

## CFAsT: Content-Finder AssistanT

Romain Laroche

Orange Labs, 38-40 avenue du Général Leclerc, 92130 Issy-les-Moulineaux / France

romain.laroche@orange.com

**Résumé.** Cette démonstration de CFAsT s'intéresse à "comment concevoir un système de dialogue avec un effort minimal". Cet assistant virtuel repose sur un nouveau modèle pour la génération automatique de système de dialogue construite à partir de contenus. Cette approche utilise un moteur de recherche auquel on a ajouté des fonctionnalités de dialogue : à chaque tour, le système propose trois mots-clés de manière à optimiser l'espérance de gain d'information.

**Abstract.** This CFAsT demonstration focuses on "how to design and develop a dialogue system with a minimal effort". This virtual assistant embeds a novel model for automatic generation of dialogue systems built from contents. This approach is similar to and relies on a search engine, but with augmented dialogue capabilities : at each dialogue turn, the system propose three keywords, in order to optimise the information gain expectation.

**Mots-clés :** Systèmes de dialogue, Traitement automatique des langues naturelles, Assistant virtuel.

**Keywords:** Dialogue systems, Natural language processing, Virtual assistant.

### Approche

Souvent, la recherche sur les systèmes de dialogue s'intéresse à "comment concevoir un système de dialogue avec un effort minimal". C'est le cas notamment des approches dirigées par les données (Levin *et al.*, 1998), basées sur de l'apprentissage par renforcement à partir des dialogues de la machine avec des utilisateurs (réels ou simulés). Ces dernières années, ces approches ont connu une grande quantité d'améliorations, aussi bien au niveau des techniques d'apprentissage (Pietquin *et al.*, 2011) que de la méthodologie (El Asri *et al.*, 2012). Et pourtant, l'apprentissage par renforcement n'a pu être utilisé dans les systèmes de dialogue industriels, que lorsque l'on pouvait le projeter dans un système basé sur des automates (Putois *et al.*, 2010), parce que la plupart des réserves exprimées (Paek & Pieraccini, 2008) sont toujours d'actualité.

Notre approche contourne ces réserves en basant son modèle sur les contenus, plutôt que sur les interactions avec l'utilisateur, pour générer un système de dialogue. Le Content Finder AssistanT (CFAsT) est un outil permettant l'implantation automatique de système de dialogue très simples. Comme l'approche CFAsT se base sur les contenus, les applications générées peuvent être comparées aux moteurs de recherche, mais la méthodologie n'est pas le même : le moteur de recherche cherche les meilleurs contenus étant donnée une requête utilisateur, tandis que le CFAsT aide l'utilisateur à formuler sa requête pour que les réponses soient les plus pertinentes possibles.

Le flux de dialogue est illustré par la figure 1. L'utilisateur entre dans une boucle de dialogue dans laquelle l'historique de dialogue est la concaténation de toutes ses requêtes, augmentée de l'ensemble des contenus qu'il a déjà visionnés (et qui ne l'ont donc pas satisfait). A chaque tour de dialogue, le CFAsT utilise un moteur de recherche standard pour proposer les cinq contenus les plus pertinents et donner des statistiques (nombre de contenus répondant à la requête). Mais il utilise également un algorithme de calcul d'entropie pour évaluer l'espérance de gain d'information suite aux possibles actions de l'utilisateur quand le système lui proposera un ensemble de trois mots-clés. De cette manière le système choisira les mots-clés maximisant cette espérance de gain d'information. Si l'utilisateur sélectionne un contenu, il est amené à dire s'il répond à sa requête.

Placer l'utilisateur dans le cadre d'un dialogue permet de recueillir ses retours quant à la pertinence des contenus qu'on lui propose. Cette information est ensuite utilisée en ligne pour optimiser les futures recommandations. Par ailleurs, l'application générée automatiquement peut être surchargée par des branches de dialogue développées manuellement, des stratégies de désambiguïsation, des informations lexicales (synonymie, mots outils) ou encore des comportements standards face aux questions hors périmètre.

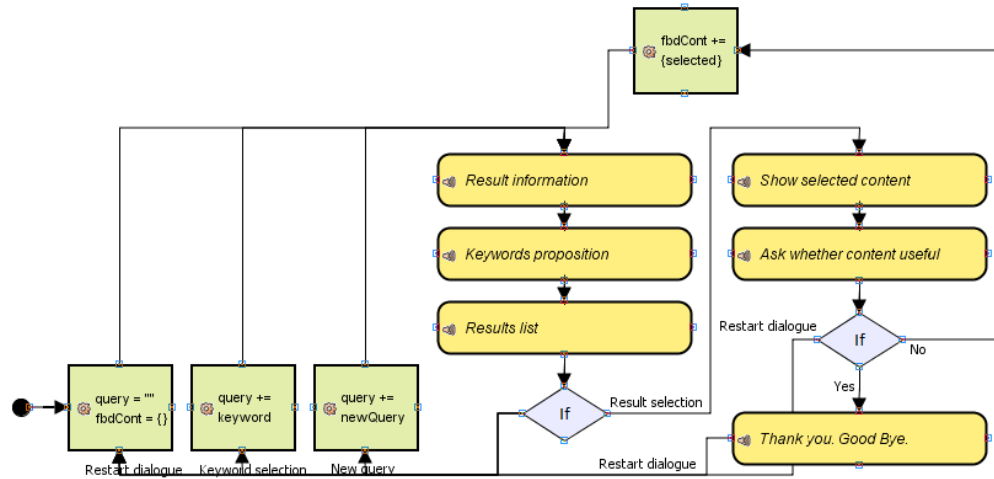


FIGURE 1 – Flux de dialogue dans le CFAsT.

## Discussion

A ce jour, une dizaine d'applications ont été générées grâce à cet outil au stade de preuves de concept, en français et en anglais, sur des corpus plus ou moins volumineux (de 160 à 7k contenus et de 818 à 30k termes), et sur des données plus ou moins structurées (pages web, articles de proceedings, FAQ, documents de bureautique, mails, vidéos, fiches de restaurant, ...). Les applications à faible coût ainsi obtenues sont générées après moins d'une heure de travail.

Depuis un moteur de recherche amélioré, surchargé de quelques stratégies de dialogue, jusqu'à un assistant conversationnel utilisant un moteur de recherche comme parachute en cas d'incompréhension de l'entrée utilisateur, tout le spectre est possible. Contrairement aux systèmes de dialogue utilisant de l'apprentissage par renforcement, les outils du CFAsT ne prétendent pas générer des applications optimales, ni même plus performantes que celles développées manuellement, mais simplement de le faire à coût faible voire nul. Il est ensuite à la charge du développeur d'enrichir, ou non, le dialogue avec des stratégies manuelles suivant le besoin.

L'objectif de ce démonstrateur est de réaliser qu'il est possible de générer une application de dialogue simple automatiquement. Nos travaux futurs sur ce sujet porteront sur la saisie de requêtes multi-champs avec une topologie inter-champs optimisée grâce à un système de recommandation (Meyer & Fessant, 2011) et à l'expérience, et sur la proposition automatique de stratégies de dialogue au concepteur humain.

Disserto, la suite logicielle à l'aide de laquelle est développé le CFAsT est avant tout utilisée pour la conception de systèmes vocaux, et notre intention est d'étendre le modèle à du dialogue parlé. Il reste de sérieux obstacles pour atteindre cet objectif : à l'oral le système devra notamment trouver une certaine forme de concision.

## Références

- EL ASRI L., LAROCHE R. & PIETQUIN O. (2012). Reward function learning for dialogue management. In *Proceedings of STAIRS*, p. 95–106.
- LEVIN E., PIERACCINI R. & ECKERT W. (1998). Using markov decision process for learning dialogue strategies. In *Proceedings of ICASSP*.
- MEYER F. & FESSANT F. (2011). Reperio : A generic and flexible industrial recommender system. In *Proceedings of IEEE/WIC/ACM WI-IAT*, p. 502–505 : IEEE Computer Society.
- PAEK T. & PIERACCINI R. (2008). Automating spoken dialogue management design using machine learning : An industry perspective. *Speech Communication*, **50**, 716–729.
- PIETQUIN O., GEIST M. & CHANDRAMOHAN S. (2011). Sample efficient on-line learning of optimal dialogue policies with kalman temporal differences. In *Proceedings of IJCAI*, p. 1878–1883 : AAAI Press.
- PUTOIS G., LAROCHE R. & BRETIER P. (2010). Online reinforcement learning for spoken dialogue systems : The story of a commercial deployment success. In *Proceedings of SIGDIAL*, Tokyo (Japan).