

# Aide à l'automatisation de conception de systèmes de dialogue

Jean-Leon Bouraoui\*, Vincent Lemaire\*

\*2 Avenue Pierre Marzin, 22300 Lannion  
{jeanleon.bouraouilvincent.lemaire}@orange.com,  
<http://vincentlemaire-labs.fr/>

**Résumé.** L'article décrit un processus industriel en cours de recherche / développement. L'objectif est d'obtenir une modélisation non supervisée de la structure de dialogues finalisés appartenant à un domaine donné (par exemple réservation de trains). Ce type de modélisation présente de nombreux intérêts, le principal étant de servir de base à la conception de l'architecture d'un agent dialoguant. La modélisation obtenue est représentée par un graphe présentant les principales étapes des dialogues et les transitions entre elles. La technique adoptée consiste à appliquer du CoClustering sur le corpus cible de dialogues, afin d'obtenir les principaux thèmes qui y figurent. On calcule ensuite les transitions entre thèmes dans chaque dialogue, pour obtenir le graphe décrivant les principaux thèmes du corpus et leur séquentialité. Nous décrivons en détail le processus mis en place, et situons cette approche par rapport aux travaux connexes. Enfin, nous présentons les verrous scientifiques restants.

## 1 Introduction

En intelligence artificielle, les agents dialoguants connaissent un gain de popularité auprès du grand public ; et ce d'autant plus qu'ils bénéficient des avancées dans la compréhension des contenus et du contexte. Cela est le fait notamment d'applications mobiles telles que Siri (Apple), Google Now (Google), ou Cortana (Microsoft) ou Alexa (Amazon) . Pour quantifier cet intérêt grandissant pour la technologie des interfaces dialoguantes et des agents dialoguants en particulier, nous pouvons citer la récente étude du cabinet d'analyse Gartner, qui place les systèmes conversationnels parmi les 10 technologies stratégiques pour 2017<sup>1</sup>.

Une des tendances actuelles est de proposer des dispositifs logiciels de conception d'agents dialoguants, personnalisables selon les besoins et le domaine d'application (par exemple, réservation de voyages, commande de produits ou de services, etc.). L'un des enjeux de ces dispositifs est de pouvoir être mis en place rapidement, sachant qu'il n'existe actuellement pas de système générique, et qu'une adaptation de l'agent à un domaine d'application donné prend du temps.

Dans ce contexte nous présentons, comme base à discussion au cours de l'atelier TextMine d'EGC 2017, une méthodologie ayant pour but de mettre en place une solution d'assistance

---

1. <http://www.gartner.com/newsroom/id/3482617>

semi-automatique à la création ou l'adaptation d'un agent dialoguant pour un domaine applicatif donné.

## 2 Description de la problématique

Dans cet article, nous appellerons dialogue un échange d'informations entre deux interlocuteurs (sachant qu'un dialogue peut faire intervenir plus de deux interlocuteurs). Un interlocuteur peut être un humain ou une machine (au sens large : un système artificiel, logiciel ou matériel). Nous nous intéressons aux dialogues finalisés, qui cherchent à atteindre un but : les interlocuteurs vont collaborer pour l'atteinte de ce but.

On appelle dans cet article "corpus textuel" un ensemble de  $n$  dialogues relatifs à un domaine particulier, (par exemple transcriptions de dialogues de réservations de trains, ou tchats d'interactions entre un téléconseiller et un client). Chaque dialogue est composé de  $t$  tours de parole, un tour de parole correspondant à ce que dit l'un des interlocuteurs sans interruption (la plupart du temps, une ou plusieurs phrases).

Dans un premier temps, on cherche à associer chaque tour de parole à une "classe" donnée notée  $L_c$ . Une classe correspond à l'intention que l'interlocuteur exprime dans son tour de parole ; prenons comme exemple un dialogue entre un client et le service client d'un opérateur téléphonique : le tour de parole où l'agent demande au client de s'identifier appartient à une classe spécifique (nommée par exemple *DmdIdent*) ; celui où le client procède à son identification est relatif à une autre classe (nommée par exemple *ReplIdentClient*).

Ensuite, on détermine les thèmes  $T_t$ , qui regroupent un ensemble de classes relatives à un sujet commun. Reprenons notre exemple du service client : les thèmes pourront être l'identification du client (*IdentClient* ; classes correspondantes : la demande par le téléconseiller, et la réponse du client), la discussion du problème (*ExpProb* ; classes correspondantes : la présentation par le client, et la demande de précision par le téléconseiller), etc.

L'association des classes aux tours de parole des différents dialogues du corpus pourra donc être représentée comme indiqué dans la figure 1, et le regroupement des classes en thèmes comme représenté dans le tableau 1.

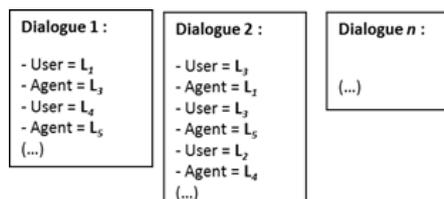


FIG. 1 – Association de classes aux tours de paroles

Notre but est de déterminer automatiquement : (i) les classes ; (ii) les thèmes ; (iii) les transitions entre thèmes et ceci au sein de chaque dialogue, de manière à obtenir une représentation du déroulement typique des dialogues du corpus. La représentation souhaitée est celle d'un graphe orienté montrant les principales transitions entre thèmes, comme celui représenté (et simplifié) sur la figure 2. Notre postulat est que, selon la position dans le dialogue, un tour

Thèmes	$T_1$	$T_2$
Classes	$L_1, L_4, L_5$	$L_3, L_6, L_7$

TAB. 1 – Regroupement des classes en fonction des thèmes auxquelles elles correspondent

de parole donné présente plus de probabilité d’appartenir à une classe donnée (i.e. un cluster), qu’un autre ; cette information est donc prise en compte lors du processus, et l’un des objectifs de notre travail est de vérifier la validité de ce postulat.

Le graphe ainsi obtenu présente de multiples intérêts. Le principal est l’initialisation de la conception du système de dialogue : il pourra servir de base à la modélisation de l’architecture d’un agent dialoguant spécialisé sur le domaine cible, et ainsi en faciliter l’exécution. A l’heure actuelle, cette tâche est la plupart du temps effectuée manuellement : soit a priori, à partir de la représentation que le concepteur se fait des dialogues possibles portant sur une tâche et un domaine donné ; soit a posteriori, à partir de la consultation de corpus existants ; dans les deux cas, le processus est coûteux en temps.

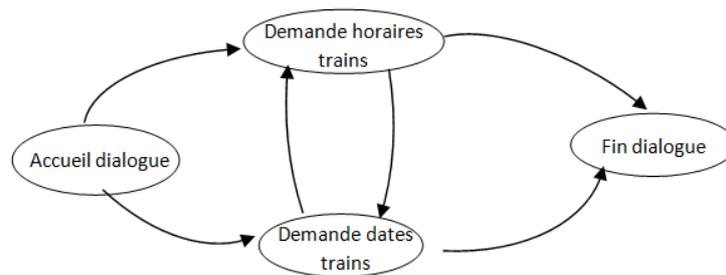


FIG. 2 – Représentation graphique des principales transitions entre thèmes

De plus, le graphe, ainsi que les étapes parcourues pour l’obtenir, permettra au concepteur, sans connaissance préalable du domaine d’application, d’avoir une première compréhension du contenu thématique des dialogues, de leur structuration, et plus généralement la connaissance des informations les plus pertinentes pour la réalisation de l’agent dialoguant.

### 3 Travaux connexes

Dans la littérature, on peut regrouper sous trois catégories les différentes approches employées pour répondre à notre problématique, selon qu’elles procèdent par l’identification des thèmes et de leur séquentialité, qu’elles utilisent le Deep Learning, ou qu’elles adoptent une approche *ad hoc*.

- Identification des thèmes puis des séquences de thèmes : les travaux appartenant à cette catégorie se différencient principalement selon la méthode utilisée pour identifier les

thèmes des dialogues. Ainsi, Bangalore et al. (2008) et Chotimongkol (2008) utilisent des clusters à cette fin, tandis que Paul (2012) et Zhai et Williams (2015) utilisent le modèle de la Latent Dirichlet Allocation (LDA), qui permet d'identifier les "thèmes" dans un document ou ensemble de documents textuels. Dans tous les cas de figure, ces auteurs utilisent les Hidden Markov Models (HMM) pour modéliser les transitions entre thèmes.

- Deep Learning : A notre connaissance, le premier article ayant proposé l'utilisation, pour les agents dialoguants, de techniques appartenant au Deep Learning, est Vinyals et Quoc (2015). Les auteurs utilisent le modèle *seq2seq* pour modéliser l'enchaînement des tours de parole entre interlocuteurs, dans un réseau neuronal récurrent. Deux domaines d'application sont décrits, dont l'un relève du dialogue finalisé (tchats de dépannage informatique).
- Autres approches : Nous regroupons dans cette catégorie les travaux qui utilisent des méthodes *ad hoc* à la place ou en complément d'une modélisation des données. Ainsi, D'Haro et al. (2009) appliquent des heuristiques logicielles pour constituer un modèle de dialogue à partir d'une base de données applicatives. Des heuristiques sont également utilisées dans Chalamalla et al. (2008) et Negi et al. (2009) pour passer d'une représentation en clusters à une modélisation du dialogue. Nous citons aussi Laroche (2015) qui intègre des techniques de Recherche d'Information dans l'architecture d'un agent dialoguant.

## 4 Description de notre approche

### 4.1 Détermination des classes et des thèmes

On utilise ici une technique de CoClustering qui permet d'obtenir une "copartition" de la matrice mots x tour de paroles. Étant données, deux (ou plus) variables catégorielles ou numériques, on réalise un partitionnement simultané des variables : les valeurs de variables catégorielles sont groupées en clusters et les variables numériques sont partitionnées en intervalles – ce qui revient à un problème de coclustering. Le produit des partitions uni-variées forme une partition multivariée de l'espace de représentation, i.e., une grille ou matrice de cellules et il représente aussi un estimateur de densité jointe des variables. Afin de choisir la "meilleure" grille (connaissant les données) de l'espace de modèles, une approche Bayésienne dite Maximum A Posteriori (MAP) est utilisée. La méthode employée est basée sur l'approche MODL décrite dans (Boullé et al., 2014). Comme pour toute méthode d'apprentissage automatique, notre approche nécessite une quantité de données : un volume minimum est requis pour que le clustering soit pertinent.

Un outil de Visualisation des CoClusters trouvés est ensuite utilisé (Guerraz et al., 2015), ce qui permet une analyse fine et un "profilage" des clusters obtenus. Nous ne décrivons pas ici tous les détails mais l'utilisateur peut choisir la granularité de la grille nécessaire à son analyse tout en contrôlant soit le nombre de parties soit le taux d'information mais aussi réaliser un taggage (par exemple à l'aide de mots clefs) des clusters obtenus.

## 4.2 Détermination des transitions entre thèmes

Dans notre approche de la problématique, un thème correspond à un cluster de tours de parole. Un thème défini ainsi peut être relié à un ou plusieurs autres thèmes, en fonction de la fréquence observée de leurs successions dans les dialogues du corpus.

La représentation obtenue est un graphe orienté, dont les nœuds sont les clusters, et les arcs sont les successions entre clusters. La génération du graphe se fait en plusieurs phases successives, schématisées dans la figure 3.

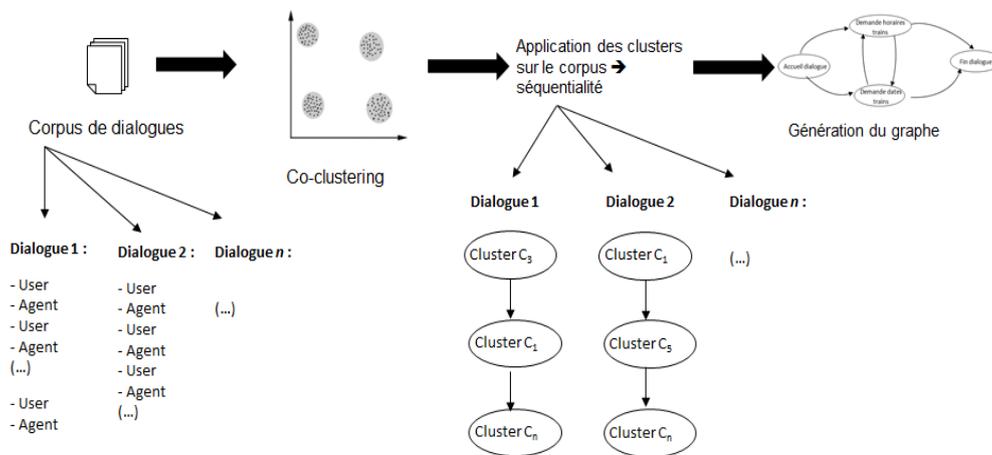


FIG. 3 – Chaîne de traitement

La première consiste à obtenir un corpus standardisé, à l'aide d'un outil de prétraitement de texte (interne à Orange Labs et non décrit ici) paramétré avec la liste de stopwords utilisée pour le stemming français par la librairie NLTK. Cette liste de stopwords permet de supprimer des mots a priori inutiles pour l'apprentissage (par exemple articles, prépositions, etc.).

La représentation ainsi obtenue est ensuite utilisée par l'outil de CoClustering mentionné ci-dessus, qui génère les clusters auxquels appartient chaque tour de parole du corpus. Les énoncés sont associés à des classes (ici : des clusters de tours de paroles) en fonction de la maximisation du contraste entre la distribution des co-parties (ici : des CoClusters de mots) et la distribution espérée sous l'hypothèse d'indépendance (connaissant les marginales). Les partitions induites par un CoClustering sur les deux entités sont des clusterings. La notion de similarité associée est liée à la façon dont les individus d'un cluster d'une entité se distribuent sur les clusters de l'autre entité.

On obtient ainsi un ensemble de clusters de tours de paroles, chaque cluster correspondant à un thème donné, et dénommé par un identifiant unique.

Les identifiants de cluster sont ensuite projetés sur le corpus initial, afin de retrouver la séquentialité des clusters. Autrement dit, à chaque tour de parole du corpus initial est désormais associé l'identifiant du cluster auquel le tour de parole appartient.

A partir de là, chaque dialogue peut ainsi être parcouru comme une séquence de cluster/thèmes. On peut ainsi calculer, sur l'ensemble des dialogues, les transitions entre chaque

cluster; le début et la fin de chaque dialogue sont pris en compte de manière à éviter des transitions erronées.

Les fréquences de transition entre clusters ainsi calculées sur l'ensemble des dialogues sont stockées dans une matrice. Celle-ci est utilisée pour générer le graphe correspondant. Il est possible de choisir un seuil minimum de fréquence de transitions, à partir duquel les transitions sont affichées : cela permet de générer des graphes plus ou moins complexes à lire. D'autre part, la possibilité de n'afficher que des transitions supérieures à un seuil permet de limiter la prise en compte de transitions non pertinentes, provoquées par des erreurs dans le clustering.

### 4.3 Illustration sur un cas concret

Nous avons utilisé l'approche décrite dans la section précédente sur différents corpus de dialogues, appartenant à des domaines variés. A titre d'illustration, évoquons ici un corpus de 7 407 dialogues de "tchats" d'assistance en ligne de clients d'Orange avec des téléconseillers; chaque dialogue comporte une quinzaine de tours de paroles en moyenne.

Nous avons appliqué à ce corpus le pré-traitement puis le CoClustering évoqués dans section précédente. On obtient 44 clusters différents, regroupant les tours de parole du corpus. Les identifiants des clusters sont ensuite associés, dans le corpus même, aux tours de paroles correspondants; on récupère ainsi la séquentialité des clusters, qui va ensuite permettre la génération des graphes.

En appliquant le seuillage défini plus haut (avec différentes valeurs de seuil par graphe), on obtient des visualisations graphiques globalement pertinentes : l'analyse manuelle du contenu des clusters obtenus à partir d'un seuil élevé montre que la plupart des clusters sélectionnés correspondent bien à des phases bien spécifiques de ce type de dialogue : phases de salutations, d'identification du client, de description du problème rencontré, de l'identification de la solution à apporter, et phases de remerciement et de fin de dialogue.

L'ordonnement de ces clusters/thèmes en termes d'architecture des dialogues montre également un déroulement séquentiel typique de ces différentes phases, nonobstant certaines erreurs de clustering. Celles-ci consistent en deux grandes catégories : soit l'hétérogénéité du cluster (un cluster correspond à plusieurs thèmes au lieu d'un seul), soit au contraire la redondance de clusters (plusieurs clusters différents correspondent au même thème).

A titre d'exemple, la figure 4 présente un graphe obtenu à partir de l'application de la chaîne de traitement sur le corpus mentionné ci-dessus. A fins de lisibilité, ce graphe n'affiche que 32 clusters : ceux dont le nombre de transitions est supérieur à 300; les nombres apparaissant à côté de chaque arc correspond aux fréquences de transition; chaque cluster est désigné par l'identifiant attribué automatiquement par Khiops.

La figure 5 présente un autre graphe, obtenu à partir des mêmes données, cette fois après analyse du contenu des clusters du graphe de la figure 4. Pour le générer, nous avons utilisé un outil (interne à Orange Labs) de visualisation et de manipulation des données. A partir de l'analyse, cet outil nous a permis de renommer les clusters, fusionner certains d'eux, et en rendre la présentation encore plus lisible, comme pourrait le faire le concepteur d'un agent dialoguant pour obtenir une première architecture sans connaissance préalable du domaine.

Pour l'instant nous évaluons les résultats obtenus selon deux critères. Le premier est l'homogénéité des clusters obtenus. Le second est la qualité et la régularité des transitions entre

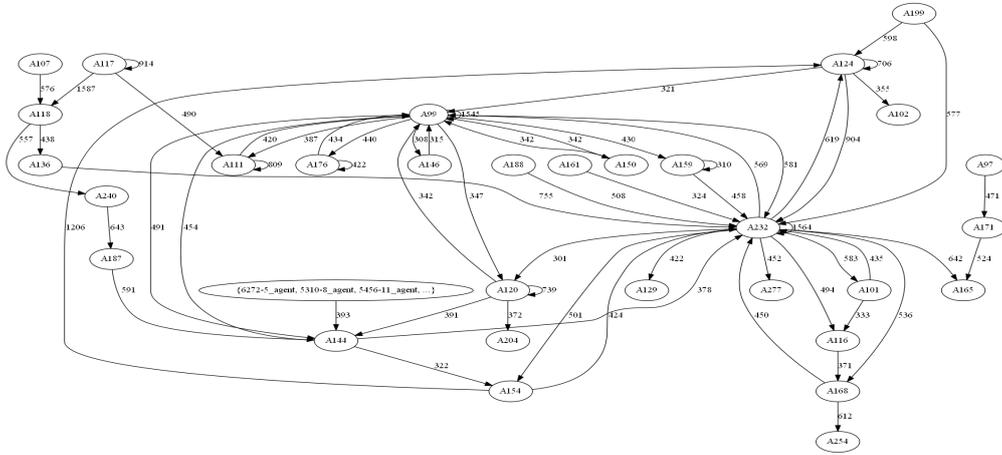


FIG. 4 – Graphe initial (seuil de fréquence minimum : 300)

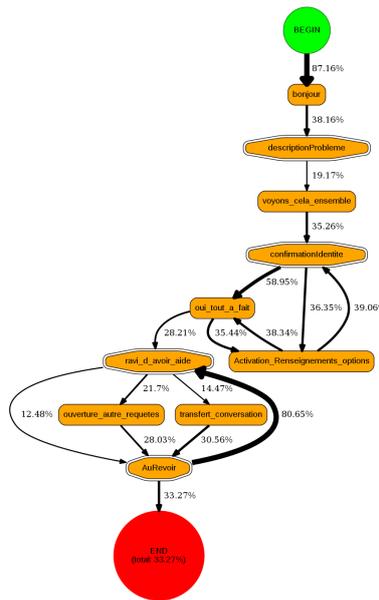


FIG. 5 – Graphe pour conception d'architecture

clusters ; ce second critère dépend fortement du premier. L'évaluation est en cours par les auteurs ainsi que par une ergonomiste.

## 5 Illustration lors de l'atelier

Lors de l'atelier une présentation plus détaillée des résultats obtenus sera effectuée pour illustrer concrètement notre approche. Nous présenterons les différents corpus utilisés jusqu'ici, les clusters obtenus, et la manière d'en étudier le contenu en regard des corpus initiaux. Nous présenterons aussi les graphes obtenus en fonction des fréquences de succession, et l'interface que nous utilisons pour les manipuler afin d'en déduire une architecture de systèmes de dialogues.

## 6 Conclusion

Dans cet article, nous avons présenté une méthode non supervisée pour obtenir une première version d'une architecture d'un système de dialogue, indépendamment du domaine couvert par celui-ci. La méthode comporte deux grandes étapes : d'une part l'application d'un algorithme de CoClustering sur un corpus de dialogues appartenant au domaine ciblé ; d'autre part, la prise en compte de la séquentialité des clusters obtenus pour représenter le déroulement prototypique d'un dialogue.

Il reste de nombreux verrous scientifiques autour de cette approche. Plusieurs sont relatifs à la technique même de clustering. Nous pensons notamment à la problématique de la thématisation des clusters : comment déterminer les mots ou tours de paroles les plus représentatifs du cluster ? La question de la sélection des clusters les plus pertinents, c'est à dire les plus homogènes au regard d'une thématique donnée, se pose aussi, en corrélation avec la granularité/taille optimale des clusters. Il serait également intéressant d'étudier l'efficacité d'optimisations linguistiques du corpus pour diminuer le nombre de paramètres utilisés pour le clustering : par exemple lemmatisation des mots, neutralisation des Entités Nommées.

En ce qui concerne la modélisation de la succession des clusters, de nombreuses problématiques se posent aussi. Nous avons vu que dans la littérature, c'est souvent un HMM qui est utilisé à cette fin ; mais nous pensons qu'il serait intéressant d'étudier d'autres approches telles que celles des Conditional Random Fields (CRF).

Enfin, la question se pose de la capacité de réutilisabilité des informations obtenues (notamment les classes et thèmes) d'un domaine à l'autre. Un déploiement de ces informations vers des domaines proches du domaine initial est faisable et prévu dans la suite Khiops. Pour des domaines plus éloignés, une approche relevant de l'apprentissage par transfert (*transfer learning*) est envisageable.

## Références

- Bangalore, S., G. DiFabrizio, et A. Stent (2008). Learning the structure of task-driven human-human dialogs. *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*, 201–208.
- Boullé, M., R. Guigourès, et F. Rossi (2014). Analyse exploratoire par k-coclustering avec khiops coviz. In *Advances in Knowledge Discovery and Management*, Volume 527, pp. 15–35.

- Chalamalla, A., S. Negi, S. Joshi, et L. V. Subramaniam (2008). Identification of class specific discourse patterns. *CIKM '08*.
- Chotimongkol, A. (2008). *Learning the Structure of Task-Oriented Conversations from the Corpus of In-Domain Dialogs*. Phd thesis, Carnegie Melon University.
- D'Haro, L. F., R. Cordoba, J. M. Lucas, R. Barra-Chicote, et R. San-Segundo (2009). Speeding up the design of dialogue applications by using database contents and structure information. *SIGDIAL 2009*, 160–169.
- Guerraz, B., M. Boullé, D. Gay, V. Lemaire, et F. Clérot (2015). Analyse exploratoire par k-coclustering avec khiops coviz. In *Atelier CluCo, Extraction et Gestion des Connaissances (EGC)*.
- Laroche, R. (2015). Speeding up the design of dialogue applications by using database contents and structure information. *18th International Conference on Intelligence in Next Generation Networks*, 231–238.
- Negi, S., S. Joshi, A. Chalamallay, et L. V. Subramaniam (2009). Automatically extracting dialog models from conversation transcripts. *2009 Ninth IEEE International Conference on Data Mining*.
- Paul, M. (2012). Mixed membership markov models for unsupervised conversation modeling. *EMNLP-CoNLL '12*, 231–238.
- Vinyals, O. et V. L. Quoc (2015). A neural conversational model. *International Conference on Machine Learning*, 231–238.
- Zhai, K. et J. Williams (2015). Discovering latent structure in task-oriented dialogues. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 36–46.

## Annexe

Nous remercions Nicolas Voisine, qui nous a fourni Patatext, l'outil de prétraitement de texte pour Khiops, ainsi que Fabien Dupont et Laurent Roussarie pour leur participation à la chaîne de clustering et de visualisation des graphes, et pour leur aide à l'utilisation.

Nous remercions également Nathalie Legay pour sa participation à l'étude des graphes obtenus.

Nous remercions enfin Romain Laroche, qui a contribué à la rédaction de l'introduction de cet article.

## Summary

The paper describes an on-going industrial process of research / development. Our goal is to obtain an unsupervised modeling of the structure of task-oriented dialogues. The dialogues are related to a given domain (for instance, train booking). This modeling presents several advantages; notably its use as a basis for conceiving a conversational agent architecture. The modeling is represented by a graph. It displays the main stages of the dialogues and the transitions between them. Our approach consists in applying the coclustering to the

## Aide à l'automatisation de conception de systèmes de dialogue

targeted dialogue corpus. Thus we obtain the main themes that appear in the corpus. We then compute the theme transitions within each dialogue. The resulting graph describes the main themes in the corpus, and their overall sequential organization. We describe the details of the process that we set up, and our position in regard to the related works. We eventually present the remaining scientific problems to address.