries seldom align precisely. Therefore, we also provide an array of metrics relevant to the field of text segmentation : P_k, *WindowDiff* and the *Generalized Hamming Distance (GHD)*. The P_k metric is a probabilistically motivated error metric for the assessment of segmentation algorithms (Beeferman et al., 1999). *WindowDiff* compares the number of segment boundaries found within a fixed-sized window to the number of boundaries found in the same window of text for the reference segmentation (Pevzner and Hearst, 2002). The *GHD* is an extension of the Hamming distance[12] that gives partial credit for near misses (Bookstein et al., 2002).

4.3 Preprocessing

To reduce noise in the corpus we filter out undesirable emails based on several criteria, the first of which is encoding. Messages that are not UTF-8 encoded are removed from the selection. The second criterion is MIME type: we keep single-part plain text messages only, and remove those with HTML or other special contents. In addition, we choose to consider only replies to thread starters. This choice is based on the assumption that the alignment module would have more difficulty in recognizing properly sentences that were repeatedly transformed in successive replies. Indeed, these replies - that would contain quoted text from other messages - would be more likely to be poorly labelled through automatic annotation. The last criterion is length. The dataset being built from a mailing list that can cover very technical discussions, users sometimes send very lengthy messages containing many lines of copied-and-pasted code, software logs, bash command outputs, etc. The number of these messages is marginal, but their lengths being disproportionately high, they can have a negative impact on the segmenter's performance. We therefore exclude messages longer than the average message length plus the standard length deviation. After filtering, the dataset is left with 6,821 messages out of 33,915 (20%).

For building the segmenter features, we use the Stanford Part-Of-Speech Tagger for morpho-syntactic tagging (Toutanova et al., 2003), and the WordNet lexical database for lemmatization (Miller, 1995).

5 Experiments

Table 1 shows the summary of all obtained results. On the left side are shown results about segmentation metrics, on the right side results about information retrieval metrics. First, we examine baseline scores, and display them in the top section. Second, in the middle section, we show results for segmenters based on individual feature sets (with A standing for n-grams, B for information structure, C for TextTiling and D for miscellaneous features). Finally, in the lower section, we show results based on feature sets combinations.

[12] Wikipedia article on the Hamming distance: `http://en.wikipedia.org/wiki/Hamming_distance`

	Segmentation metrics			Information Retrieval metrics		
	WD	P_k	GHD	P	R	F_1
regular baseline	.59	.25	.60	.31	.49	.38
TextTiling baseline	.41	.07	.38	.75	.44	.56
$\phi(A)$ with A = n-grams	.38	**.05**	.39	**1**	.39	.56
$\phi(B)$ with B = info. structure	.43	.11	.38	.60	.68	**.64**
$\phi(C)$ with C = TextTiling	.39	.05	.38	.94	.40	.56
$\phi(D)$ with D = misc. features	.41	.09	.38	.69	.49	.57
$\phi(A+B+C+D)$	.38	**.05**	.39	**1**	.39	.56
$\phi(\phi(A)+\phi(B)+\phi(C)+\phi(D))$	.38	.06	.36	.81	.47	.59
$\phi(A) \cup \phi(B+C+D)$	.45	.12	.40	.58	**.69**	.63
$\phi(A) \cup \delta(\phi(B+C+D))$	**.36**	.06	**.34**	.80	.53	**.64**

Table 1: Comparative results between baselines and tested segmenters. All displayed results show *WindowDiff* (*WD*), P_k and *GHD* as error rates, therefore a lower score is desirable for these metrics. This contrasts with the three IR scores, for which a low value denotes poor performance. Best scores are shown bolded.

5.1 Baseline segmenters

The first section of Table 1 shows the results obtained by both of our baselines. Unsurprisingly, TextTiling performs much better than the basic regular segmentation algorithm across all metrics save recall.

5.2 Segmenters based on individual feature sets

The second section of Table 1 shows the results for four different classifiers, each trained with a distinct subset of the feature set. The ϕ function is the classification function, its parameters are features, and its output a prediction. While all classifiers easily beat the regular baseline, and match the TextTiling baseline when it comes to IR metrics, only the thematic and the n-grams segmenters manage to surpass TextTiling when performance is measured by segmentation metrics. In terms of IR scores, the n-grams classifier in particular stands out as it manages to achieve an outstanding 100% precision, although this result is mitigated by a meager 39% recall. It is also interesting to see that the thematic classifier, based only on contextual information about TextTiling output, performs better than the TextTiling baseline.

5.3 Segmenters based on feature sets combinations

The last section of Table 1 shows the results of four different segmenters. The first one, $\phi(A+B+C+D)$, is a simple classifier that takes all available features into account. Its results are exactly identical to that of the n-grams classifier, most certainly due to the fact that other features are filtered out due to the sheer number of lexical features. The second one, $\phi(\phi(A)+\phi(B)+\phi(C)+\phi(D))$, uses as features the outputs of the four classifiers trained on each individual feature set. Results show this approach isn't significantly better. The third one, $\phi(A) \cup \phi(B+C+D)$, segments according to the union of the boundaries detected by a classifier trained on n-grams features and those identified by a classifier trained on all other features. This idea is motivated by the fact that we know all boundaries found by the n-grams classifier to be accurate ($P = 1$). Doing this allows the segmenter to obtain the best possible recall ($R = .69$), but at the expense of precision ($P = .58$). The last one, $\phi(A) \cup \delta(\phi(B+C+D))$, attempts to increase the n-grams classifier's recall without sacrificing too much precision by being more selective about boundaries. The δ function is the "cherry picking" function, which filters out boundaries predicted without sufficient confidence. Only those identified by the n-grams classifier and those classified as boundaries with a confidence score of at least .99 by a classifier trained on the other feature sets are considered. This system outperforms all others both in terms of segmentation scores and F_1, however it is still relatively conservative and the segmentation ratio (the number of true boundaries divided by the number of guessed boundaries) remains significantly lower than expected, at 0.67. Tuning the minimum

confidence score (c) allows to adjust P from .58 ($c = 0$) to 1 ($c = 1$) and R from .39 ($c = 1$) to .69 ($c = 0$).

6 Related work

Three research areas are directly related to our study: a) collaborative approaches for acquiring annotated corpora, b) detection of email structure, and c) sentence alignment. In the (Wang et al., 2013)'s taxonomy of the collaborative approaches for acquiring annotated corpora, our approach could be related to the *Wisdom of the Crowds* (WotC) genre where motivators are altruism or prestige to collaborate for the building of a public resource. As a major difference, we did not initiate the annotation process and consequently we did not define annotation guidelines, design tasks or develop tools for annotating which are always problematic questions. We have just rerouted *a posteriori* the result of an existing task which was performed in a distinct context. In our case the burning issue is to determine the adequacy of our segmentation task. Our work is motivated by the need to identify important snippets of information in messages for applications such as being able to determine whether all the aspects of a customer request were fully considered. We argue that even if it is not always obvious to tag topically or rhetorically a segment, the fact that it was a human who actually segmented the message ensures its quality. We think that our approach can also be used for determining the relevance of the segments, however it has some limits, and we do not know how labelling segments with dialogue acts may help us do so.

Detecting the structure of a thread is a hot topic. As mentioned in Section 1, very little works have been done on email segmentation. We are aware of recent works in linear text segmentation such as (Kazantseva and Szpakowicz, 2011) who addresses the problem by modelling the text as a graph of sentences and by performing clustering and/or cut methods. Due to the size of the messages (and consequently the available lexical material), it is not always possible to exploit this kind of method. However, our results tend to indicate that we should investigate in this direction nonetheless. By detecting sub-units of information within the message, our work may complement the works of (Wang et al., 2011; Kim et al., 2010) who propose solutions for detecting links between messages. We may extend these approaches by considering the possibility of pointing from/to multiple message sources/targets.

Concerning the alignment process, our task can be compared to the detection of monolingual text derivation (otherwise called plagiarism, near–duplication, revision). (Poulard et al., 2011) compare, for instance, the use of n–grams overlap with the use of text hapax. In constrast, we already know that a text (the reply message) derives from another (the original message). Sentence alignment has also been a very active field of research in statistical machine translation for building parallel corpora. Some methods are based on sentence length comparison (Gale and Church, 1991), some methods rely on the overlap of rare words (cognates and named entities) (Enright and Kondrak, 2007). In comparison, in our task, despite some noise, the compared text includes large parts of material identical to the original text. The kinds of edit operation in presence (no inversion[13] only deletion, insertion and substitution) lead us to consider the Levenshtein distance as a serious option.

7 Future work

The main contribution of this work is to exploit the human effort dedicated to reply formatting for training discursive email segmenters. We have implemented and tested various segmenter models. There is still room for improvement, but our results indicate that the approach merits more thorough examination. Our segmentation approach remains relatively simple and can be easily extended. One way would be to consider contextual features in order to characterize the sentences in the original message structure. As future works, we plan to complete our current experiments with two new approaches for evaluation. The first one will consists in comparing the automatic segmentation with those performed by human annotators. This task remains tedious since it will then be necessary to define an annotation protocol, write guidelines and build other resources. The second evaluation we plan to perform is an extrinsic evaluation. The idea will be to measure the contribution of the segmentation in the process of detecting the dialogue acts, i.e. to check if existing sentence-level classification systems would perform better with such contextual information.

[13] When computing the Levenshtein distance, the inversion edit operation is the most costly operation.

References

[Austin1970] John L. Austin. 1970. *Quand dire c'est faire*. Éditions du Seuil.

[Beeferman et al.1999] Doug Beeferman, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine learning*, 34(1-3):177–210.

[Bookstein et al.2002] Abraham Bookstein, Vladimir A Kulyukin, and Timo Raita. 2002. Generalized hamming distance. *Information Retrieval*, 5(4):353–375.

[Enright and Kondrak2007] Jessica Enright and Grzegorz Kondrak. 2007. A fast method for parallel document identification. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 29–32, Rochester, New York, April. Association for Computational Linguistics.

[Gale and Church1991] William A. Gale and Kenneth Ward Church. 1991. A program for aligning sentences in bilingual corpora. In *Proceedings of the 29th Annual Meeting of the Association of Computational Linguistics (ACL)*.

[Hearst1997] Marti A Hearst. 1997. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, 23(1):33–64.

[Joty et al.2013] Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2013. Topic segmentation and labeling in asynchronous conversations. *Journal of AI Research (JAIR)*, 47:521–573.

[Kazantseva and Szpakowicz2011] Anna Kazantseva and Stan Szpakowicz. 2011. Linear text segmentation using affinity propagation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 284–293, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Kim et al.2010] Su Nam Kim, Li Wang, and Timothy Baldwin. 2010. Tagging and linking web forum posts. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, CoNLL '10, pages 192–202, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Klimt and Yang2004] Bryan Klimt and Yiming Yang. 2004. The enron corpus: A new dataset for email classification research. In Jean-François Boulicaut, Floriana Esposito, Fosca Giannotti, and Dino Pedreschi, editors, *ECML*, volume 3201 of *Lecture Notes in Computer Science*, pages 217–226. Springer.

[Kruijff-Korbayová and Kruijff1996] Ivana Kruijff-Korbayová and Geert-Jan M. Kruijff. 1996. Identification of topic-focus chains. In S. Botley, J. Glass, T. McEnery, and A. Wilson, editors, *Approaches to Discourse Anaphora: Proceedings of the Discourse Anaphora and Anaphora Resolution Colloquium (DAARC96)*, volume 8, pages 165–179. University Centre for Computer Corpus Research on Language, University of Lancaster, UK, July 17-18.

[Lampert et al.2009a] Andrew Lampert, Robert Dale, and Cécile Paris. 2009a. Segmenting email message text into zones. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2 - Volume 2*, EMNLP '09, pages 919–928, Stroudsburg, PA, USA. Association for Computational Linguistics.

[Lampert et al.2009b] Andrew Lampert, Robert Dale, and Cécile Paris. 2009b. Segmenting email message text into zones. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 919–928. Association for Computational Linguistics.

[Lavergne et al.2010] Thomas Lavergne, Olivier Cappé, and François Yvon. 2010. Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 504–513. Association for Computational Linguistics, July.

[Miller1995] George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

[Pevzner and Hearst2002] Lev Pevzner and Marti A Hearst. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational Linguistics*, 28(1):19–36.

[Poulard et al.2011] Fabien Poulard, Nicolas Hernandez, and Béatrice Daille. 2011. Detecting derivatives using specific and invariant descriptors. *Polibits*, (43):7–13.

[Qadir and Riloff2011] Ashequl Qadir and Ellen Riloff. 2011. Classifying sentences as speech acts in message board posts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 748–758. Association for Computational Linguistics.

[Ratinov and Roth2009] L. Ratinov and D. Roth. 2009. Design challenges and misconceptions in named entity recognition. In *CoNLL*, 6.

[Sankoff and Kruskal1983] D Sankoff and J B Kruskal. 1983. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts. ISBN 0-201-07809-0.

[Tavafi et al.2013] Maryam Tavafi, Yashar Mehdad, Shafiq Joty, Giuseppe Carenini, and Raymond Ng. 2013. Dialogue act recognition in synchronous and asynchronous conversations. In *Proceedings of the 14th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL 2013)*, SIGDIAL'13.

[Toutanova et al.2003] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for Computational Linguistics.

[Ulrich et al.2008] J. Ulrich, G. Murray, and G. Carenini. 2008. A publicly available annotated corpus for supervised email summarization. In *AAAI08 EMAIL Workshop*, Chicago, USA. AAAI.

[von Ahn2006] L. von Ahn. 2006. Games with a purpose. *Computer*, 39(6):92–94.

[Wang et al.2011] Li Wang, Diana Mccarthy, and Timothy Baldwin. 2011. Predicting thread linking structure by lexical chaining. In *Proceedings of the Australasian Language Technology Association Workshop 2011*, pages 76–85, Canberra, Australia, December.

[Wang et al.2013] Aobo Wang, CongDuyVu Hoang, and Min-Yen Kan. 2013. Perspectives on crowdsourcing annotations for natural language processing. *Language Resources and Evaluation*, 47(1):9–31.

Bibliographie

[Austin, 1975] Austin, John Langshaw. 1975. *How to do things with words*. Vol. 1955. Oxford university press.

[Beeferman *et al.* , 1999] Beeferman, Doug, Berger, Adam, & Lafferty, John. 1999. Statistical models for text segmentation. *Machine learning*, **34**(1-3), 177–210.

[Bookstein *et al.* , 2002] Bookstein, Abraham, Kulyukin, Vladimir A, & Raita, Timo. 2002. Generalized hamming distance. *Information retrieval*, **5**(4), 353–375.

[De Felice *et al.* , 2013] De Felice, Rachele, Darby, Jeannique, Fisher, Anthony, & Peplow, David. 2013. A classification scheme for annotating speech acts in a business email corpus. *Icame journal*, **37**, 71–105.

[Fort *et al.* , 2011] Fort, Karën, Adda, Gilles, & Cohen, K. Bretonnel. 2011. Amazon mechanical turk : Gold mine or coal mine ? *Computational linguistics*, **37**(2), 413–420.

[Hearst, 1997] Hearst, Marti A. 1997. Texttiling : Segmenting text into multi-paragraph subtopic passages. *Computational linguistics*, **23**(1), 33–64.

[Joty *et al.* , 2011] Joty, Shafiq, Carenini, Giuseppe, & Lin, Chin-Yew. 2011. Unsupervised modeling of dialog acts in asynchronous conversations. *Pages 1807–1813 of : Proceedings of the twenty-second international joint conference on artificial intelligence - volume volume three*. IJCAI'11. AAAI Press.

[Joty *et al.* , 2013] Joty, Shafiq, Carenini, Giuseppe, & Ng, Raymond. 2013. Topic segmentation and labeling in asynchronous conversations. *Journal of ai research (jair)*, **47**, 521–573.

[Kazantseva & Szpakowicz, 2011] Kazantseva, Anna, & Szpakowicz, Stan. 2011. Linear text segmentation using affinity propagation. *Pages 284–293 of : Proceedings of the conference*

on empirical methods in natural language processing. EMNLP '11. Stroudsburg, PA, USA : Association for Computational Linguistics.

[Klimt & Yang, 2004a] Klimt, Bryan, & Yang, Yiming. 2004a. The enron corpus : A new dataset for email classification research. *Pages 217–226 of :* Boulicaut, Jean-François, Esposito, Floriana, Giannotti, Fosca, & Pedreschi, Dino (eds), *Ecml*. Lecture Notes in Computer Science, vol. 3201. Springer.

[Klimt & Yang, 2004b] Klimt, Bryan, & Yang, Yiming. 2004b. Introducing the enron corpus. *In : Ceas*.

[Kruijff-Korbayová & Kruijff, 1996] Kruijff-Korbayová, Ivana, & Kruijff, Geert-Jan M. 1996. Identification of topic-focus chains. *Pages 165–179 of :* Botley, S., Glass, J., McEnery, T., & Wilson, A. (eds), *Approaches to discourse anaphora : Proceedings of the discourse anaphora and anaphora resolution colloquium (daarc96)*, vol. 8.

[Lampert *et al.* , 2006] Lampert, Andrew, Dale, Robert, & Paris, Cécile. 2006. *Classifying speech acts using verbal response modes*.

[Lampert *et al.* , 2009a] Lampert, Andrew, Dale, Robert, & Paris, Cécile. 2009a. Segmenting email message text into zones. *Pages 919–928 of : Proceedings of the 2009 conference on empirical methods in natural language processing : Volume 2 - volume 2*. EMNLP '09. Stroudsburg, PA, USA : Association for Computational Linguistics.

[Lampert *et al.* , 2009b] Lampert, Andrew, Dale, Robert, & Paris, Cécile. 2009b. Segmenting email message text into zones. *Pages 919–928 of : Proceedings of the 2009 conference on empirical methods in natural language processing : Volume 2-volume 2*. Association for Computational Linguistics.

[Lavergne *et al.* , 2010] Lavergne, Thomas, Cappé, Olivier, & Yvon, François. 2010. Practical very large scale CRFs. *Pages 504–513 of : Proceedings the 48th annual meeting of the association for computational linguistics (ACL)*. Association for Computational Linguistics.

[Levenshtein, 1966] Levenshtein, Vladimir I. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Page 707 of : Soviet physics doklady*, vol. 10.

[Miller, 1995] Miller, George A. 1995. Wordnet : a lexical database for english. *Communications of the acm*, **38**(11), 39–41.

[Pevzner & Hearst, 2002] Pevzner, Lev, & Hearst, Marti A. 2002. A critique and improvement of an evaluation metric for text segmentation. *Computational linguistics*, **28**(1), 19–36.

[Qadir & Riloff, 2011] Qadir, Ashequl, & Riloff, Ellen. 2011. Classifying sentences as speech acts in message board posts. *Pages 748–758 of : Proceedings of the conference on empirical methods in natural language processing*. Association for Computational Linguistics.

[Ratinov & Roth, 2009] Ratinov, L., & Roth, D. 2009 (6). Design challenges and misconceptions in named entity recognition. *In : Conll*.

[Sankoff & Kruskal, 1983] Sankoff, D, & Kruskal, J B. 1983. *Time warps, string edits, and macromolecules : The theory and practice of sequence comparison*. Reading, Massachusetts : Addison-Wesley Publishing Company, Inc. ISBN 0-201-07809-0.

[Schmid, 1994] Schmid, Helmut. 1994. Probabilistic part-of-speech tagging using decision trees. *Pages 44–49 of : International conference on new methods in language processing*.

[Searle, 1976] Searle, John R. 1976. *A taxonomy of illocutionary acts*. Linguistic Agency University of Trier.

[Tavafi *et al.* , 2013] Tavafi, Maryam, Mehdad, Yashar, Joty, Shafiq, Carenini, Giuseppe, & Ng, Raymond. 2013. Dialogue act recognition in synchronous and asynchronous conversations. *In : Proceedings of the 14th annual meeting of the special interest group on discourse and dialogue (sigdial 2013)*. SIGDIAL'13.

[Toutanova *et al.* , 2003] Toutanova, Kristina, Klein, Dan, Manning, Christopher D, & Singer, Yoram. 2003. Feature-rich part-of-speech tagging with a cyclic dependency network. *Pages 173–180 of : Proceedings of the 2003 conference of the north american chapter of the association for computational linguistics on human language technology-volume 1*. Association for Computational Linguistics.

[Traum, 2000] Traum, David R. 2000. 20 questions on dialogue act taxonomies. *Journal of semantics*, **17**(1), 7–30.

[Ulrich *et al.* , 2008a] Ulrich, J., Murray, G., & Carenini, G. 2008a. A publicly available annotated corpus for supervised email summarization. *In : Aaai08 email workshop*. Chicago, USA : AAAI.

[Ulrich *et al.* , 2008b] Ulrich, J., Murray, G., & Carenini, G. 2008b. A publicly available annotated corpus for supervised email summarization. *In : Aaai08 email workshop*. Chicago, USA : AAAI.

[von Ahn, 2006] von Ahn, L. 2006. Games with a purpose. *Computer*, **39**(6), 92–94.

[Wang *et al.* , 2013] Wang, Aobo, Hoang, CongDuyVu, & Kan, Min-Yen. 2013. Perspectives on crowdsourcing annotations for natural language processing. *Language resources and evaluation*, **47**(1), 9–31.