Description d'un système de compréhension automatique de la parole

Salma Jamoussi, Kamel Smaïli, Jean-Paul Haton

Description d'un système de compréhension automatique de la parole

Salma Jamoussi, Kamel Smaïli et Jean-Paul Haton

LORIA/INRIA-Lorraine
615 rue du Jardin Botanique, BP 101, F-54600 Villers-lès-Nancy, France
Tél : int+ 33 3 83 59 30 00, Fax : int+ 33 3 83 27 83 19
{jamoussi, smaili, jph}@loria.fr

Résumé La compréhension automatique de la parole peut être considérée comme un problème d'association entre deux langages différents. En entrée, la requête exprimée en langage naturel et en sortie, juste avant l'étape d'interprétation, la même requête exprimée en terme de concepts. Un concept représente un sens bien déterminé. Il est défini par un ensemble de mots partageant les mêmes propriétés sémantiques. Dans cet article, nous proposons une méthode à base de réseau bayésien pour l'extraction automatique des concepts ainsi qu'une nouvelle approche pour la représentation vectorielle des mots. Cette représentation aide le réseau bayésien à regrouper les mots, construisant ainsi la liste adéquate des concepts à partir du corpus d'apprentissage. Nous conclurons cet article par la description d'une étape de post-traitement au cours de laquelle, nous étiquetons nos requêtes et nous générons les commandes SQL appropriées validant ainsi, notre approche de compréhension.

Mots clés Compréhension de la parole, concepts sémantiques, réseaux bayésiens, étiquetage sémantique, catégorisation automatique.

1 Introduction

Dans la littérature, plusieurs méthodes de compréhension de la parole ont été proposées. La plupart de ces méthodes se fondent sur des approches stochastiques de décodage conceptuel qui permettent d'approcher la compréhension automatique, réduisant ainsi le recours à l'expertise humaine. Cependant, ces méthodes nécessitent une étape d'apprentissage supervisé, ce qui signifie qu'il y a une étape antérieure d'annotation manuelle du corpus d'apprentissage [1,3,4]. Dans de telles approches fondées sur le décodage conceptuel, l'étape d'annotation consiste à segmenter les données d'apprentissage en des segments conceptuels représentant chacun un sens bien déterminé [1]. Il s'agit donc de trouver tout d'abord les différents concepts relatifs au corpus, de segmenter ensuite les phrases de ce corpus, de les étiqueter en utilisant les concepts trouvés et de procéder enfin à l'apprentissage automatique. Faire tout ce travail d'une façon manuelle constitue sans doute une phase fastidieuse et coûteuse. De plus, l'extraction manuelle est sujette à la subjectivité et aux erreurs humaines. Automatiser cette tâche permettra donc de réduire ou d'annuler l'intervention humaine et surtout de pouvoir réutiliser ce même procédé lorsqu'on change de contexte.

Dans cet article, nous commençons par décrire l'architecture générale de notre système de compréhension de la parole, basée sur l'approche proposée dans [4]. Ensuite, nous présentons une nouvelle approche pour extraire automatiquement les concepts sémantiques. Pour ce faire, nous utilisons un réseau bayésien pour la classification non supervisée, appelé AutoClass. Puis, nous exposons une nouvelle méthode pour la représentation vectorielle des mots afin de les regrouper pour former des concepts. Enfin, nous abordons la dernière étape du processus de compréhension, au cours de laquelle nous étiquetons les requêtes et nous générons les commandes SQL associées.

2 La compréhension automatique de la parole

Le problème de compréhension de la parole peut être vu comme un problème de mise en correspondance entre une chaîne de mots en entrée et une suite de mots dans un langage plus restreint véhiculant les idées principales d'une phrase. Il s'agit, dans un premier temps, d'associer les mots de la phrase en entrée du système à des messages dans un langage sémantique intermédiaire (souvent appelés concepts). Dans un second temps, afin de satisfaire la requête émise en entrée, on traduit les concepts obtenus en actions ou réponses et on parle dans ce cas de l'étape d'interprétation de la phrase.

L'entrée du système peut être donnée sous forme textuelle ou sous forme d'un signal de parole, sa sortie exprimée en tant qu'actions ou commandes n'est qu'une conversion d'une liste de concepts donnée par un module intermédiaire de traduction sémantique et fournissant le sens littéral de la phrase. Un concept est une classe de mots traitant d'un même sujet et partageant des propriétés communes. Par exemple, les mots *hôtel, chambre, auberge et studio* peuvent tous correspondre au concept "*hébergement*" dans une application touristique. Dans [4], les auteurs définissent un modèle général pour la compréhension automatique de la parole qui, en raison de sa simplicité et de son efficacité, a été repris dans plusieurs autres travaux [1,3]. Nous avons adopté la même architecture générale (voir figure 1), mais nous proposons des techniques différentes au sein de chacune de ses composantes.

Fig. 1. Architecture générale d'un système de compréhension automatique de la parole.

Dans notre travail, nous nous intéressons à une application de consultation de pages "Favoris" (Bookmarks en anglais). Pour ce faire, nous utilisons un corpus du projet européen MIAMM dont l'objectif est de construire une plate-forme de dialogue oral multimodale. Le corpus contient 71287 requêtes différentes exprimées en langue française. Chaque requête exprime une manière particulière d'interroger la base. Des exemples de ces requêtes sont donnés dans la table 1. Ces requêtes sont fournies au système de compréhension sous leur forme textuelle. Notre but est de fournir à la fin les requêtes SQL correspondantes qui, en les exécutant, répondront aux demandes des utilisateurs.

Tab. 1. Quelques exemples de requêtes du corpus MIAMM.

Montre-moi le contenu de mes favoris.
Je voudrais savoir si tu peux me prendre le contenu que j'aime.
Est-ce que tu veux me sélectionner les titres que je préfère.
Est-il possible que tu me passes le premier de mes favoris.
Te serait-il possible de m'indiquer quelque chose de pareil.
Tu peux faire voir uniquement décembre 2001.
Il faut que tu me présentes la liste que j'ai utilisée tôt ce matin.
Je te demande de me passer les chansons que j'ai écoutées ce matin.

3 Extraction automatique des concepts

Au cours de cette étape, nous cherchons à identifier les concepts sémantiques liés à notre application. La détermination manuelle de ces concepts est une tâche très lourde. Il nous faut donc trouver une méthode automatique qui, pourrait ne pas donner des résultats aussi performants que ceux obtenus par la méthode manuelle, mais qui, en contre partie, permet une automatisation complète du processus de compréhension.

Partant du principe d'automatisation de cette tâche de catégorisation, nous avons opté pour des méthodes de classification non supervisée. Notre but final étant de trouver des concepts cohérents de l'application, le meilleur moyen d'y parvenir est de regrouper les mots en fonction de leurs propriétés sémantiques. La méthode à utiliser va donc regrouper les mots du corpus en différentes classes, construisant ainsi les concepts de l'application. Pour ce faire, nous avons utilisé une méthode basée sur les réseaux bayésiens en raison de leur fondement mathématique fort et le mécanisme d'inférence puissant sous-jacent. Le réseau bayésien utilisé s'appelle AutoClass, il accepte en entrée des valeurs réelles, mais aussi des valeurs non numériques comme des mots, des caractères etc. En résultat, il fournit

des probabilités d'appartenance des éléments en entrée, aux classes trouvées. Il suppose l'existence d'une variable multinomiale cachée qui peut représenter les différentes classes auxquelles appartiennent les éléments en entrée. AutoClass est basé sur le théorème de Bayes et il est décrit en détail dans [2].

Une fois l'outil de classification choisi, il nous reste à chercher une représentation adéquate des mots. En effet, un mot peut avoir plusieurs caractéristiques possibles, mais rares sont celles qui peuvent lui donner une représentation sémantique complète. Dans notre travail, nous avons décidé d'utiliser deux types d'information : le contexte des mots et la similarité du mot à représenter avec tous les autres mots du lexique. Afin d'exprimer cette similarité, nous avons utilisé la mesure de l'information mutuelle moyenne qui permet de trouver des ressemblances contextuelles entre mots. Nous associons donc à chaque mot un vecteur à M éléments, où M est la taille du lexique. L'élément numéro j de ce vecteur représente la valeur de l'information mutuelle moyenne entre le mot numéro j du lexique et le mot à représenter. La formule de l'information mutuelle moyenne [5] entre deux mots w_a et w_b est donnée par :

$$\begin{aligned} I(w_a : w_b) = P(w_a, w_b) \log \frac{P(w_a|w_b)}{P(w_a)P(w_b)} + P(w_a, \overline{w}_b) \log \frac{P(w_a|\overline{w}_b)}{P(w_a)P(\overline{w}_b)} + \\ P(\overline{w}_a, w_b) \log \frac{P(\overline{w}_a|w_b)}{P(\overline{w}_a)P(w_b)} + P(\overline{w}_a, \overline{w}_b) \log \frac{P(\overline{w}_a|\overline{w}_b)}{P(\overline{w}_a)P(\overline{w}_b)} \end{aligned} \quad (1)$$

Où $P(w_a, w_b)$ est la probabilité de trouver les deux mots w_a et w_b dans la même phrase, $P(w_a \mid w_b)$ est la probabilité de trouver le mot w_a sachant qu'on a déjà rencontré le mot w_b, $P(w_a)$ est la probabilité de trouver le mot w_a et $P(\overline{w}_a)$ est la probabilité de ne pas avoir rencontré le mot w_a etc.

Combiner contexte et mesure d'information mutuelle consiste à représenter chaque mot par une matrice d'information mutuelle moyenne à dimension $M \times 3$. La première colonne correspond au vecteur d'information mutuelle moyenne décrit précédemment, la deuxième colonne représente l'information mutuelle moyenne entre un mot quelconque du vocabulaire et le contexte gauche du mot à représenter. Idem pour la troisième colonne mais concernant le contexte droit. La $j$ème valeur de la deuxième colonne est la moyenne pondérée des informations mutuelles moyennes entre le $j$ème mot du vocabulaire et le vecteur constituant le contexte gauche du mot W_i en question. Elle est calculée comme suit :

$$IMM_j(C_g^i) = \frac{\sum_{w_g \in \textit{ contexte gauche de } W_i} I(w_j : w_g) \times K_{wg}}{Nb\_occ} \quad (2)$$

Où $IMM_j(C_g)$ représente l'information mutuelle moyenne entre le mot w_j du lexique et le contexte gauche du mot W_i. $I(w_j : w_g)$ représente l'information mutuelle moyenne entre le mot numéro j du lexique et le mot w_g qui appartient au contexte gauche du mot W_i. K_{wg} est le nombre de fois où le mot w_g est trouvé comme contexte gauche du mot W_i et Nb_occ est le nombre total d'occurrence du mot W_i dans le corpus. Le mot W_i sera donc représenté par une matrice comme le montre la figure 2. Cette matrice exploite un maximum d'informations sur le mot à représenter, ce qui a pu aider le réseau bayésien dans sa tâche de classification et nous a permis d'obtenir de bons résultats. Nous obtenons donc une liste de 12 concepts bien cohérents avec notre application. Des exemples de ces résultats sont donnés au niveau de la table 2.

$$W_i = \begin{bmatrix} I(w_1 : w_i) & IMM_1(C^i_g) & IMM_1(C^i_d) \\ I(w_2 : w_i) & IMM_2(C^i_g) & IMM_2(C^i_d) \\ \vdots & \vdots & \vdots \\ I(w_j : w_i) & IMM_j(C^i_g) & IMM_j(C^i_d) \\ \vdots & \vdots & \vdots \\ I(w_M : w_i) & IMM_M(C^i_g) & IMM_M(C^i_d) \end{bmatrix}$$

Fig. 2. Représentation matricielle du mot W_i.

Tab. 2. Quelques exemples de concepts obtenus en utilisant la représentation matricielle.

Concept	**Groupe de mots**
Favoris	Favoris, préférés, choisi, apprécié, adoré, aimé
Similarité	Similaire, semblable, pareil, équivalent, ressemblant, synonyme, proche, identique, rapproché
Demande	Souhaite, faut, désire, désirerais, peux, pourrais, veux, voudrais, possible, aimerais, souhaiterais
Ordre	Montrer, indiquer, sélectionner, trouver, donner, afficher, présenter, prendre, passer, chercher

4 Étiquetage et post-traitement

La dernière étape consiste à fournir les commandes SQL associées aux requêtes textuelles émises en entrée. C'est au cours de cette phase que nous entamons l'étape d'interprétation des requêtes. En effet, disposant de l'ensemble des concepts qui régissent notre application, nous pouvons attribuer à chaque requête ses concepts appropriés. Pour ce faire, il nous suffit d'associer à chaque mot dans la phrase sa classe sémantique correspondante.

Ensuite, nous pouvons passer à la deuxième composante de notre modèle, le "Convertisseur de représentation", où il s'agit de convertir les concepts trouvés en commandes SQL permettant d'extraire l'information requise de notre base de données. Pour ce faire, nous avons réalisé un moteur d'inférence qui à chaque concept, fait correspondre une ou plusieurs sous-requêtes génériques. Dans une requête SQL générique, les concepts interviennent au niveau des conditions. Ainsi, par exemple, si nous trouvons le concept "Date", nous ne connaissons pas la valeur de cette date mais, nous pouvons indiquer dans la requête générée qu'il y a une condition sur la date. Ce moteur d'inférence prend en compte bien sûr les répétitions, les oublis, les demandes multiples et implicites ainsi d'autres phénomènes de la parole spontanée. Dans la phase suivante, nous instancions chaque concept, dans la requête générique obtenue, par sa valeur qui est déduite en revenant à la phrase initiale. Ainsi, nous obtenons une vraie commande SQL que nous pouvons exécuter pour extraire les pages recherchées. Dans la figure 3, nous donnons un exemple illustrant les différentes étapes suivies afin d'aboutir à une commande SQL finalisée. Les résultats obtenus sont encourageants, en effet, en terme de requêtes SQL correctes, nous obtenons un taux de 100% avec le corpus d'apprentissage et un taux de 92.5% avec un corpus de test contenant 400 phrases différentes.

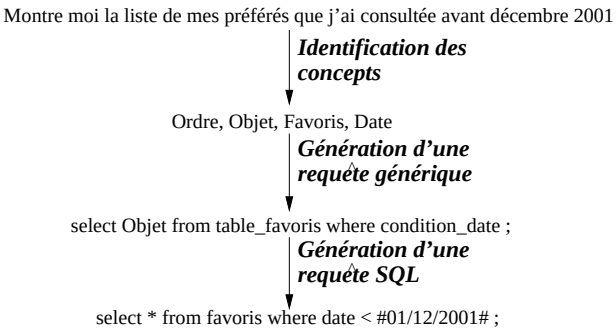

Fig. 3. Chaîne de traitement appliquée à une requête en langage naturel.

5 Conclusion

Dans cet article, nous sommes partis du principe que le problème de la compréhension automatique est un problème d'association entre deux langages différents, le langage naturel et le langage des concepts. Les concepts sont des entités sémantiques regroupant un ensemble de mots qui partagent les mêmes propriétés sémantiques et qui expriment une certaine idée. Nous avons proposé une nouvelle méthode pour l'extraction automatique des concepts, ainsi qu'une approche d'étiquetage et de génération automatique des requêtes SQL correspondantes aux demandes des utilisateurs. Les tâches d'extraction de concepts et d'étiquetage sont d'habitude réalisées manuellement. Elles constituent la phase la plus délicate et la plus coûteuse dans le processus de compréhension. La méthode proposée dans cet article a permis d'éviter ce recours à l'expertise humaine et nous a donné 92.5% de bonnes réponses sur un corpus de test de 400 requêtes exprimées en langage naturel.

Nous envisageons d'étendre le module de post-traitement de façon à ce qu'il puisse réagir face à de nouveaux mots clés non pris en compte par les concepts. Pour ce faire, il faut adapter notre modèle à la phase d'exploitation pour que nous puissions ajouter des mots aux concepts. Nous souhaitons aussi intégrer notre module de compréhension dans un système de reconnaissance automatique de la parole afin de réaliser une application interactive exploitable.

Références

1. C. Bousquet-Vernhettes and N. Vigouroux. Context use to improve the speech understanding processing. In *International Workshop on Speech and Computer, SPECOM'01*, Moscow, Octobre 2001.
2. P. Cheeseman and J. Stutz. Bayesian classification (autoclass) : Theory and results. In *Advances in Knowledge Discovery and Data Mining*. U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R. Uthurusamy, 1996.
3. H. Maynard and F. Lefèvre. Apprentissage d'un module stochastique de compréhension de la parole. In *24èmes Journées d'Étude sur la parole*, Nancy, Juin 2002.
4. R. Pieraccini, E. Levin, and E. Vidal. Learning how to understand language. In *Proceedings 4rd European Conference on Speech Communication and Technology*, Berlin, 1993.
5. R. Rosenfeld. *Adaptive Statistical Language Modeling : A Maximum Entropy Approach*. PhD thesis, School of Computer Science Carnegie Mellon University, Pittsburgh, PA 15213, Avril 1994.