Détection automatique d'opinions dans des corpus de messages oraux

Nathalie Camelin[1]*, Géraldine Damnati*[2]*, Frédéric Béchet*[1]*, Renato De Mori*[1] *

[1] LIA/CNRS - University of Avignon, BP1228 84911 Avignon cedex 09 France
[2] France Télécom R&D - TECH/SSTP/RVA 2 av. Pierre Marzin 22307 Lannion Cedex 07, France
{nathalie.camelin,frederic.bechet,renato.demori}@univ-avignon.fr
geraldine.damnati@francetelecom.com

ABSTRACT

Telephone surveys are often used by Customer Services to evaluate their clients' satisfaction and to improve their services. Large amounts of data are collected to observe the evolution of customers' opinions. Within this context, the automatization of the process of these databases becomes a crucial issue. This paper addresses the automatic analysis of audio messages where customers are asked to give their opinion over several dimensions about a Customer Service. Interpretation methods that integrate automatically and manually acquired knowledge are proposed. Experimental results, done on a database collected from a deployed Customer Service in real conditions with real customers are given.

1. INTRODUCTION

La détection d'opinions ou encore d'assertions objectives ou subjectives dans un texte est un domaine de recherche en pleine expansion [5, 1]. Du point de vue des utilisateurs, les deux principales applications de ce type de détection concerne d'une part l'analyse automatique d'opinions dans des messages contenant l'avis de consommateurs sur un produit ou un phénomène particulier [3] et d'autre part l'analyse de la subjectivité d'une phrase pour les systèmes de résumé automatique ou de question/réponse [4]. D'un point de vue scientifique, la problématique posée par la détection d'opinions se situe dans le cadre de la compréhension automatique de messages. Au niveau sémantique, ce problème constitue une possibilité d'aborder un niveau intermédiaire entre la simple détection des entités présentes et l'analyse sémantique complète du message, qui n'est pas envisageable sur des messages complexes.

La principale originalité de ce travail réside dans le type de message traité : nous abordons ici le problème de la détection d'opinions dans des messages oraux collectés auprès de *vrais* utilisateurs. Cette étude nous permettra ainsi, d'une part de tester la robustesse des processus de détection d'opinions aux erreurs de reconnaissance et aux disfluences ; et d'autre part de développer un module de transcription automatique de parole spécifique à ce type de corpus particulièrement difficile.

2. DESCRIPTION DU CORPUS

Les personnes sont invitées par un court message à appeler un numéro gratuit qui leur permet d'exprimer leur satisfaction vis à vis du service-client qu'ils ont récemment appelé. En composant ce numéro, le message vocal suivant les invite à laisser un message : *[. . .] Vous avez récemment contacté notre service clientèle. Nous souhaitons nous assurer que vous avez été satisfait de l'accueil et de la suite donnée à votre appel. Vous pourrez me laisser votre réponse après le top sonore. [. . .]*

Du fait que les messages ont été enregistrés à l'origine dans l'optique d'un traitement par opérateur, aucune consigne de nature à faciliter le traitement automatique n'a été donnée : pas de conseils sur le mode d'élocution, question ouverte et même incitation à laisser des commentaires. Ainsi, les messages recueillis sont *réalistes* et de longueur variable (d'une dizaine à plusieurs centaines de mots).Pour cette étude un ensemble de 1779 messages, collectés sur une période de 3 mois, a été transcrit manuellement au niveau mots, opinions et marqueurs (indication de disfluence et marqueurs discursifs).

Au niveau des opinions, quatre critères ont été retenus : l'*accueil*, l'*attente*, l'*efficacité* et un dernier critère regroupant le reste des critères évoqués par le locuteur : *autre*. Ces critères s'expriment chacun selon deux polarités différentes : *plus* ou *moins*. Cela fait un total de 8 étiquettes sémantiques ou *concepts*. Dans la transcription manuelle, au sein de chaque message, l'expression d'une opinion sur l'un de ces critères est indiqué par des balises. Nous disposons ainsi d'un corpus de 1079 segments, chacun porteur d'une ou plusieurs opinions particulières. Le but du traitement automatique est de retrouver ces segments et de les étiqueter avec la ou les opinions appropriées.

Nb concept par message	Répartition (% corpus)	Taille moyenne (nb mots)
0	19.2	61.0
1	51.3	40.3
2 et plus	29.5	60.8

TAB. 1: Répartition des messages dans le corpus en fonction du nombre de concepts exprimés

La taille moyenne des messages en fonction du nombre d'opinions exprimées est présentée dans le tableau 1. Même si les messages exprimant une seule opinion sont les plus courts, on voit qu'un message long n'est pas forcément le signe d'un nombre plus grand d'opinions exprimées, notamment les messages constitués uniquement de digressions et porteur d'aucune opinion sont en moyenne les plus longs. Un autre problème que pose ces messages est qu'un même concept peut être vu plusieurs fois dans un message avec des opinions contraires. Cela se ren-

*Travaux réalisés en collaboration avec France Télécom's R&D - contrat 021B178

contre quand la personne n'est pas entièrement satisfaite (*e.g.* :satisfaite du service-client mais pas du résultat) ou qu'une notion temporelle rentre en jeu dans son discours. Un exemple de message est donné dans le tableau 2.

oui c'est monsieur NOMS PRENOMS j'avais appelé donc le service client ouais j'ai été très bien accueilli des bons renseignements sauf que ça ne fonctionne toujours pas donc je sais pas si j'ai fait une mauvaise manipulation ou y a un problème enfin voilà sinon l'accueil était et les conseils très judicieux même si le résultat n'est pas n'est pas là merci au revoir

TAB. 2: Exemple de message contenant plusieurs opinions

3. MODÈLES DE LANGAGE SPÉCIFIQUES AUX OPINIONS ET SEGMENTATION AUTOMATIQUE

Du fait du degré de liberté laissé aux utilisateurs dans l'énoncé de leur message, on observe une assez grande dispersion dans la distribution des fréquences des mots. Ceci est d'autant plus le cas dans les portions des messages où les utilisateurs relatent l'origine de leur problème qui peut être de nature assez variée. Une fois les noms propres filtrés, le corpus d'apprentissage dans son ensemble contient 2981 mots différents pour un nombre total 51056 occurrences. Près de la moitié des mots n'apparaissent qu'une seule fois dans le corpus d'apprentissage, et la restriction du lexique aux mots d'occurrence supérieure ou égale à 2, conduit à un lexique de 1564 mots pour un taux de mots hors-vocabulaire égal à 2,8%. Un premier modèle de type bigram a donc été construit sur la base de ce lexique réduit. Aux mots du vocabulaire s'ajoutent des éléments spécifiques aux données, tels qu'un modèle de rejet particulier pour les noms propres ou encore une grammaire de numéros de téléphones. Ces éléments sont intégrés au modèle bigram.

Parallèlement, une première tentative de segmentation automatique a été réalisée, avec pour objectif de proposer un découpage des messages pour faciliter la tâche de classification en aval. L'idée est d'évaluer l'apport d'une segmentation a priori et non supervisée des messages en utilisant un automate bruit/parole pour détecter automatiquement les pauses réalisées par les locuteurs. Même s'il n'y a pas a priori de corrélation entre la présence de pauses et le changement de thématique, cette première approche a le mérite d'être simple à mettre en oeuvre et servira de baseline pour la suite de l'étude. Les segments isolés par l'automate bruit/parole sont soumis indépendamment les uns des autres au système de reconnaissance et les hypothèses de reconnaissance associées sont transmises aux modules de classification. Ce modèle portera le nom de *RECO1* dans la section 5.

Cette segmentation s'est avérée insuffisante. En effet, il subsiste d'une part des segments assez longs et porteurs de plusieurs expressions d'opinion (enchaînés sans pause). Il arrive d'autre part que des portions porteuses d'opinion soient tronquées par la segmentation automatique (si l'utilisateur hésite par exemple alors qu'il exprime une opinion). Du fait du nombre très important de disfluence au sein des messages, mais aussi souvent de la mauvaise qualité acoustique des messages, le taux d'erreur mot moyen obtenu avec ce modèle sur l'ensemble du corpus est de 58%. Ce taux très important est à relativiser car il inclus toutes les répétitions, hésitations et digressions effectuées par les utilisateurs.

Dans un deuxième temps, les problématiques de segmentation et de reconnaissance ont été intégrées à travers un nouveau type de modèle de langage. L'idée est de ne modéliser explicitement que les portions de messages porteuses d'opinion. Pour cela, un sous-corpus a été extrait pour chaque étiquette qui regroupe l'ensemble des segments associés à cette étiquette dans le corpus d'apprentissage initial. Un sous-modèle bigram a ainsi été estimé pour chaque étiquette à partir du sous-corpus associé. Par ailleurs, un modèle englobant de type bigram portant sur les étiquettes elles-mêmes a été estimé pour modéliser les enchaînements entre les différents segments d'opinion. Les portions qui ne correspondent à aucune expression d'opinion sont quant à elles modélisées par une boucle de phonèmes en contexte, sans contraintes a priori sur les enchaînements de phonèmes. L'ensemble est compilé au sein d'un unique modèle, appelé *RECO2* dans la présentation des expériences de la section 5 . La figure 1 présente ces trois types de modèles sur un exemple de message.

L'ensemble des segments extraits sur toutes les étiquettes représente environ 18700 occurrences de mots et le nombre de mots différents par sous-corpus ne dépasse pas 780 pour une moyenne de 470. Le premier intérêt est donc d'avoir réduit fortement le champ lexical. Par ailleurs, les messages se caractérisent globalement par un haut degré de disfluence. Or à nouveau, les parties les plus disfluentes ne sont pas celles où le locuteur exprime son opinion mais plutôt celles où il relatent l'origine de son problème initial. On observe ainsi une réduction du degré de disfluence dans les segments extraits. Ceci est illustré dans le tableau 3.

Indicateur	# messages	# segments
pauses remplies	6.1	5.0
faux départs	1.9	1.7
reprises	4.2	3.9
répétitions	2.0	2.3
marqueurs discursifs	4.3	1.2

TAB. 3: Pourcentage des indicateurs de disfluence dans le corpus global et dans le corpus extrait

Hormis les répétitions, qui ne sont pas les phénomènes les plus problématiques pour la reconnaissance, l'ensemble des indicateurs ont un pourcentage plus faible dans les segments d'opinion extraits. La baisse la plus significative concerne les marqueurs discursifs qui sont assez difficiles à modéliser du fait de la variété de leurs contextes d'apparition et qui peuvent perturber le traitement ultérieur des messages du fait de leur ambiguïté. Les mots "bon" ou "bien" par exemple peuvent à la fois être porteurs de sens pour une opinion et neutres quand ils sont employés pour articuler le discours.

4. CLASSIFICATION AUTOMATIQUE

La détection d'opinions dans un message peut se ramener à une tâche de classification : attribuer à un message une étiquette relative à l'expression d'une opinion particulière.

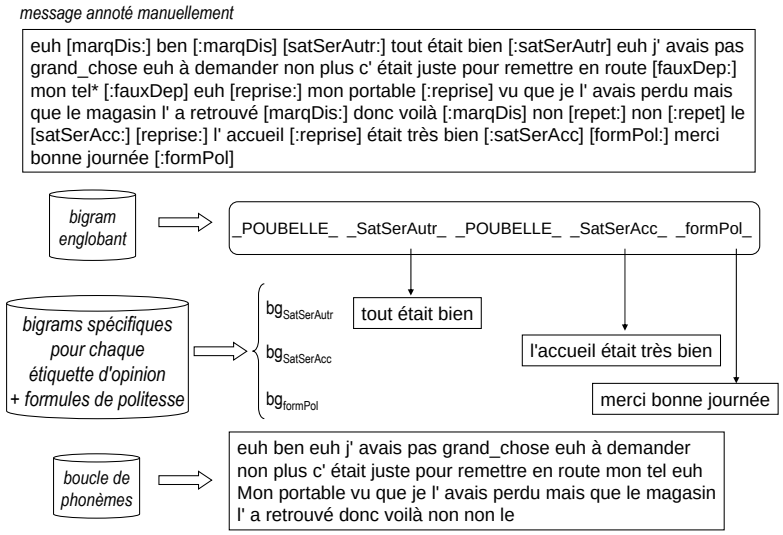

FIG. 1: Exemple de message annoté avec les 3 types de modèles de langage utilisés

Pour cette étape, nous utilisons une méthode de classification qui a prouvé son efficacité sur des tâches de classification de texte : les machines à support vectoriel ou *SVM*. L'implémentation des SVM utilisée dans cette étude est l'outil *SVMTorch* [2].

Comme mentionné précédemment, un message peut être caractérisé par plusieurs concepts (différents, identiques, contraires ...). Nous avons appris 8 classifieurs binaires, un par étiquette d'opinion, et l'étiquetage d'un message consiste à appliquer ces 8 classifieurs puis à en concaténer les décisions pour produire.

Les modèles sont appris sur le même corpus d'apprentissage que celui utilisé pour les modèles de langage présentés au paragraphe 3. *SVMTorch* est un classifieur qui prend en entrée un vecteur de données représentant le message à classer. Ce vecteur est de taille fixe pour un modèle considéré. Le choix des paramètres consiste donc à modéliser le message sous la forme d'un vecteur. De très nombreuses modélisations sont possible pour classer des données textuelles telles que la technique du *sac de mots*, la représentation par vecteurs de n-grammes, des n-grammes à trous, etc. Cette étude étant centrée sur l'impact de différents modèles de segmentation et de reconnaissance de parole sur les performances de classification, plus que sur l'étude de la modélisation pour les classifieurs, nous avons limité nos expériences aux deux représentation suivantes :

1. la modélisation la plus simple consiste à considérer comme vecteur d'entrée, le lexique complet de l'application. Un message est alors représenté par un vecteur de 2981 composantes. Les composantes non nulles de ce vecteur sont le nombre d'occurrences de chaque mot apparaissant dans le message. L'ordre des mots dans le message n'entre pas en jeu.

2. Une approche typique de l'analyse des opinions dans un texte consiste à créer un lexique contenant un ensemble de mots (appelés *seeds*) susceptibles d'exprimer une polarité positive ou négative [6]. Cette modélisation consiste ainsi à limiter le vecteur représentant un message aux seuls mots *seeds* du lexique.

Pour cette dernière modélisation, un ensemble de mots *polarisés* a été listé manuellement. Exemple : aberrant, compliments, discourtois, embêtement, ... Afin de généraliser la liste de mots polarisés obtenue, chaque mot est remplacé par son lemme. C'est cet ensemble de 565 lemmes polarisés que nous notons *seeds*. Le message est alors représenté par un vecteur de 565 composantes.

5. EXPÉRIENCES

Deux expériences sont faites en parallèle selon les deux types de paramètres choisis (mots ou *seeds*). Les résultats obtenus sont présentés par le calcul de la précision P, le rappel R et une combinaison précision/rappel : la F-mesure F.

Le corpus testé représente 33% des 1779 messages collectés et transcrits manuellement. Il est transcrit sous trois formes :
- *REF* : les messages sont transcrits manuellement ;
- *RECO1* : les messages sont transcrits automatiquement. Le modèle utilisé est de type bigram sur le lexique complet de l'application.
- *RECO2* : les messages sont transcrits automatiquement avec le modèle présenté en section 3

5.1. *Classification de messages sans segmentation préalable*

Les différents modèles appris sur le corpus d'apprentissage détectent les concepts recherchés sur le corpus de test. Les résultats sont indiqués dans le tableau 4.

Les résultats montrent bien la difficulté de la tâche. Ils sont d'ailleurs similaires à ceux obtenus dans ce type de caractérisation [1] avec une précision approchant les 60% sur du texte propre.

	mots			seeds		
(en%)	P	R	F	P	R	F
REF	59.2	37.3	45.8	57.9	43.6	49.8
RECO1	52.7	28.0	36.5	53.9	37.7	44.4
RECO2	51.5	34.9	41.6	52.3	40.0	45.4

TAB. 4: Résultat sur le message non-segmenté, référence manuelle (REF) et deux modèles de reconnaissance de parole (RECO1) et (RECO2)

Malgré un fort taux d'erreur mot, la détection des critères dans le message transcrit automatiquement ne fait perdre que 7 points de précision. Le rappel lui décroît de 10 points. L'utilisation des modèles *RECO2* au lieu des modèles *RECO1* permet de pallier cette perte de rappel avec moins de 3 points de différence avec le rappel obtenu sur le texte propre.

L'utilisation des seeds au lieu des mots permet d'améliorer la F-mesure sur le texte propre de 4 points. On remarque que c'est surtout le rappel qui tire avantage de ce type de paramètre. De même en ce qui concerne *RECO1* et *RECO2*, l'utilisation des seeds augmente le rappel de 10 points pour *RECO1* et de plus de 5 points pour *RECO2* et permet ainsi de passer d'une F-mesure de 36.52% à une F-mesure de 45.36% et de n'être plus qu'à 4 points de la F-mesure maximale obtenue jusqu'ici sur le texte propre.

5.2. Segmentation de messages

Les bons résultats obtenus avec *RECO2* et l'utilisation des seeds montrent que le traitement du message par segment permet de mieux cerner l'information pertinente. Comme pour *RECO2*, un sous-corpus ne contenant que les segments d'interventions porteurs de sens d'après l'annotation manuelle du corpus d'apprentissage, est extrait. Tous les modèles évoqués précédemment sont ré-appris sur ce sous-corpus. Le message d'entrée est lui aussi segmenté. Cette segmentation dépend du type de transcription du corpus.

– REF : segments obtenus manuellement, ce sont ceux porteurs de l'étiquette sémantique recherchée.
– RECO1 : segments obtenus automatiquement selon les pauses marquées par le locuteur durant son discours.
– RECO2 : segments obtenus automatiquement à chaque changement de modèles de transcription.

Chaque message est alors représenté par un ensemble de segment. Chaque segment est testé indépendamment par chaque modèle. L'ensemble des réponses obtenues pour un même message est ensuite rassemblé pour ne juger finalement que l'étiquette globale obtenue par le message sur l'ensemble de ses segments. Les résultats obtenus sont présentés dans le tableau 5.

	mots			seeds		
(en%)	P	R	F	P	R	F
REF	75.8	60.0	67.0	72.8	68.7	70.7
RECO1	49.0	33.9	40.0	48.1	43.9	45.9
RECO2	39.7	63.8	48.9	40.2	62.9	49.0

TAB. 5: Comparaison de 3 méthodes de segmentation de messages

Les résultats du corpus REF sont nettement améliorés avec une augmentation de la F-mesure de 20 points. Dans une moindre mesure, les résultats obtenus sur *RECO1* et *RECO2* sont eux aussi améliorés. On observe que c'est surtout le rappel qui est amélioré. En effet, l'apprentissage sur les segments porteurs de sens cible précisément les segments que l'on retrouve donc plus facilement. *RECO2* montre l'augmentation du rappel la plus remarquable allant de 20 à 30 points. Ceci s'explique par la construction même du message de *RECO2* qui recherche exactement les mêmes segments que ceux qui ont servis à l'apprentissage des modèles discriminants.

6. CONCLUSIONS ET PERSPECTIVES

La spécification du module de transcription à notre tâche ainsi que la recherche d'information pertinente pour la construction des modèles de classification et la représentation du message nous ont permis d'améliorer les premiers résultats.
En effet, les derniers résultats obtenus sur le texte propre sont acceptables dans ce type de tâche qui reste très difficile. Cette difficulté est amplifiée par la translation du problème de détection d'opinions sur des messages oraux énoncés par de *vrais* utilisateurs. Malgré le fort taux d'erreurs induit par cette parole spontanée, la détection d'opinions dans les messages issues du module de transcription obtient des résultats qui dépassent notre première baseline sur le texte propre.

Il s'agit maintenant de persévérer dans une recherche d'information plus pertinente en étendant par exemple les *seeds* à des patterns afin de capter le contexte et donc de mieux structurer l'information. L'intégration d'informations d'autres niveaux, par exemple prosodique, semble également indispensable, tant il est vrai que d'une part toute l'information portée par un message ne se résume pas à sa transcription lexicale, et que d'autre part la transcription automatique des messages est elle même peu fiable.

RÉFÉRENCES

[1] Yejin Choi, Claire Cardie, Ellen riloff, and Siddharth Patwardhann. Identifyind sources of opinions with conditional random fields and extraction patterns. In *Proceedings of HLT/EMNLP*, pages 355–362, Vancouver, 2005.

[2] Ronan Collobert, Samy Bengio, and Johnny Mariéthoz. Torch : a modular machine learning software library. In *Technical Report IDIAP-RR02-46, IDIAP*, 2002.

[3] Ana-Maria Popescu and Oren Etzioni. Extracting product features and opinions from reviews. In *Proceedings of HLT/EMNLP*, pages 339–346, Vancouver, 2005.

[4] Ellen Riloff and Janyce Wiebe. Learning extraction patterns for subjective expressions. In *Conference on Empirical Methods in Natural Language Processing*, 2003.

[5] Jayce Wiebe, Theresa Wilson, and Claire Cardie. Annotationg expressions of opinions and emotions in language. In *Language Resources and Evaluation*, volume 39, pages 165–210, 2005.

[6] Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of HLT/EMNLP*, pages 347–354, Vancouver, 2005.