

Gestion de dialogue oral Homme-machine en arabe

Younès Bahou (1), Amine Bayoudhi (2), Lamia Hadrich Belguith (3)

(1) bahou_younes@yahoo.fr, (2) bayoudhi.amine@gmail.com,
(3) l.belguith@fsegs.rnu.tn

Laboratoire LARIS-MIRACL, FSEGS – Université de Sfax, Tunisie.

Résumé. Dans le présent papier, nous présentons nos travaux sur la gestion du dialogue oral arabe Homme-machine. Ces travaux entrent dans le cadre de la réalisation du serveur vocal interactif SARF (Bahou et al., 2008) offrant des renseignements sur le transport ferroviaire tunisien en langue arabe standard moderne. Le gestionnaire de dialogue que nous proposons est basé sur une approche structurale et est composé de deux modèles à savoir, le modèle de tâche et le modèle de dialogue. Le premier modèle permet de *i*) compléter et vérifier l'incohérence des structures sémantiques représentant les sens utiles des énoncés, *ii*) générer une requête vers l'application et *iii*) récupérer le résultat et de formuler une réponse à l'utilisateur en langage naturel. Quant au modèle de dialogue, il assure l'avancement du dialogue avec l'utilisateur et l'identification de ses intentions. L'interaction entre ces deux modèles est assurée grâce à un contexte du dialogue permettant le suivi et la mise à jour de l'historique du dialogue.

Abstract. In this paper, we present our research work on Human-machine Arabic oral dialogue management. This work enters in the context of SARF system (Bahou et al., 2008) an interactive vocal server that provides information on Tunisian railway using modern standard Arabic. The dialogue manager that we propose is based on a structural approach and consists of two models namely, the task model and the dialogue model. The first model is used to *i*) complete and verify the incoherence of semantic structures representing the useful meaning of utterances, *ii*) generate a query to the application and *iii*) get back the results and formulate an answer to the user in natural language. As for the dialogue model, it assures the dialogue progress with the user and the identification of her or his intentions. The interaction between these two models is assured by a dialogue context that allows monitoring and updating the dialogue history.

Mots-clés: gestion du dialogue Homme-machine, dialogue oral arabe, modèle de tâche, modèle de dialogue.

Keywords: Human-machine dialogue management, Arabic oral dialogue, task model, dialogue model.

1 Introduction

L'objectif des Systèmes de Dialogue Oral Homme-Machine (SDOHM) est de rendre la machine capable de dialoguer d'une manière naturelle avec l'Homme. Pour ce faire, ces systèmes doivent être dotés d'un gestionnaire de dialogue qui occupe une place très importante dans ce type de systèmes. Le domaine de la gestion de dialogue oral en langues Latines a bénéficié de plusieurs travaux de recherche permettant le développement et l'évolution de ce domaine. Cependant, la gestion de dialogue en langue arabe n'a pas eu le

même sort. Ce retard est expliqué d'un côté par la complexité et la diversité dialectale de la langue arabe et d'un autre côté par la manque d'outils aidant à la réalisation de ces systèmes (i.e., corpus, système de reconnaissance vocale, etc.)

Pour cela, nous avons fixé comme objectif la conception et la réalisation d'un gestionnaire de dialogue en langue arabe. Nous projetons de mettre notre système à la disposition des chercheurs afin d'encourager les travaux sur ce domaine et d'aboutir au développement des SDOHM destinés à la communauté arabe. Nous avons choisi la langue Arabe standard moderne afin de viser toute la communauté arabe vu qu'elle peut être parlée par tous les arabes. Cependant, les dialectes arabes sont très variés et diffèrent d'un pays à un autre, voire même d'une région à une autre au sein du même pays.

Ce travail entre dans le cadre de la réalisation du serveur vocal interactif SARF (Bahou et al., 2008) offrant des renseignements sur le transport ferroviaire tunisien en langue arabe standard moderne. Nous visons, à travers le présent travail, à appliquer et à tester l'approche structurelle dans notre méthode de gestion de dialogue en arabe.

Cet article s'articule autour de deux parties principales. La section 2 présente un bref aperçu des approches de gestion de dialogue. La section 3 détaille la méthode que nous proposons pour la gestion de dialogue en arabe.

2 Principales approches de gestion de dialogue

La gestion de dialogue cherche à assurer une communication fiable et efficace avec l'utilisateur. Cette communication, pour atteindre ces qualités, doit être bien entretenue et bien modélisée, problème majeur dans la gestion du dialogue. Dans cette section, nous présentons les solutions proposées pour résoudre ce problème à travers l'illustration des principales approches de gestion de dialogue.

Notons que plusieurs méthodes et de nombreuses classifications de ces dernières ont été citées dans la littérature. Nous proposons de classer les différentes méthodes en quatre approches : l'approche structurelle, l'approche intentionnelle, l'approche orientée plan et l'approche conventionnelle.

2.1 Approche structurelle

Cette approche prend son origine du modèle genevois (Roulet et al., 1991) à partir duquel plusieurs travaux ont été inspirés et ont apporté en contrepartie des améliorations remarquables en offrant à ce modèle la dynamique qui lui manque. Parmi ces travaux, nous citons ceux de Luzzati (Luzzati, 1989), de Rosset et Lamel (Rosset, Lamel, 2001) et de Minker et Bennacef (Minker, Bennacef, 2004).

L'idée fondamentale sur laquelle se base l'approche structurelle est l'existence d'une structure à l'intérieur de laquelle le dialogue peut se tisser (Caelen, Xuereb, 2007). Cette structure décrit autour d'elle des unités hiérarchiques plus complexes entretenant des relations fonctionnelles. La définition de ces deux niveaux d'analyse hiérarchique et fonctionnelle donne naissance à une grammaire hors-contexte décrivant les régularités des échanges dans le dialogue.

Un des modèles les plus importants dans l'approche structurelle, est celui proposé par Luzzati (Luzzati, 1989). Ce modèle constitue une extension très intéressante du modèle genevois dans le sens où il a introduit l'aspect dynamique avec l'utilisation de deux axes d'analyse. Le

premier est un axe *régissant* qui permet de faire progresser la tâche quand cela est possible et le deuxième est un axe *incident* qui permet de traiter les incidences telles que les demandes de précision, d'explication, de confirmation ou de reformulation.

Les modèles de l'approche structurelle se basent sur l'idée de l'existence d'une grammaire générative capable de décrire le dialogue à partir des marqueurs linguistiques et des indices pragmatiques. Ainsi, n'importe quel dialogue peut être soumis aux règles imposées par cette structure descriptive. Or, on est sensé modéliser un dialogue faisant intervenir deux interlocuteurs, donc il n'y a pas de raison pour percevoir leurs interventions d'une manière identique et objective, surtout que l'un des deux interlocuteurs est une machine. D'autre part, l'objectif de la modélisation du dialogue est de décrire et étudier le fonctionnement des conversations, non de doter les systèmes informatiques d'une compétence conversationnelle.

2.2 Approche orientée plan

Contrairement à l'approche structurelle, l'approche fondée sur la planification ne s'est intéressée ni au dialogue lui-même ni à sa structure, mais elle s'est intéressée aux interlocuteurs produisant ce dialogue. En effet, les modèles de l'approche orientée plan considèrent que chaque interlocuteur s'engageant dans la conversation a un comportement rationnel qui l'amène à établir des plans lui permettant d'atteindre ses buts. Ces plans sont définis comme étant un ensemble d'actions ordonnées amenant à réaliser un but donné. L'idée centrale de l'approche orientée plan consiste donc à préétablir l'ensemble des plans possibles de la tâche puis à reconnaître le plan de l'interlocuteur parmi ces plans.

Les travaux effectués autour de cette approche sont inspirés de la théorie de planification (Nilson, 1980) et nous pouvons citer à titre d'exemple les travaux de Rickel et al. sur le remplissage des pages Web multimodales (Rickel et al., 2001) et de Sidner et al. sur la réalisation d'un robot à guide (Sidner et al., 2003).

Mais bien que la planification soit une approche efficace pour guider l'utilisateur à franchir les obstacles qui l'empêchent de réaliser ses buts, elle reste une approche incapable d'aborder les problèmes rencontrés dans les systèmes de dialogue surtout au niveau de la modélisation des règles conversationnelles et surtout qu'il n'y a aucune distinction entre les actions liées à la tâche et celles à la communication. En plus, cette efficacité peut coûter très cher puisque les systèmes deviennent très rigides (i.e., on ne peut traiter que des dialogues qui sont dirigés par la tâche et dont le but est connu). Aussi, les incidences hors tâche du dialogue sont quasiment impossibles à traiter (Caelen, Xuereb, 2007). De même, l'hypothèse sur laquelle se base cette approche admet un nombre limité de plans prédéfinis à l'avance, ce qui n'est pas le cas réellement. Et même si c'est le cas, le nombre d'inférences utilisées dans le processus de reconnaissance des plans est élevé, ce qui peut ralentir les systèmes surtout avec la complexité des méthodes heuristiques.

2.3 Approche intentionnelle

L'approche intentionnelle est en réalité une extension de l'approche orientée plan. Seulement, elle a la faveur d'ajouter un niveau cognitif à la modélisation du dialogue car il ne suffit pas de viser l'organisation de la tâche et du discours pour bien adapter la réponse, mais il faut prendre en considération les intentions de l'interlocuteur. Ainsi, cette approche se développe autour de la logique intentionnelle évoluée avec l'apparition des modèles BDI (*Belief, Desire, Intention*) en intelligence artificielle.

Les premiers travaux dans ce courant sont ceux de Cohen, Perrault et Allen (Cohen, Perrault, 1979) (Allen, Perrault, 1980) qui sont ensuite repris et raffinés par Sadek dans le développement du système ARTIMIS/AGS pour la consultation de l'annuaire des services audiotel (emploi et météo) par téléphone (Sadek, 1996).

Les modèles de l'approche intentionnels ont bien poussé les résultats dans le domaine du dialogue Homme-machine puisqu'ils sont caractérisés par une modélisation assez complète. Le problème reste dans l'hypothèse de rationalité et dans la pertinence de la logique des modèles BDI vis-à-vis le comportement instable humain. Ce qui rend cette approche plus efficace et plus convenable pour les systèmes multi-agents que pour les humains. Une autre limite est au niveau de la mise en œuvre de la reconnaissance des intentions des interlocuteurs qui semble être très difficile et complexe.

2.4 Approche conventionnelle

L'approche conventionnelle est basée sur la théorie des jeux de dialogues conventionnels (Husltin, 2000). Cette théorie suppose que chacun des interlocuteurs est engagé dans un jeu social dont les tours de parole représentent les coups. L'enjeu se manifeste dans la pertinence de ces coups, c'est-à-dire, chacun va tenter essentiellement de maximiser ses gains et réduire ses pertes en suivant les règles du jeu. Le problème peut se ramener ainsi à la gestion des buts qui sont traités selon des stratégies prenant en considération l'état des interlocuteurs et l'état du dialogue.

Parmi les travaux les plus persistants dans cette approche, nous citons ceux de Rouillard pour le système HALPIN dédié à la recherche d'information en utilisant l'hyper-dialogue en Français (Rouillard, 2000) et les travaux de Nguyen pour le système MELINA pour l'organisation des réunions (Nguyen, 2005).

L'approche conventionnelle a connu aussi d'autres travaux qui ont été classés comme étant soit des modèles argumentatifs (Baker, 1994) ou soit orientés interaction (Dessales, 1993). Mais ces modèles ont été considérés explicatifs de l'interaction humaine et n'ont pas été adaptés dans le dialogue Homme-machine.

Les points forts de cette approche se situent surtout dans le calcul de pertinence qui permet d'évaluer l'adéquation d'un acte de dialogue par rapport au but du locuteur. Cependant, la définition des règles sociales reste un problème dans ces modèles car le comportement humain face à la machine n'obéit pas en général à ces règles (Caelen, Xuereb, 2007).

3 Méthode proposée pour la gestion de dialogue

Dans cette section nous proposons une méthode pour la gestion de dialogue oral Homme-machine utilisant la langue arabe standard moderne. Ainsi, la méthode proposée est inspirée des premiers travaux de Roulet (Roulet et al., 1991) qui ont fondu les principes des analyses hiérarchiques et fonctionnelles pour rendre compte des régularités des échanges dans les dialogues. Cette méthode se base sur l'approche structurelle et permet de gérer l'interaction avec les utilisateurs et de répondre à leurs demandes d'informations.

Avant d'exposer les étapes de notre méthode, nous tenons à signaler que nous avons utilisé notre propre corpus d'étude construit selon la technique du Magicien d'Oz. Ce corpus nous a permis le recensement des actes de dialogues, des marqueurs linguistiques et l'ensemble des règles de réécriture du dialogue composant notre grammaire formelle. Le tableau suivant résume quelques informations sur le corpus d'étude que nous avons utilisé.

Nombre de locuteurs	Nombre de dialogues	Taille en heures	Nombre de mots	Nombre d'énoncés
50	300	11	92598	7590

Tableau 1 : Caractéristiques du corpus d'étude utilisé

La méthode proposée est constituée principalement par deux modèles : modèle de tâche et modèle de dialogue (voir figure 1). Le premier modèle s'occupe d'exécuter les tâches confiées à la machine à savoir interpréter et compléter les demandes des utilisateurs, formuler les requêtes à communiquer à la base de données de l'application et générer les réponses à l'utilisateur. Tandis que le deuxième prend en charge toutes les interactions avec les utilisateurs en identifiant leurs actes de dialogue, en décidant la nature de l'intervention machine adéquate et en assurant le suivi et l'avancement du dialogue en cas d'incidence ou de rupture. L'interaction entre ces deux modèles est assurée grâce à un contexte du dialogue permettant le suivi du dialogue et la mise à jour de l'historique.

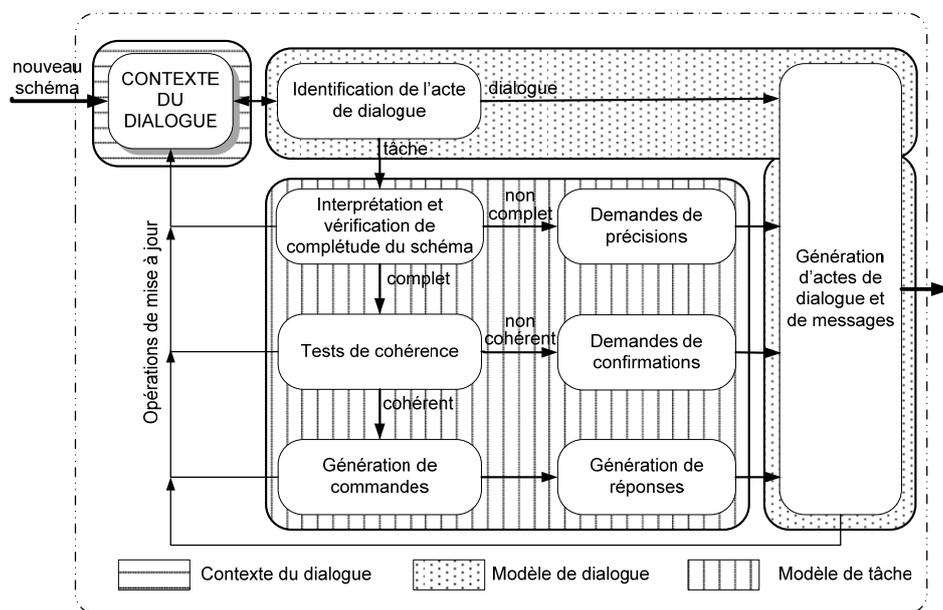


Figure 1 : Les étapes de la méthode proposée pour la gestion de dialogue

Nous présentons dans ce qui suit le fonctionnement de chacun de ces deux modèles ainsi que les ressources requises dans chaque étape de la méthode proposée.

3.1 Le modèle de tâche

Le rôle principal du modèle de tâche est de compléter la structure sémantique (un schéma dans notre cas) générée par le système de compréhension et de vérifier sa cohérence afin d'en extraire les informations nécessaires à la génération des requêtes vers l'application. Ensuite, il exploite les informations retournées par l'application pour formuler des réponses vers l'utilisateur. Néanmoins, la validation du schéma sémantique reste la tâche la plus délicate du modèle de tâche. En effet, un schéma doit passer par cinq étapes de validation avant d'être jugé comme un schéma complet et valide. Ces étapes sont (voir figure 2) : l'interprétation, l'instanciation des attributs à partir de l'historique, l'instanciation des attributs par des valeurs par défaut, la vérification de la cohérence et la vérification de la complétude.

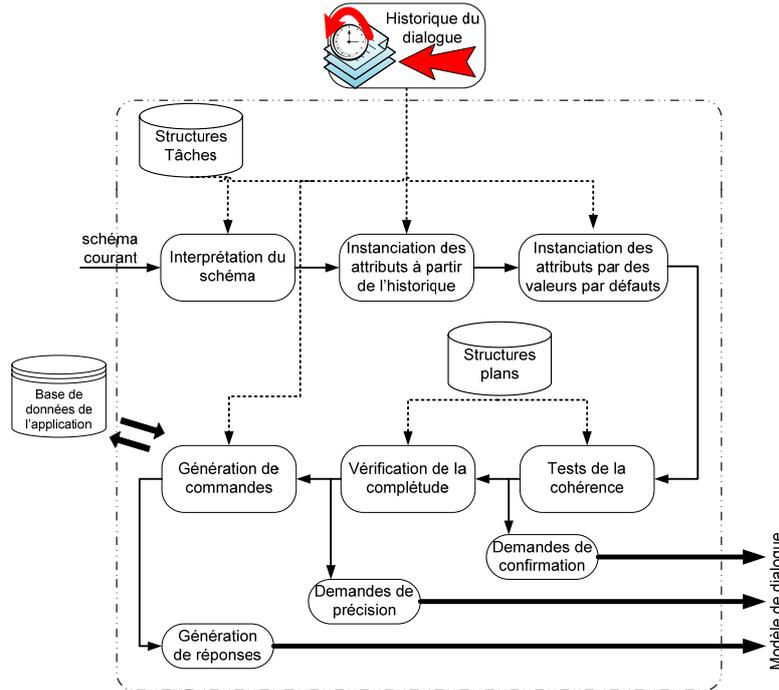


Figure 2 : Représentation détaillée du modèle de tâche

Dans ces étapes, le modèle de tâche utilise deux principales ressources à savoir, les structures *tâches* et les structures *plans* (Minker, Bennacef, 2004). Les structures tâches (voir figure 3.a) contiennent les règles relatives à l'interprétation, la formulation des commandes et la génération de réponses. Quant aux structures plans (voir figure 3.b), elles regroupent les règles de cohérence et les règles de complétude des schémas.

Tâche horaire	Plan horaire
<p>INTERPRETATIONS</p> <p>if(ville_arrivee="عاصمة" [EASmp] (capitale)) Then(ville_arrivee="تونس" [twns] (Tunis))</p> <p>COMMANDES</p> <p>SELECT heure_depart, heure_arrivee FROM voyage WHERE ville_depart=@ville_depart AND ville_arrivee=@ville_arrivee AND ...</p> <p>REPONSES</p> <p>if (resultat) Then</p> <p>(... مواعيد انطلاق القطار هي كالاتي: ...) [mwAEyd AnTIAq AlqTAr hy kAl/ty: ...] (Les horaires de départ du train sont comme suit:...)</p>	<p>PLANS</p> <p>if(!ville_arrivee) Then</p> <p>ville_arrivee : (إلى أي مدينة تريد السفر ؟) [IY Oy mdynp tryd Alsfr] (C'est quoi votre destination ?)</p> <p>if(! heure_depart_relatif) Then</p> <p>heure_depart_relatif : (في أي وقت تريد السفر ؟) [fy Oy wqt tryd Alsfr] (Dans quelle heure voulez-vous voyager ?)</p> <p>REGLES DE COHERENCE</p> <p>if(ville_depart=@ville_arrivee) Then</p> <p>(مدينة الوصول هي نفسها مدينة الانطلاق) [mdynp AlwSwl hy nfshA mdynp AlAnTIAq] (La ville de départ est la même que celle d'arrivée)</p>

a- Structure tâche horaire simplifiée

b- Structure plan horaire simplifiée

Figure 3 : Représentations simplifiées des structures tâche et plan horaire

Dans les sections suivantes, nous exposons en détail les étapes du modèle de tâche. Afin d'expliquer chaque étape, nous prenons l'exemple du schéma horaire de la figure 4.a obtenu par notre système de compréhension suite à la demande d'un utilisateur l'heure de départ du premier "أول" [Owl] train express "سريع" [sryE] à destination de la capitale "عاصمة" [EASmp].

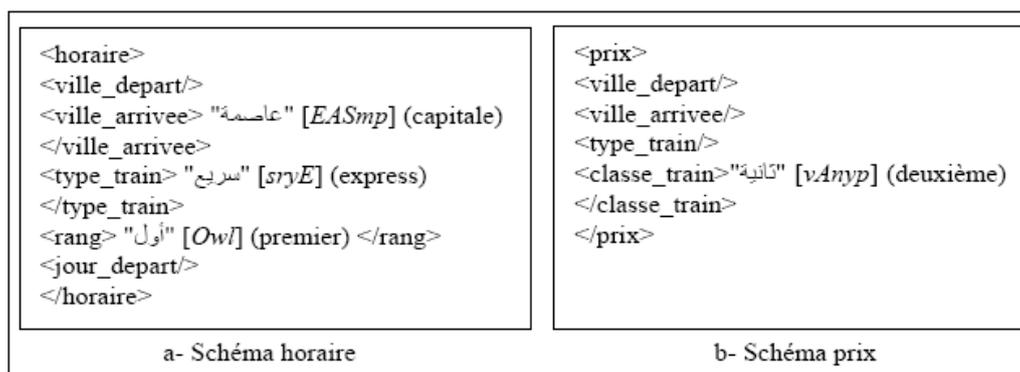


Figure 4 : Schémas horaire et prix générés par notre système de compréhension suite à deux énoncés du même dialogue

- **Interprétation du schéma sémantique :** le schéma sémantique généré par notre système de compréhension peut contenir plusieurs expressions qui nécessitent des reformulations pour qu'elles soient acceptables par le SGBD. Ainsi, une étape d'interprétation du schéma s'avère nécessaire pour reformuler et standardiser ces expressions. Cette étape utilise un ensemble de règles d'interprétation dégagées à partir de notre corpus d'étude. Ces règles sont regroupées dans les structures tâches.

Prenons à titre d'exemple le schéma horaire de la figure 4.a, la valeur assignée à l'attribut ville d'arrivée est "عاصمة" [EASmp] (capitale) nécessite une interprétation afin d'être changée en "تونس" [twns] (Tunis). Aussi, la valeur du rang du train "أول" [Owl] (premier) sera convertie en nombre 1.

- **Instanciation des attributs à partir de l'historique :** étant donné la continuité de l'aspect dialogique, l'utilisateur peut omettre quelques détails qu'il a déjà mentionnés dans ses interventions précédentes. Ce qui se traduit au niveau de l'énoncé sous forme d'ellipses et d'anaphores et entraîne la présence d'attributs avec des valeurs vides dans le schéma sémantique. Cette étape vise à résoudre les ellipses et les anaphores par l'utilisation de l'historique du dialogue afin de rechercher des instances possibles aux attributs vides. Dans le cas où la recherche était fructueuse, l'instance trouvée est automatiquement accordée à l'attribut en question.

Prenons à titre d'exemple le schéma prix illustré par la figure 4.b. Ce schéma comporte des attributs non instanciés par le système de compréhension comme la ville de départ, la ville d'arrivée, le type du train et le jour du voyage. Ces informations sont volontairement omises par l'utilisateur vu qu'il les a déjà précisées lors de sa question précédente (schéma horaire de la figure 4.a). Ainsi, dans cette étape, le schéma prix sera complété à partir du schéma horaire déjà complété, validé et stocké dans l'historique du dialogue puisque ces deux schémas font partie du même dialogue.

- **Instanciation des attributs par des valeurs par défaut :** une deuxième cause qui explique l'omission de l'utilisateur de certains détails dans sa demande, est que celui-ci les considère comme étant évidents. Pour cela, cette étape permet d'instancier quelques attributs par des valeurs par défaut (i.e., connues d'avance). Ces attributs et leurs valeurs sont recensés à partir du corpus d'étude et sont stockés dans les structures tâches.

Par exemple, l'attribut jour de départ du schéma horaire de la figure 4.a peut être instancié par la valeur du jour système. Aussi, l'attribut ville de départ peut être instancié par la valeur de la ville par défaut (i.e., le système est installé dans la station de la gare de cette ville).

- **Vérification de la cohérence** : cette vérification est nécessaire pour valider la cohérence des informations apportées par le schéma. Elle permet d’alerter l’utilisateur sur les erreurs de cohérence pouvant être commises par manque d’attention lors de la formulation de sa demande. La découverte de ces erreurs se fait grâce à un ensemble de règles de cohérence recensées à partir du corpus d’étude et stockées dans les structures plan. Dans le cas de présence des incohérences, des demandes de confirmation seront générées à l’utilisateur pour corriger les informations données à la machine.

A titre d’exemple par le schéma prix de la figure 4.b l’utilisateur cherche à savoir le prix d’un billet pour la deuxième classe dans le train express. Or, les trains express en Tunisie ne contiennent que la classe confort. Donc un message sera généré à l’utilisateur pour l’alerter de l’erreur.

- **Vérification de la complétude** : Cette étape permet de vérifier l’existence, dans le schéma courant, de toutes les informations nécessaires permettant la génération d’une requête valide vers l’application. Dans le cas échéant, des demandes de précisions seront générées vers l’utilisateur afin qu’il puisse compléter les informations qu’il a oubliées. Pour cela, chaque tâche est dotée d’un plan contenant une liste ordonnée des attributs nécessaires et des messages de demande de précisions des attributs omis par l’utilisateur.

Une fois le schéma sémantique complété, une requête SQL est générée vers la base de données de l’application afin de dégager les informations recherchées par l’utilisateur. Ces informations sont ensuite reformulées sous forme des réponses en langage naturel et envoyées à l’utilisateur. Notons que les interventions de la machine peuvent être aussi sous forme de demandes de confirmations qui ne sont pas directement communiquées à l’utilisateur mais elles passent à travers le modèle de dialogue que nous exposons dans la section suivante.

3.2 Le modèle de dialogue

Le modèle de dialogue s’occupe des interactions avec l’utilisateur (voir figure 5). Il permet de gérer le dialogue et d’assurer sa continuation en identifiant les intentions et en prévoyant les solutions convenables pour les situations critiques comme la rupture de dialogue, Il permet aussi de communiquer avec le modèle de tâche afin de récupérer les questions ainsi que les réponses à générer à l’utilisateur.

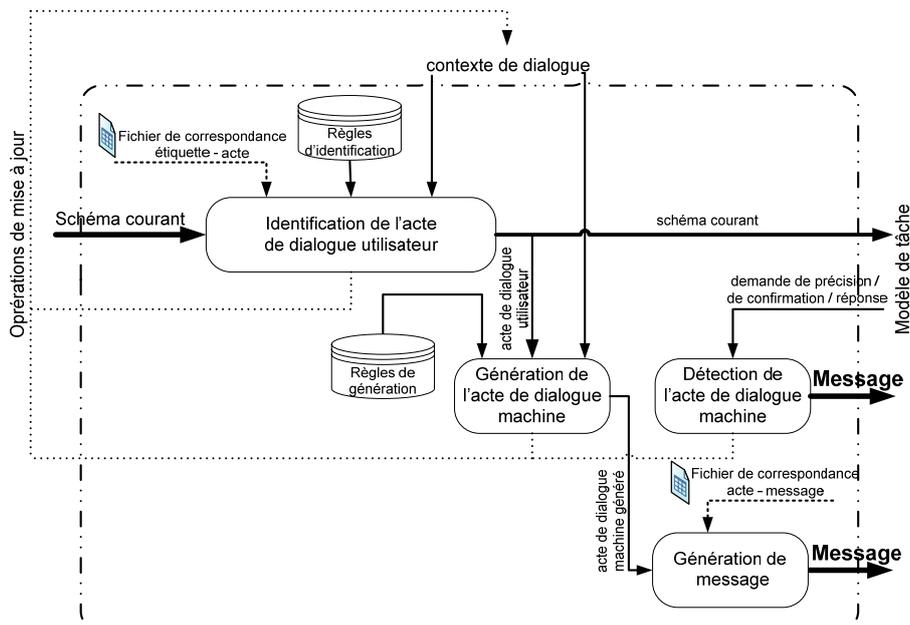


Figure 5 : Représentation détaillée du modèle de dialogue

Le rôle principal du modèle de dialogue est de reconnaître l'acte de dialogue, qui constitue la plus petite unité du dialogue, dans l'intervention de l'utilisateur. L'identification des actes est effectuée grâce à des marqueurs linguistiques et des indices pragmatiques que nous avons recensés à partir de notre corpus d'étude et qui sont stockés dans un fichier de correspondance étiquette – acte. Ce fichier regroupe un ensemble de paires *étiquette sémantique / acte de dialogue* utiles lors de l'identification des actes de dialogue utilisateur.

Les actes de dialogue sont regroupés dans des structures plus complexes appelés les sous-dialogues et sont ensuite rassemblés dans une grammaire formelle décrivant l'ensemble des règles de réécriture du dialogue. Cette grammaire permet de poursuivre et de contrôler l'évolution du dialogue pour faciliter l'identification des actes d'une part et pour détecter les incidences d'autre part.

En plus des marqueurs linguistiques et de la grammaire formelle, le modèle de dialogue utilise des règles pour l'identification des actes de dialogue utilisateur. Par exemple, Si le dernier acte de dialogue est un acte de demande de précision, de reformulation ou de confirmation alors l'acte de dialogue courant est une réponse.

Selon l'acte de dialogue utilisateur identifié, le modèle de dialogue décide du traitement que doit subir le schéma sémantique provenant du système de compréhension (voir figure 5). Si l'acte de dialogue est une formalité d'ouverture ou de fermeture de dialogue ou un acte de relance, alors le modèle de dialogue génère l'acte machine et le message correspondant en utilisant un fichier de correspondance acte-message. Dans le cas où l'acte identifié est différent de ces trois actes, le schéma sémantique est envoyé vers le modèle de tâche pour être traité comme nous l'avons expliqué dans la section 3.1.

4 Conclusion

Dans ce papier, nous avons proposé une méthode pour la gestion de dialogue oral en arabe standard moderne. Cette méthode est basée sur l'approche structurale et repose sur deux modèles à savoir, le modèle de tâche et le modèle de dialogue. Le modèle de tâche permet d'interpréter et de compléter les demandes des utilisateurs, formuler les requêtes à communiquer à la base de données de l'application et générer les réponses à l'utilisateur. Le modèle de dialogue permet l'identifiant des actes de dialogue, décide la nature de l'intervention machine adéquate et assure le suivi et l'avancement du dialogue en cas d'incidence ou de rupture.

Comme perspectives, nous envisageons d'étendre les deux modèles de dialogue pour couvrir certains problèmes non résolus par notre système de compréhension à savoir, le traitement des disfluences complexes et imbriquées, ainsi que le traitement des mots considérés comme hors-vocabulaire par le système de compréhension.

Références

Allen J. F., Perrault C. R. (1980). Analysing intention in dialogues. *Artificial Intelligence*, volume 15, N°3, Pages 23-46.

Bahou Y., Belguith Hadrich L., Ben Hamadou A. (2008). Towards a Human-Machine Spoken Dialogue in Arabic. *LREC'08, Workshop HLT within the Arabic World: Arabic Language and local languages processing Status Updates and Prospects*, Marrakech, Maroc.

- Baker M. (1994). A model for negotiation in teaching-learning dialogues. *Journal of Artificial Intelligence in Education*, Volume 5, N°2, Pages 199-254.
- Caelen J., Xuereb A. (2007). *Interaction et pragmatique: jeux de dialogue et de langage*. Paris : Hermès-Lavoisier.
- Cohen P. R., Perrault C. R. (1979). Elements of a plan based theory of speech acts. *Cognitive science*, volume 3, Pages 177-212.
- Dessales J. L. (1993). Modèle cognitif de la conversation spontanée appliqué à l'apprentissage. *Thèse de doctorat*, ENST, Paris, France.
- Husltin J. (2000). Dialogue model for inquiry and transaction. *Thèse de doctorat*, Université de Twente, Pays-Bas.
- Luzzati D. (1989). Recherches sur le dialogue homme-machine: modèles linguistiques et traitement automatique. *Thèse d'Etat*, Paris III, France.
- Minker W., Bennacef S. (2004). *Speech And Human-Machine Dialog*. Pays-Bas : Kluwer Academic Publishers Group.
- Nguyen H. (2005). Dialogue homme-machine : modélisation de multisessions. *Thèse de doctorat*, Université Joseph Fourier, France.
- Nilson N. J. (1980). *Principles of artificial intelligence*. San Francisco : Morgan Kaufmann Publishers Inc.
- Rickel J., Lesh N., Rich C., Sidner C. L., Gertner A. (2001). Using a model of collaborative dialogue to teach procedural tasks. *Workshop on Tutorial Dialog Systems, AIED*, San Antonio, Texas, USA.
- Rosset S., Lamel L. (2001). Gestionnaire de dialogue pour un système d'informations à reconnaissance vocale. *TALN'01*, Tours, France.
- Rouillard J. (2000). Hyperdialogue sur Internet: Le système HALPIN. *Thèse de doctorat*, Université Grenoble I, France.
- Roulet E., Auchlin A., Moeschler J., Rubattel C., Schelling M. (1991). *L'articulation du discours en français contemporain*. 3^{ème} éd. Berne : Peter Lang.
- Sadek D. (1996). Le dialogue homme-machine: de l'ergonomie des interfaces à l'agent intelligent dialoguant. *OFTA, ARAGO 18 Nouvelles interfaces homme-Machine*, Pages 277-321.
- Sidner C. L., Lee C. H., Lesh N. (2003). The Role of Dialog in Human Robot Interaction. *International Workshop on Language Understanding and Agents for Real World Interaction*, Sapporo, Japan.