Sujet de stage - M2R IF
Année 2017-2018

Extraction d'automates pondérés d'un réseau de neurones récurrent

Lieu : Centre de Mathématique et d'Informatique, Technopole de Chateau-Gombert, Marseille
Encadrants : Rémi Eyraud (`Remi.Eyraud@lif.univ-mrs.fr`)
Stéphane Ayache (`Stephane.Ayache@lif.univ-mrs.fr`)
Poursuite en thèse : possible

Contexte

Réseau de neurones récurent

Le succès récent des applications utilisant les réseaux de neurones a généré un renouveau de l'intelligence artificielle en pointant les projecteurs sur l'apprentissage machine. Des domaines aussi variés que la médecine, l'astrophysique, la bio-informatique, le traitement du signal, et même le droit sont en train d'être révolutionnés par l'arrivée des réseaux de neurones dits profonds et des masses de données auxquelles ils peuvent s'attaquer.
Cet essor rapide souffre toutefois de 2 défauts principaux qui pourraient poser problème à terme :

— le manque de résultats théoriques démontrant la validité de l'approche ;
— l'absence d'interprétabilité des modèles appris.

Ce dernier point, primordiale en apprentissage, est à nuancer : en computer vison par exemple, il est possible d'analyser a posteriori les sorties des couches de convolution et d'y observer des filtres de différent niveau.

Lorsque les données sont séquentielles et/ou temporelles, par exemple dans le traitement de la langue, les réseaux utilisés sont dits *récurrents* : la mise à jour des poids tient compte des éléments vus précédemment. L'interprétation de tels réseaux est difficile et leur emploi relève le plus souvent de l'utilisation d'une boite noire [4].

Apprentissage d'automates pondérés

En parallèle au succès des réseaux de neurones, des résultats tant théoriques que pratiques ont été obtenus dernièrement pour l'apprentissage de modèles au coeur de l'informatique théorique : les automates à états finis. En particulier, une famille d'algorithmes efficaces a été étudiée, sous le nom d'apprentissage spectral, pour les automates pondérés, c'est-à-dire des automate éventuellement non-déterministes dont les arcs portent des poids à valeurs réelles (voir la Figure 1).

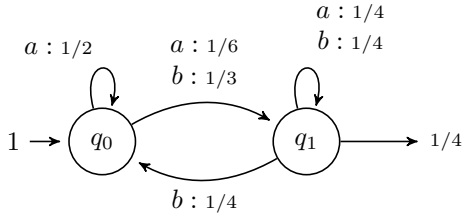

FIGURE 1 – Un exemple d'automate pondéré.

Ces algorithmes d'apprentissage reposent sur l'utilisation d'une matrice particulière, appelée matrice de Hankel, qui représente l'information présente dans les données : dans sa variante classique, les lignes sont des préfixes et les colonnes des suffixes. Une case contient alors la valeur de la fonction représentée par l'automate sur le mot formé du préfixe et du suffixe correspondant. A partir de

cette matrice, l'automate correspondant peut être retrouvé à l'aide d'une décomposition en valeurs singulières [2].

En plus de garanties théoriques, ces modèles sont facilement interprétables : leur structure nous renseigne sur celle du langage (stochastique) représenté et ils sont facilement représentables graphiquement.

Objectifs du stage

Ce stage mêlera aspects théoriques et expérimentaux, le candidat devra étudier, implémenter et analyser divers travaux récents en apprentissage automatique et en informatique fondamentale.

L'objectif du travail proposé est d'adapter l'algorithme d'apprentissage spectral d'automates pondérés pour interpréter un réseau de neurones récurent (RNN) appris par ailleurs. L'idée est d'utiliser le réseau pour remplir la matrice de Hankel, puis d'utiliser la matrice pour obtenir un automate pondéré.

Ce stage se déroulera donc en cinq étapes :

1. Étude bibliographique : apprentissage spectral d'automates pondérés, réseaux de neurones récurrents.
2. Prise en main des outils : Keras pour le deep learning et Scikit-SpLearn [1] pour l'apprentissage spectral.
3. Définition, analyse, et implémentation d'un algorithme spectral d'extraction d'un automate pondéré d'un réseau de neurones récurrent appris préalablement.
4. Étude du comportement de l'algorithme sur différent jeux de données artificielles (PAutomaC, SPiCe) et réelles (traitement de la langue, bio-informatique, ...).
5. A la lumière du succès des étapes précédentes, différentes continuations peuvent être envisagées : étude théorique du processus, mise en place d'une approche type *Generative Adversarial Network (GAN)* [3] pour profiter des interactions entre le RNN et l'automate, analyse fine du comportement des RNN par la proposition d'un protocole reposant sur l'interprétabilité de l'automate, ...

Le langage de programmation utilisé sera le Python.

Références

[1] Denis Arrivault, Dominique Benielli, François Denis, and Remi Eyraud. Sp2Learn : A toolbox for the spectral learning of weighted automata. In *Proc. of The 13th International Conference on Grammatical Inference*, volume 57 of *Proceedings of Machine Learning Research*, pages 105–119, 2016.

[2] Borja Balle, Xavier Carreras, Franco M. Luque, and Ariadna Quattoni. Spectral learning of weighted automata. *Machine Learning*, 96(1) :33–63, Jul 2014.

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680. 2014.

[4] Christian W. Omlin and C. Lee Giles. Knowledge-based neurocomputing. chapter Symbolic Knowledge Representation in Recurrent Neural Networks : Insights from Theoretical Models of Computation, pages 63–116. 2000.