Apprentissage par renforcement pour la personnalisation d'un logiciel d'enseignement des langues

Lucie Daubigney, Matthieu Geist, Olivier Pietquin

Apprentissage par renforcement pour la personnalisation d'un logiciel d'enseignement des langues

Conférence EIAH 2011 (Environnements Informatiques pour l'Apprentissage Humain)

Lucie Daubigney[*], Matthieu Geist[*], Olivier Pietquin[*,]**

** Supélec*
2, Rue Edouard Belin
57070 Metz - France
{lucie.daubigney,matthieu.geist}@supelec.fr
*** UMI 2958 (GeorgiaTech – CNRS)*
2, Rue Edouard Belin
57070 Metz - France
{olivier.pietquin}@supelec.fr

RÉSUMÉ. *Dans le cadre du projet INTERREG ALLEGRO, financé par les Fonds Européens de Développement Régional (FEDER), une interface d'enseignement des langues (français – allemand) est développée. Cette interface a pour objectif de personnaliser l'enseignement selon le profil des apprenants et de s'adapter aux performances de ceux-ci. Une phase de collecte de données est prévue et des méthodes d'apprentissage automatique de stratégie d'interaction entre le logiciel et l'apprenant seront utilisées à partir de ces données. Particulièrement, l'apprentissage par renforcement sera au cœur du système pour alterner de manière optimale les phases d'enseignement et d'évaluation. Cet article présente la modélisation du problème qui sera utilisée ainsi que des résultats préliminaires encourageants.*

MOTS-CLÉS : *personnalisation, apprentissage par renforcement.*

1. Contexte

Le projet ALLEGRO est financé par les Fonds Européens de DEveloppement Régional (FEDER) dans le cadre du programme INTERREG IV pour la Grande Région qui comprend, entre autres, la région Lorraine en France et le pays de Sarre en Allemagne. Ce projet rassemble dans son consortium le centre INRIA Nancy – Grand-Est (équipes Parole et TALARIS), le département de linguistique computationnelle de l'Université de Saarbrücken, le centre DFKI de Saarbrücken et l'équipe IMS du campus de Metz de Supélec. L'aspect novateur du projet ALLEGRO réside dans le développement et la mise à disposition de technologies avancées d'enseignement par le web (e-Learning) ayant pour finalité d'accompagner l'accroissement de la multilingualité dans ces régions frontalières. Il s'agit donc de développer une application web permettant, en particulier, de favoriser l'apprentissage de la langue française en Allemagne et de la langue allemande en France.

Dans ce contexte nous proposons d'améliorer la personnalisation de l'interface en termes d'adaptation au profil et aux performances des apprenants par le biais de méthodes d'apprentissage automatique (ou *machine learning* en anglais) et spécifiquement d'apprentissage par renforcement comme nous allons le décrire plus en détails par la suite. L'apprentissage automatique se base essentiellement sur des données réelles et la plateforme web visée nous permettra d'acquérir ces données par le biais du traçage (anonymisé) des interactions entre les apprenants et le système. Une première version de l'interface, laissant à l'apprenant une grande liberté d'interaction, permettra de recueillir un premier jeu de données assez rapidement.

2. Position du problème

Dans le cadre de la relation entre enseignant et apprenant, il a été depuis longtemps démontré que l'apprentissage se fait de manière plus efficace lorsqu'un enseignant est mis à disposition de chaque élève [BLOOM 68]. Il est naturel de penser que cette situation idéale pourrait être atteinte grâce à des environnements d'enseignement informatiques, installés sur des ordinateurs personnels ou disponibles sur le web. Néanmoins, la situation n'est idéale que si l'enseignant adapte sa méthode à l'apprenant, à ses attentes et à ses capacités. Or, les environnements d'enseignement informatiques sont souvent conçus de manière à s'adresser à un public moyen et leur comportement est pratiquement identique, quel que soit l'apprenant. Ainsi, nous nous trouvons dans une situation non seulement différente du cas idéal mais souvent bien plus mauvaise puisque le système, bien que physiquement dans une relation individuelle avec l'apprenant, est souvent conçu pour un nombre très important d'élèves. Ce qui est bon en moyenne pouvant être arbitrairement mauvais en tout point, le caractère individuel de ces interfaces n'est donc pas forcément gage de performances. C'est particulièrement le cas en ce qui concerne l'apprentissage des langues où les défauts (en termes de confusion lexicale ou de prononciation) peuvent être très différents d'un apprenant à l'autre.

Il est donc souhaitable de procurer aux interfaces des facultés d'adaptation aux attentes, aux besoins et aux capacités de chacun. Pour ce faire, il faut leur permettre de réagir en situation, grâce aux interactions avec l'apprenant. Dans notre cas, nous supposerons que la liberté de personnalisation de l'interface (que nous appellerons aussi « système ») se situe

dans l'ordonnancement des phases d'enseignement et d'évaluation. Une phase d'enseignement aura pour but de faire évoluer les connaissances de l'apprenant tandis qu'une phase d'évaluation aura pour but de quantifier cette connaissance. Ainsi, dans un contexte donné, défini d'après le passé de l'interaction avec l'apprenant, le système devra *prendre une décision* entre ces deux types de choix. L'adaptation se situe donc au niveau de la *séquence* de ces décisions qui doit être différente pour chacun suivant l'évolution de l'interaction entre le système et l'apprenant.

Le problème d'adaptation du comportement de l'interface à l'apprenant peut donc se voir comme un problème d'optimisation de prise de décisions séquentielles. Nous avons choisi de répondre à ce problème par des méthodes d'apprentissage automatique (ou apprentissage machine) à partir de données recueillies grâce à une interface donnant à l'apprenant le choix de la séquence. Dans le monde de l'apprentissage machine, l'optimisation de séquences de décisions se fait dans le cadre des processus décisionnels de Markov (PDM) et de l'apprentissage par renforcement [SUTTON & BARTO 98].

Dans la suite de cet article, nous commençons par présenter le paradigme général de l'apprentissage par renforcement. Ensuite, nous décrivons comment le problème qui nous occupe peut se placer dans le contexte des processus décisionnels de Markov. Enfin, nous présentons une première expérimentation démontrant l'intérêt de la méthode.

3. Processus décisionnels de Markov et apprentissage par renforcement

Dans le paradigme de l'apprentissage par renforcement, nous considérons le problème du contrôle d'un système dynamique et stochastique. Les configurations de ce système sont supposées appartenir à un ensemble d'*états S*, et les décisions que le contrôleur peut prendre pour modifier ces configurations appartiennent à un ensemble d'*actions A*. Lorsque le contrôleur applique une action *a* sur le système alors qu'il est dans l'état *s*, celui-ci transite vers un nouvel état *s'* selon une probabilité de *transition* p(*s*,*a*,*s'*) = $T_{sas'}$ qui ne dépend que de l'état *s* et de l'action *a* mais pas du passé des séquences d'états-actions (c'est la propriété de Markov). Après chaque transition, le système stochastique génère une récompense *r* selon une fonction intrinsèque *R*. Formellement, le *tuple* {*S,A,T,R,*γ} définit un *processus décisionnel de Markov* (PDM) où γ est un facteur d'actualisation dont l'utilité est expliquée ci-après. Le but de l'apprentissage par renforcement est de trouver une *politique* de contrôle (notée π), c'est-à-dire l'association d'une action à chaque état, qui maximise la récompense moyenne sur le long terme définie par :

$$J = E^{\pi}\left[\sum_{i=0}^{\infty} \gamma^{i} r_{i}\right]$$

Il existe de nombreux algorithmes pour apprendre la politique de contrôle optimale, selon que l'on connaisse ou pas les probabilités de transition (*T*) (*programmation dynamique* ou *apprentissage par renforcement*) [SUTTON & BARTO 98] et selon que l'on interagisse directement avec le système [WATKINS 89] ou que l'on apprenne à partir de jeux de données fixes [LAGOUDAKIS & PARR 03] obtenus sur le système avant apprentissage (apprentissage *au fil de l'eau* ou *hors ligne*).

4. Système d'enseignement et PDM

Comme indiqué plus tôt, la personnalisation d'un système d'enseignement informatique peut être considérée comme l'adaptation d'une séquence de décisions résultant en l'alternance de phases d'enseignement et d'évaluation. L'utilisation d'apprentissage par renforcement pour ce faire a déjà été proposé dans [BECK *et al* 00] et [IGLESIAS *et al* 09]. Nous nous distinguons de ces travaux en proposant une nouvelle modélisation et en utilisant un algorithme d'apprentissage travaillant sur un jeu de données fixes (et donc ne nécessitant pas d'interagir avec l'apprenant pendant l'optimisation du système). Pour trouver la séquence optimale de décisions par apprentissage par renforcement, il faut transposer le problème de personnalisation dans le cadre des PDM, et ainsi définir les états, les actions et les récompenses. En ce qui concerne les actions, la transposition est relativement simple puisqu'il y a deux types d'actions : démarrer une phase d'enseignement ou une phase d'évaluation. Définir un état est plus complexe. Nous cherchons à représenter le *contexte* de l'interaction, soit l'information nécessaire et suffisante pour prendre une décision (afin d'être compatible avec la propriété de Markov). Ici, nous définirons l'état par un vecteur à deux dimensions :

- la première dimension représente le taux de bonnes réponses fournies par l'apprenant jusqu'ici (une valeur entre 0 et 1) ;
- la seconde dimension représente le nombre de phases d'enseignement reçues par l'apprenant jusqu'ici.

Enfin, la récompense est donnée par le taux de bonnes réponses de l'apprenant lorsqu'il est interrogé. Il s'agit donc de maximiser ce taux, selon les capacités de l'apprenant, et non pas d'arriver à un objectif fixé (par exemple 90% de bonnes réponses dans le travail décrit dans [IGLESIAS *et al* 09]).

5. Expérience

N'ayant pas encore de donnée réelle à notre disposition, nous les avons simulées avec un modèle d'apprenant inspiré de [CORBETT & ANDERSON 94]. Néanmoins ce modèle est inconnu du système d'apprentissage par renforcement. Il est seulement utilisé pour générer des données compatibles avec notre représentation d'état. La politique utilisée pour obtenir ces données est une politique totalement aléatoire (choisissant à tout instant entre enseignement et évaluation avec une probabilité égale). Nous avons ensuite appliqué l'algorithme Least Square Policy Iteration (LSPI) [LAGOUDAKIS 03] sur ces données et avons appris une stratégie optimale d'interaction pour notre apprenant simulé. Les résultats de cet apprentissage sont donnés sur la Figure 1. Sur cette figure, le cumul de récompense moyen est tracé en fonction du nombre d'interaction utilisé pour l'apprentissage. Pour quantifier la reproductibilité des résultats, l'apprentissage est réalisé 100 fois sur 100 jeux de données différents et la politique apprise est testée 1000 fois avec l'apprenant simulé. Ainsi, l'intervalle de confiance à 95% est aussi tracé, montrant que le résultat de l'apprentissage est peu sensible au caractère aléatoire des données. Ceci est rassurant car en situation réelle, il ne nous sera pas possible de générer autant de données que l'on souhaite. La courbe obtenue par apprentissage (en bleu) est comparée à la récompense obtenue par la politique aléatoire (en rouge) ou par une politique simple alternant systématiquement phase

d'apprentissage et phase d'évaluation (en vert). Il est donc clair qu'après un peu plus de 500 interactions, la politique apprise est meilleure qu'une politique non-adaptée. On peut aussi voir que l'intervalle de confiance à 95% diminue avec le nombre d'interactions bien qu'il ne change plus beaucoup après 1500 interactions.

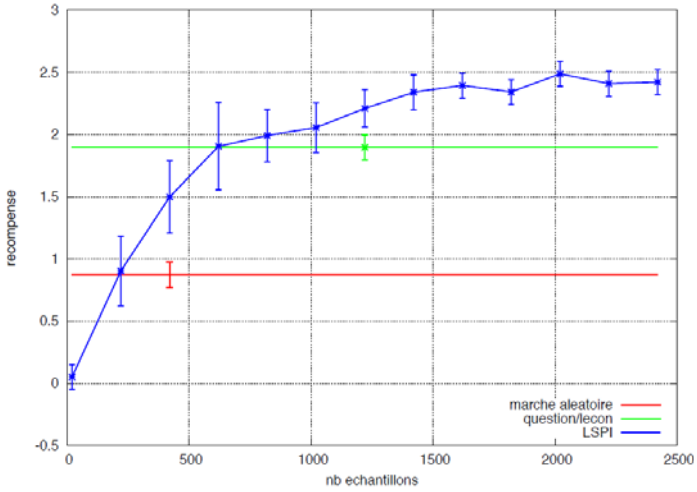

Figure 1: Résultats de LSPI

6. Conclusion

Dans cet article, nous avons proposé une méthode de personnalisation d'une interface d'enseignement se basant sur l'apprentissage par renforcement. Une première expérience donne de bonnes raisons de penser que cette approche est pertinente. Elle sera utilisée en pratique dans le cadre d'un projet multi-partenaire ayant pour but la mise en ligne d'une interface d'enseignement des langues et pourra bénéficier des données collectées dans ce contexte.

7. Bibliographie

[BECK *et al* 00] J.E. Beck, B.P. Woolf & C.R. Beal. "ADVISOR : A machine learning architecture for intelligent tutor construction." In *Proceedings of the National Conference on Articial Intelligence*, pages 552{557. Menlo Park, CA; Cambridge, MA; London ; AAAI Press ; MIT Press ; 1999, 2000.

[BLOOM 68] B.S. Bloom. "Learning for mastery". *Evaluation comment*, vol. 1, no. 2, pages 1{5, 1968.

[CORBETT & ANDERSON 94] A.T. Corbett & J.R. Anderson. Knowledge tracing : Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, vol. 4, no. 4, pages 253{278, 1994.

[IGLESIAS *et al* 09] A. Iglesias, P. Martinez, R. Aler & F. Fernandez. Learning teaching strategies in an adaptive and intelligent educationalsystem through reinforcement learning. *Applied Intelligence*, vol. 31, no. 1, pages 89{106, 2009.

[LAGOUDAKIS 03] M.G. Lagoudakis & R. Parr. "Least-squares policy iteration". *The Journal of Machine Learning Research*, vol. 4, pages 1107:1149, 2003.

[SUTTON & BARTO 98] R.S. Sutton & A.G. Barto. *Reinforcement learning : An introduction*. The MIT press, 1998.

[WATKINS 89] C.J.C.H.Watkins. *Learning from delayed rewards*. PhD thesis, King's college, Cambridge University, 1989.