Analyse faiblement supervisée de conversation en actes de dialogue

Kate Thompson[1] Nicholas Asher [1] Philippe Muller[2] Jeremy Auguste [3]
(1) IRIT, CNRS, (2) IRIT, Université de Toulouse, (3) LIS, Université Aix-Marseille
kate.thompson@irit.fr, nicholas.asher@irit.fr, philippe.muller@irit.fr,
jeremy.auguste@lis-lab.fr

RÉSUMÉ

Nous nous intéressons ici à l'analyse de conversation par *chat* dans un contexte orienté-tâche avec un conseiller technique s'adressant à un client, où l'objectif est d'étiqueter les énoncés en actes de dialogue, pour alimenter des analyses des conversations en aval. Nous proposons une méthode légèrement supervisée à partir d'heuristiques simples, de quelques annotations de développement, et une méthode d'ensemble sur ces règles qui sert à annoter automatiquement un corpus plus large de façon bruitée qui peut servir d'entrainement à un modèle supervisé. Nous comparons cette approche à une approche supervisée classique et montrons qu'elle atteint des résultats très proches, à un coût moindre et tout en étant plus facile à adapter à de nouvelles données.

ABSTRACT

Weakly supervised dialog act analysis

We are interested here in conversation analysis in the form of written *chats* in a task-oriented context between a customer and technical assistant. Our objective is to provide dialog act labels to each utterance, as a source of information for downstream tasks. We experimented with a weakly supervised approach based on heuristic rules, a few development annotations, and an ensemble method that is used to annotate more data in a noisy manner. This can be leveraged to train a more classical supervised approach. We compare this method to a classically supervised model and show that results are comparable, at a fraction of the annotating cost and arguably a better generalisation.

MOTS-CLÉS : Dialogue, chat, actes de dialogue, apprentissage faiblement supervisé.

KEYWORDS: dialog, chat, dialog act, weakly supervised learning.

1 Introduction

L'analyse de conversation est une perspective intéressante dans le cadre de la gestion de la relation client, notamment à cause de l'essor des plate-formes de conseil mettant en relation des clients avec des conseillers pouvant les aider à régler des problèmes techniques ou commerciaux. Une part importante de ces échanges a lieu par écrit dans le cadre d'échange par *chat*. Alors que l'analyse de corpus de conversation sous forme orale a une longue histoire, avec des corpus reconnus (Godfrey *et al.*, 1992; Carletta, 2007), les conversations sous forme écrite permis par les nouvelles formes de media sociaux (chat, forums, microblogging) ont fait l'objet de moins d'attention, même si cela commence à changer, par exemple les travaux sur les forums (Wang *et al.*, 2012), le chat, notamment en anglais (Asher *et al.*, 2016) et récemment en français (Damnati *et al.*, 2016).

L'analyse en actes de dialogue est un premier niveau d'analyse de la conversation qui permet de caractériser la fonction de communication d'un énoncé ou d'un tour de parole, en déterminant si un énoncé est une offre, une réponse, un retour sur un échange, ou si l'énoncé a une fonction sociale ou en rapport avec une tâche à réaliser. Ce niveau sert souvent de base à des analyses plus précises des échanges. Peu de travaux ont étudié la particularité des conversations par chat pour ce niveau d'analyse, à quelques exceptions près, comme (Perrotin *et al.*, 2018) pour des conversations client-conseillers. Dans tous les cas, les approches automatiques de l'annotation en actes de dialogue utilisent une supervision forte, qui nécessite une quantité de données non négligeables, le corpus Switchboard étant un exemple typique (Ji *et al.*, 2016; Kumar *et al.*, 2018). Ces modèles sont très dépendants du type d'interaction et des sujets de conversation, et *a fortiori* rien n'indique qu'ils se généralisent à de l'interaction écrite.

Nous proposons ici une méthode fondée sur une supervision indirecte, à partir de règles d'annotation superficielles, qui permet d'entraîner un modèle d'ensemble utilisé pour annoter automatiquement un corpus large. Ce dernier corpus sert alors à superviser l'entraînement d'un autre modèle, selon une méthodologie popularisée sous le nom de *data programming* par ses auteurs (Ratner *et al.*, 2016; Bach *et al.*, 2017). Nous évaluons ici l'apport de cette méthode en la comparant à une méthode plus classiquement supervisée, en montrant que quelques dizaines de règles assez simples, essentiellement lexicales, permette d'atteindre une exactitude de 80%, qui n'est atteinte de façon supervisée qu'avec plusieurs milliers de conversations, le meilleur modèle classiquement supervisé atteignant 84-86% sur des données comparables (Perrotin *et al.*, 2018). Dans cette étude et la notre la typologie des actes de dialogue employée correspond à une granularité intermédiaire avec 10 étiquettes différentes, décrites ci-dessous.

2 Un modèle de "programmation par les données"

Pour développer un modèle indirectement ou faiblement supervisé, nous suivons la méthode de *data programming* définie par (Ratner *et al.*, 2016; Bach *et al.*, 2017), qui consiste en les étapes suivantes :

- l'écriture manuelle d'heuristiques de catégorisation, partielles et potentiellement contradictoires, mais en cherchant une bonne couverture des données ;
- l'apprentissage d'un modèle génératif qui cherche à approximer la probabilité conjointe des catégories prédites et de l'exactitude et la couverture des règles écrites ; on peut voir cette étape comme une forme d'ensemble de classifieurs pondérés en fonction de leur accord mutuel ;
- à partir de ce modèle, l'annotation automatique "bruitée" des données (c'est-à-dire avec une distribution de probabilités sur toutes les catégories au lieu d'une étiquette unique) ;
- enfin, un modèle supervisé standard peut-être appris sur ces données annotées, en cherchant à coller à la distribution des étiquettes (avec une fonction de perte ajustée à ce cadre).

Ce modèle est conçu au départ comme un outil d'extraction d'informations, notamment de relations entre entités nommées. Il permet sur ces tâches d'obtenir des scores proches d'approches supervisées, sans nécessiter d'annotation de données d'entraînement, l'annotation étant réservée au développement des règles et à l'évaluation. Les auteurs ont aussi montré que des non-experts pouvaient développer des règles dans un temps équivalent à de l'annotation manuelle classique, avec une fiabilité égale, sinon meilleure (Ratner *et al.*, 2016).

Nous adaptons ici cette approche au cas de la classification de textes, où l'énoncé d'un acte de dialogue est impliqué dans une relation unaire (le type d'acte de dialogue).

Acte	N	%	sup. %	Description
STA	479	45.20	39.16	affirmation/apport d'information
INQ	149	14.10	19.21	demande d'information
ACK	113	10.70	6.62	acquiescement
OPE	100	9.50	4.07	ouverture du dialogue
PPR	63	5.90	15.18	proposition de résolution du problème
CLO	49	4.50	5.48	clôture du dialogue
PRO	37	3.50	5.73	énoncé du problème
CLQ	34	3.20	1.74	question de clarification
TMP	27	2.50	2.43	mise en pause du dialogue
OTH	8	0.76	0.38	autre

TABLE 1 – Distribution des actes de dialogues dans le jeu de développement utilisé pour définir les heuristiques, avec pourcentages comparés à la distribution dans les données d'entraînement de l'approche supervisée de (Perrotin *et al.*, 2018).

3 Données utilisées

Pour cette expérience nous utilisons les données récoltées dans le cadre du projet Datcha [1] avec l'opérateur téléphonique Orange, portant sur l'étude de relation clients par chat, et constitué de conversations écrites entre un télé-conseiller et un client cherchant à résoudre un problème, technique ou commercial.

Nous avons sélectionné un sous-ensemble de conversations, dont la majeure partie sert d'ensemble d'entrainement pour le modèle génératif, une petite partie sert d'ensemble de développement pour la mise au point des heuristiques de catégorisation, et une autre petite partie sert de données de tests pour évaluer l'approche et pouvoir la comparer aux alternatives. Seule les parties test et développement sont manuellement annotées.

Les dialogues sont segmentés automatiquement, à partir des journaux d'interaction client-conseiller, qui liste les échanges verbaux et des métadonnées sur le contexte (service contacté, enquête de satisfaction par exemple) qui ne sont pas utilisées ici, à l'exception de certaines interventions automatique de la plate-forme de support, qui sont reliées au contexte de la conversation. Pour cela, chaque retour à la ligne d'un participant au chat est considéré comme délimitant la fin d'un acte, et nous appliquons le segmenteur en phrases de l'outil CoreNLP (Manning *et al.*, 2014) à chaque ligne, en plus de quelques heuristiques pertinentes pour des dialogues : segmentation sur les " ?", les ouvertures ou marques fréquentes d'ouverture d'actes de dialogue : *ok, merci, d'accord, bonjour, sinon*. Chaque acte ainsi délimité est censé ne correspondre qu'à un seul des types d'actes prévus.

La partie d'entrainement est constitué de 3000 conversations, le développement de 13 conversations segmentées et annotées en actes de dialogue, et le test de 2 conversations, ce qui correspond respectivement à 155k, 1059 et 181 segments.

Nous avons suivi le schéma d'annotation choisi par (Perrotin *et al.*, 2018) et appliqué sur des données similaires. La table 1 montre la distribution des types d'actes de dialogue dans les données de développement, et la comparaison avec la distribution reportée sur le corpus de (Perrotin *et al.*, 2018).

1. http://datcha.lif.univ-mrs.fr/

Acte	Locuteur	énoncé
OPE	INFO	Vous entrez en conversation avec TC1.
INQ	TC	Que puis-je faire pour vous ?
CLQ	TC	sans exception ?
ACK	TC	Rassurez vous ,nous allons voir cela ensemble
PRO	CL	j'ai changé de forfait hier, j'ai pris le ...
PPR	TC	Je vous propose de recevoir par voie postale...
TMP	TC	je vous prie de rester en ligne
STA	CL	pas de tonalité
CLO	CL	bonne journée à vous
OTHER	CL	répondez moi svp

TABLE 2 – Exemple d'énoncés d'actes de dialogue avec leur type pris dans différentes conversations. TC = téléconseiller, CL = client, INFO=intervention automatique de la plate-forme de support.

On peut noter des différences importantes sur les acquiescements et les ouvertures, probablement car le travail cité ne segmente pas à l'intérieur des tours de parole, quitte à donner un acte de dialogue "principal" pour le tour, ce qui peut faire négliger les catégories d'actes plus sociales que liées à la tâche (et représente une perte d'information que nous avons voulu éviter). De même il y a une grosse différence sur les actes de propositions de résolution du problème, qu'il est difficile d'expliquer autrement que par un biais d'échantillonnage. La table 2 montre un exemple de dialogue (extrait) avec les actes associés aux énoncés, et la table 2 montre des exemples d'énoncés pour chaque acte.

4 Expérimentations

Pour évaluer l'intérêt et les performances de l'approche légèrement supervisée, nous détaillons ici l'expérimentation faite en comparant un modèle "génératif" à partir des règles produites et des dialogues non annotés, ainsi qu'un modèle discriminatif entraîné sur ces données bruitées résultantes en section 4.1, et un modèle classiquement supervisé, section 4.2.

4.1 Modèle génératif à base de règles

Le modèle génératif repose sur un ensemble d'heuristiques (51) prédisant la classe des énoncés sur la base d'informations superficielles :

- patrons lexicaux spécifiques ;
- type du locuteur (conseiller ou client) ;
- position de l'énoncé dans le dialogue (proche du début/de la fin) ;
- contenu du contexte dialogique (type du locuteur des tours précédents et/ou suivants) ;

Ces règles ont été développées à partir d'un petit ensemble de dialogues annotés manuellement, sur lesquels leur couverture et précision sont estimées.

Ces règles sont par exemple de la forme :

— **si** le locuteur est le téléconseiller, et le tour commence (à n caractères près) par j (...) (vais|suis en train|vien|confirm| fai| prend|consult) et le tour ne contient pas " ?" **alors** le type d'acte est PPR (proposition de résolution de problème)
— **si** le locuteur est le client et le tour n'est pas social/une ouverture* et le tour précédent était par le télé-conseiller et contenait une forme de proposition d'aide* **alors** le type de l'acte est un PRO (énoncé du problème). Ici les parties de règle signalées avec un * sont exprimées sous forme d'expressions régulières sur la forme du tour, en listant les alternatives possibles (comme pour la règle précédente).

Une règle par défaut est déclenchée si aucune autre ne couvre le cas considérée, et attribue l'étiquette STA (affirmation), qui est la classe majoritaire.

Ces règles sont partielles, dans le sens où elles peuvent s'abstenir d'une décision sur une instance, et peuvent couvrir en partie les mêmes instances, de façon contradictoire ou non. Sur la base de ces règles et de données (non annotées) le modèle génératif peut donner la distribution jointe des classes et de la précision des règles par maximum de vraisemblance, cf (Ratner *et al.*, 2016) p. 4. En utilisant ces distributions sur les données de la partie d'entraînement (3000 dialogues), on peut alors entrainer un modèle supervisé de façon "bruitée". Nous utilisons l'implémentation Snorkel pour l'entrainement du modèle génératif [2], et un réseau de neurones récurrent pour apprendre la catégorisation des énoncés, en s'assurant de définir une fonction de perte qui prend en compte une distribution de probabilités comme référence au lieu d'un label unique (ici une mesure d'entropie croisée). Nous utilisons un bi-LSTM simple à une couche, qui utilise les 2 états finaux de la séquence comme entrée d'une couche finale avant un softmax sur les scores des classes. L'état du LSTM a une dimension de 256 les mots en entrée étant plongés dans un espace de 100 dimensions, initialisés avec des embeddings fastText (Bojanowski *et al.*, 2017) calculés sur le corpus d'entrainement, car ils donnent une certaine robustesse à ces représentations (indispensable vue la nature du langage utilisé dans les données).

Les réglages du LSTM ont été faits en observant les résultats sur le corpus de développement, sans faire des réglages très fins dans la mesure où les règles de départ sont déjà très biaisées par rapport au jeu de développement. Nous n'avons pas essayé non plus de reproduire exactement le modèle supervisé utilisé pour la comparaison dans la section suivante, calqué sur (Perrotin *et al.*, 2018), dans la mesure où nous ne pouvons savoir si les valeurs optimales pour une configuration (supervisée/générative) se généralise à l'autre. Tout juste sommes nous restés dans des espaces de paramètres et des capacités relativement comparables.

4.2 Modèle supervisé de comparaison

Le modèle supervisé utilisé dans nos expérimentations est le réseau de neurones décrit dans (Perrotin *et al.*, 2018), sans la couche CRF pour garder un modèle similaire à la partie précédente. C'est un réseau de neurones récurrent hiérarchique à deux niveaux. Le premier niveau permet de s'intéresser aux tours de paroles en prenant en entrée la séquence de mots de chaque tour. Le second niveau permet de prendre en compte l'ensemble de la conversation à partir des états cachés en sortie du premier niveau représentant les tours de paroles. Les deux niveaux sont des réseaux récurrents bidirectionnels de type LSTM. La couche de décision utilise les états cachés du LSTM du deuxième niveau afin d'obtenir une prédiction de l'acte de dialogue de chaque tour de parole. Les états cachés du premier

2. github.com/HazyResearch/snorkel

niveau sont de taille 64 et ceux du second niveau sont de taille 128. En entrée du premier niveau, les mots sont représentés par des embeddings de dimension 100 qui sont entraînés en même temps que le reste du réseau. L'information sur le scripteur de chaque tour est également donné en entrée du second niveau sous forme d'embeddings de dimension 5 entraînés par le réseau. L'entropie croisée est utilisée pour la fonction de coût et Adadelta est utilisé pour la rétropropagation du gradient.

Pour l'entraînement, nous utilisons les mêmes données que dans (Perrotin *et al.*, 2018). Dans ce corpus annoté manuellement, un seul acte de dialogue est attribué à chaque tour de parole et aucune segmentation supplémentaire n'est réalisée. Le corpus d'entraînement est composé de 2390 dialogues. Afin de pouvoir comparer ce modèle avec le modèle de la section précédente, nous utilisons ce modèle sur les 2 conversations de la partie de test.

4.3 Résultats et analyse

Le modèle génératif entraîné permet d'estimer l'apport des règles pondérées par le modèle et selon leurs paramètres estimés : nous reportons en table 3 les couvertures min, max et en moyenne des règles, ainsi que leur exactitude. Pour les modèles supervisés, le modèle entrainé sur les données

Mesure	Min	Max	Moyenne
Exactitude	0.4918	0.5035	0.4971
Couverture	0.6727	0.6846	0.6786

TABLE 3 – Performances des règles sur les données de test : exactitude du label prédit, couverture des instances par chaque règle.

annotées automatiquement par le modèle génératif atteint une exactitude de 80.1% sur le test, alors que le modèle supervisé classique n'atteint que 67.4% sur ces données de test. Il faut prendre ce score avec prudence car le modèle est entrainé sur une segmentation en actes différente, avec une distribution des étiquettes différentes comme on l'a vu. Evalué sur un jeu de test différent mais avec la même segmentation, (Perrotin *et al.*, 2018) reporte un score de 84% avec le même réseau de neurones, et 86% quand le réseau est adjoint à un CRF séquentiel. On peut noter que d'après cet article, il faut déjà 500 dialogues [3] annotés pour atteindre les 80% de la méthode non supervisée (même si là encore les conditions ne sont pas exactement les mêmes, l'ordre de grandeur est plausible).

En complément d'analyse, nous présentons les résultats par type d'acte de dialogue selon la méthode discriminante entrainée sur les données "automatiques" ou avec la supervision "classique" en table 4. Au vu de la petite taille du corpus de test, il est difficile de tirer des conclusions trop rapides sur certains labels dont le support est très bas et implique une variance importante. On peut simplement noter que les quatre classes les plus présentes dans le test sont les mêmes que dans le développement (avec un ordre légèrement différent), et que la généralisation est meilleure pour le modèle faiblement supervisé, alors que le modèle non supervisé est entrainé sur une distribution différente. Ceci peut expliquer l'écart sur ces données de test, mais il reste que le niveau de performance du modèle faiblement supervisé est comparable au supervisé quand chacun est "entrainé" sur une distribution comparable à son propre jeu de test.

3. Soit entre 20k et 30k instances d'actes.

Type d'acte	P.S.	R.S.	F.S.	P.D.	R.D.	F.D.	Support
STA	0.58	0.80	0.67	0.64	0.89	0.74	54
OPE	1.00	0.48	0.65	0.96	1.00	0.98	23
PRO	0.31	0.57	0.40	0.00	0.00	0.00	7
INQ	0.85	0.89	0.87	0.85	0.89	0.87	19
CLQ	0.62	0.62	0.62	1.00	0.50	0.67	8
ACK	1.00	0.50	0.67	0.89	0.71	0.79	34
TMP	1.00	1.00	1.00	1.00	1.00	1.00	13
PPR	0.33	0.55	0.41	1.00	0.73	0.84	11
CLO	0.86	0.67	0.75	0.80	0.89	0.84	9
OTHER	0.00	0.00	0.00	0.00	0.00	0.00	3
Moyenne	0.75	0.67	0.68	0.79	0.80	0.78	181

TABLE 4 – Résultats par type d'actes de dialogue, avec P(récision), R(appel), F(score) pour l'approche discriminante bruitée (D) et l'approche supervisée classique (S). Le support indique le nombre d'instances par classe.

5 Perspectives et conclusion

Nous avons présenté ici un modèle d'étiquettage en actes de dialogue légèrement supervisé, à partir d'heuristiques simples, de quelques annotations de développement, et une méthode d'ensemble sur ces règles qui sert à annoter automatiquement un corpus plus large de façon bruitée. Nous avons montré que ce modèle est compétitif avec une approche supervisée. Un autre avantage plausible de cette approche est de faciliter le transfert vers des données différentes, d'une part parce que la conception des règles est plus robuste, d'autre part parce qu'ajouter des spécificités de nouvelles données se fait aisément en adaptant les règles, quand un modèle supervisé nécessite soit de nouvelles annotations, soit une approche d'apprentissage par transfert aux résultats incertains. Cette hypothèse pourrait être testée avec des données nouvelles, ou bien déjà en séparant les données en sous-domaine selon les catégories de problème technique définies par l'opérateur sur sa plate-forme.

Remerciements

Ce travail a été financé par l'Agence Nationale pour la Recherche, dans le cadre du projet DATCHA (ANR-15-CE23-0003).

Références

ASHER N., HUNTER J., MOREY M., BENAMARA F. & AFANTENOS S. D. (2016). Discourse structure and dialogue acts in multiparty dialogue : the stac corpus. In *LREC*.

BACH S. H., HE B. D., RATNER A. & RÉ C. (2017). Learning the structure of generative models without labeled data. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, p. 273–282.

BOJANOWSKI P., GRAVE E., JOULIN A. & MIKOLOV T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, **5**, 135–146.

CARLETTA J. (2007). Unleashing the killer corpus : experiences in creating the multi-everything AMI meeting corpus. *Language Resources and Evaluation*, **41**(2), 181–190.

DAMNATI G., GUERRAZ A. & CHARLET D. (2016). Web chat conversations from contact centers : a descriptive study. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC 2016, Portorož, Slovenia, May 23-28, 2016.*

GODFREY J. J., HOLLIMAN E. C. & MCDANIEL J. (1992). Switchboard : telephone speech corpus for research and development. In *[Proceedings] ICASSP-92 : 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, p. 517–520 vol.1.

JI Y., HAFFARI G. & EISENSTEIN J. (2016). A latent variable recurrent neural network for discourse-driven language models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies*, p. 332–342, San Diego, California : Association for Computational Linguistics.

KUMAR H., AGARWAL A., DASGUPTA R. & JOSHI S. (2018). Dialogue act sequence labeling using hierarchical encoder with CRF. In *AAAI*, p. 3440–3447 : AAAI Press.

MANNING C. D., SURDEANU M., BAUER J., FINKEL J., BETHARD S. J. & MCCLOSKY D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, p. 55–60.

PERROTIN R., NASR A. & AUGUSTE J. (2018). Dialog Acts Annotations for Online Chats. In *25e conférence sur le Traitement Automatique des Langues Naturelles (TALN)*, Rennes, France.

RATNER A. J., SA C. D., WU S., SELSAM D. & RÉ C. (2016). Data programming : Creating large training sets, quickly. In *Advances in Neural Information Processing Systems 29 : Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, p. 3567–3575.

WANG L., KIM S. N. & BALDWIN T. (2012). The utility of discourse structure in identifying resolved threads in technical user forums. In *COLING*, p. 2739–2756 : Indian Institute of Technology Bombay.