Génération Incrémentale de Textes par Apprentissage par Renforcement Profond: Application à la génération de programmes d'exercices

1 Contexte

La thèse en cours avec Kara Technology consiste à proposer à des patients un programme d'exercices leur permettant une rééducation ou un maintien en utilisant de nouveaux dispositifs électroniques (tablettes tactiles par exemple). Il s'agit de personnaliser ainsi les programmes en fonction des données propres du patient ainsi que des exercices et protocoles disponibles. Un des objectifs est de s'adapter au mieux au ressenti des patients et aux guidages des thérapeutes.

Dans ce contexte, cette thèse tandem plus fondamentale vise à généraliser ces concepts dans un contexte apprenant/tuteur afin de générer automatiquement des programmes d'exercices visant à répondre à des attentes d'une personne cherchant à progresser sur un sujet sous la supervision d'un encadrant. Il s'agira de s'appuyer sur les avancées récentes impressionnantes en apprentissage profond pour la génération automatique de texte, notamment les architectures Transformer [12, 2, 3], pour générer des programmes adaptés sous forme textuelle. Néanmoins, les architectures actuelles n'offrent que de faibles capacités de personnalisation des textes produits, car apprises principalement de manière supervisée pour faire correspondre des entrées à des textes cibles. Dans le contexte applicatif visé, comme dans bien d'autres cadres plus généraux impliquant de la génération de langue personnalisée, il est nécessaire d'imaginer de nouveaux processus d'apprentissage, permettant un meilleur contrôle et une meilleure explicabilité des textes générés. Nous proposons de considérer la mise en oeuvre de processus d'évolution textuelle incrémentaux, où le texte cible est produit par transformations successives du texte d'origine, plutôt que généré en une étape boîte noire unique difficile à interpréter. La thèse abordera cette piste de recherche via une approche innovante de supervision basée sur l'apprentissage par renforcement profond, d'abord sur des tâches classiques du domaine (e.g., résumé automatique), avant de mettre en application les avancées réalisées dans le domaine médical pour la production de programmes d'exercice personnalisés.

2 Objectifs et verrous

L'objectif de la thèse est de proposer une nouvelle approche pour la génération automatique de textes, basée sur des processus itératifs, que l'on appliquera à la génération de programmes d'exercices. Viser un texte cible (i.e., un programme personnalisé dans notre contexte applicatif) en une seule étape de manière supervisée aboutit souvent à une génération assez difficilement contrôlable sur le niveau de détail attendu, et peu personnalisable. Dans le processus de génération itératif envisagé, chaque étape de transformation ciblera la production d'une légère évolution du texte d'entrée (e.g., texte à résumer dans le cadre de tâches de résumé automatique, programme d'entrée à personnaliser en fonction des retours du patient dans notre contexte applicatif). Ce genre de génération par transformations successives a déjà été employé avec succès dans le domaine continu pour la génération d'images [1]. Dans ce cadre, il s'agit de générer progressivement les images pour qu'elles se rapprochent de la cible visée (par un processus dit de diffusion), via une rétro-propagation du gradient de l'image finale vers les couches de transformation (voir par exemple le modèle Dall-e 2 [7]). Pour le cas d'un objet complexe comme un texte, c'est plus compliqué car dans le domaine discret ce genre de retro-propagation est impossible, chaque étape impliquant l'échantillonnage d'éléments du dictionnaire (i.e., des tokens textuels). Bien sûr on ne dispose pas de tous les textes de référence intermédiaires qui serviraient à un apprentissage entièrement supervisé. Il s'agira alors de définir des stratégies d'apprentissage adaptées permettant de contourner ces limitations. Les verrous scientifiques à lever sont multiples :

- Contrôle de la sortie désirée d'une étape de génération à partir de références finales uniquement (i.e. les objectifs cibles d'un corpus d'apprentissage) ;
- Apprentissage sans accumulation de l'erreur de génération au cours des différentes étapes ;
- Généralisation à tout type de formulation du domaine visé (protocoles, exercices, objectifs thérapeutiques...)
- Vérification factuelle et interprétabilité du processus itératif de génération pour une meilleure analyse par les praticiens.

Pour répondre à ces différents défis, nous envisageons à terme de nous placer dans un contexte d'apprentissage par renforcement, permettant de mettre en cohérence les distributions de conditionnement des modèles en apprentissage et en inférence (et ainsi éviter des effets bien connus de biais d'exposition avec dérive temporelle des modèles) et où il sera possible de considérer la maximisation multi-critères de fonctions objectifs complexes, pas nécessairement dérivables (en s'appuyant par exemple sur des mesures basées sur des systèmes de question-answering, comme

employé avec succès dans [10]).Cependant, il paraît complexe de s'attaquer à un apprentissage end-to-end de ce genre de système de front, une étape préliminaire consistera alors en la définition de briques de synthèse élémentaires, dont la sortie nous approche de la référence long-terme en terme de longueur par rapport à son entrée, et où les éléments factuels principaux ne sont pas altérés. Dans un esprit d'apprentissage progressif (i.e. curriculum learning) avec densification du reward, il s'agira ensuite de terminer l'apprentissage du module de synthèse par application successive des transformations correspondantes.

Différents types d'architectures pourront être envisagées pour générer le texte et conditionner le processus. L'utilisation de modèles LLM pre-entraînés open-source est une option, via notamment l'emploi des outils issus de la librairie HugginFace [1] (e.g. Flan-T5, Llama2, etc.), que l'on pourra chercher à adapter à notre méthodologie de résumé et à notre domaine. Pour éviter de trop diverger des distributions initiales du modèle de langue considéré (et ainsi, garantir des sorties grammaticalement correctes, des cohérences sémantiques, de bonnes capacités de référencement croisé, etc.), l'apprentissage du modèle devra considérer différentes contraintes de stabilité. Dans l'esprit PPO [9] de RLHF [8], on pourra tout d'abord chercher à garantir que les distributions de tokens restent dans une zone de confiance autour du modèle (e.g., selon une divergence de Kullback-Leibler). Plutôt que de modifier les poids de manière classique (avec donc le risque de trop déstabiliser le réseau), un complément à cette regularisation qui est largement considérée actuellement dans la communauté NLG est l'utilisation de perturbations de faible rang des poids des réseaux (LORA [4]). L'ajout de prompts continus appris est aussi une possibilité à considérer pour adapter les LLMs (*prompt tuning*, appliqué au domaine medical dans [6]), qui a l'avantage de ne pas requérir d'optimisation de l'ensemble des paramètres du LLM, uniquement ceux d'un embedding de token special. Néanmoins, toutes ces techniques nécessitent d'avoir à disposition un LLM open-source efficace. Étant donné le manque de partage des poids des modèles les plus performants par les sociétés qui les détiennent, une large partie de la communauté s'est tournée vers de l'injection en contexte, c'est à dire l'inclusion des descriptions des taĉhes à effectuer selon des instructions textuelles en entrée des modèles, profitant des très bonnes capacités des plus grands modèles de langue à analyser l'entrée fournie. Ainsi, il est possible de travailler sans ne jamais modifier aucun poids des réseaux, uniquement par application récursive de LLMs [5] (ce sera d'ailleurs l'option privilégiée pour démarrer le travail de thèse, car bien plus simple à mettre en oeuvre). Enfin, il sera possible d'accroître les capacités de ce genre de méthodes en apprenant par renforcement à conditionner les LLMs utilisés successivement (e.g., via l'utilisation d'un autre LLM de génération de prompt [11], éventuellement plus petit, que l'on adaptera pour maximiser les objectifs de la taĉhe considérée).

Références

[1] Hanting Chen, Yunhe Wang, Tianyu Guo, Chang Xu, Yiping Deng, Zhenhua Liu, Siwei Ma, Chunjing Xu, Chao Xu, and Wen Gao. Pre-trained image processing transformer, 2020.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT : pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019.

[3] Nadir Durrani, Hassan Sajjad, and Fahim Dalvi. How transfer learning impacts linguistic knowledge in deep NLP models ? In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, *Findings of the Association for Computational Linguistics : ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 4947–4957. Association for Computational Linguistics, 2021.

[4] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora : Low-rank adaptation of large language models. *arXiv preprint arXiv :2106.09685*, 2021.

[5] Yoav Levine, Itay Dalmedigos, Ori Ram, Yoel Zeldes, Daniel Jannai, Dor Muhlgay, Yoni Osin, Opher Lieber, Barak Lenz, Shai Shalev-Shwartz, et al. Standing on the shoulders of giant frozen language models. *arXiv preprint arXiv :2204.10019*, 2022.

[6] Cheng Peng, Xi Yang, Kaleb E Smith, Zehao Yu, Aokun Chen, Jiang Bian, and Yonghui Wu. Model tuning or prompt tuning ? a study of large language models for clinical concept and relation extraction. *arXiv preprint arXiv :2310.06239*, 2023.

[7] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022.

[8] Michael Santacroce, Yadong Lu, Han Yu, Yuanzhi Li, and Yelong Shen. Efficient rlhf : Reducing the memory usage of ppo. *arXiv preprint arXiv :2309.00754*, 2023.

[9] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv :1707.06347*, 2017.

[10] Thomas Scialom, Sylvain Lamprier, Benjamin Piwowarski, and Jacopo Staiano. Answers unite ! unsupervised metrics for reinforced summarization models. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3244–3254. Association for Computational Linguistics, 2019.

[11] Hao Sun. Reinforcement learning in the era of llms : What is essential ? what is needed ? an rl perspective on rlhf, prompting, and beyond. *arXiv preprint arXiv :2310.06147*, 2023.

[12] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30 : Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

1. `https://huggingface.co/`