LIS@DEFT'23 : les LLMs peuvent-ils répondre à des QCM ? (a) oui ; (b) non ; (c) je ne sais pas.

Benoit Favre

TALEP, LIS UMR 7020, CNRS/Aix-Marseille Université

5 juin 2023

Plan

Tâche DEFT 2023

- Tâche : Répondre automatiquement à des QCM extraits d'examens de pharmacie.
  - Approche classique : BERT pour chaque couple question-réponse, puis classifieur binaire (Labrak et al. 2022)
- Problématique centrale : où trouver les connaissances du domaine ?
  - Approche classique : Trouver des documents correspondant à la question et les introduire en entrée des modèles
  - Notre approche : Grands modèles de langage (LLM)
    1. Performances des modèles "instruits" sur la tâche de QCM ?
    2. Affinage des modèles à bas coût bénéfique pour traiter la tâche ?
    3. Lien entre taille des LLM et performances attendues ?

Grands modèles de langage

- Modèles génératifs : tâche de prédiction du prochain token étant donné un contexte
- Architecture transformers : couches d'auto-attention multi-tête + encodage de position
- Entraînement de nombreux paramètres sur de grandes quantités de données
  - Généralisation multi-tâche et émergence de capacités zéro-shot
  - Émergence de capacités nouvelles avec la taille des modèles
  - Lien entre taille du modèle et des données d'apprentissage (scaling laws)
- Affinage sur des instructions et/ou des préférences humaines
  - Prompts/amorces : description de la tâche, des entrées, puis génération des sorties
- $\rightarrow$ Large Langage Models (LLM)

Plan

Amorces

- Exploration de l'espace des prompts
  - Contexte ressemblant aux données d'apprentissage "Corrigé des épreuves de pharma..."
  - Mise en scène d'un dialogue "Claire est chercheuse en Pharmacie, Pierre lui pose des questions auxquelles elle répond précisément. Pierre : ... Claire : ..."
  - Description en anglais de la tâche "Please answer this MCQ..." (BLOOMz)
- Amorce finale
  - Instruction et contraintes sur la réponse
  - Question et réponses possibles
  - Sollicitation de réponse avec contraînte sur la forme (parenthèse ouvrante)

> Ceci est une question de QCM de l'examen de pharmacie. Réponds avec la ou les lettres correspondant à la bonne réponse.
>
> La diminution d'une unité pH correspond à une concentration en H+ :
> (a) 2 fois plus forte.
> (b) 10 fois plus faible.
> (c) 10 fois plus forte.
> (d) 100 fois plus forte.
> (e) 100 fois plus faible.
> Réponse(s) : (

Affinage

- L'affinage des gros modèles est très coûteux en mémoire (10x-100x la quantité nécessaire à l'inférence, donc multiples GPU)
- Low-rank Adaptation (LORA)
  - Somme entre paramètres d'origine (gelés, compressés) et matrices de faible rang apprenables

$$y = x\left(\tilde{\boldsymbol{W}}_{(m\times n)} + \boldsymbol{V}_{(m\times k)}\boldsymbol{U}_{(k\times n)}\right)^{\top} + \boldsymbol{b}, \quad k \ll min\{n, m\}$$

- Amorce pour l'affinage

> Ceci est une question de QCM de l'examen de pharmacie. Réponds avec la ou les lettres correspondant à la bonne réponse.
>
> Les complications d'une hépatite virale aiguë peuvent être à plus ou moins long terme :
> (a) Une lithiase vésiculaire.
> (b) Une hépatite chronique.
> (c) Un cancer du foie.
> (d) Une cirrhose.
> (e) Une pancréatite aiguë.
> Réponse(s) : (b) Une hépatite chronique ; (c) Un cancer du foie ; (d) Une cirrhose.

Plan

Cadre expérimental

- Expériences avec trois types de modèles
  1. Modèles ouverts instruits disponibles sur huggingface
  2. Modèles ouverts affinés sur DEFT
  3. Modèles commerciaux fermés
- Affinage
  - Modèle LLaMa 7, 13, 30, 65 milliards de paramètres
  - Paramètres quantifiés sur 8 bits
  - Entraînement sur DEFT-train pendant 1 époque
  - Taille de batch : 24 (microbatch 1), séquences coupées à 256 tokens
  - LORA avec $k = 4$, uniquement sur les matrices de projection `q_proj` et `v_proj` du mécanisme d'attention
    - ⋆ 2 à 12 millions de paramètres affinés selon modèle
    - ⋆ Entraînement en quelques heures sur un unique GPU A100-80GB

Résultats : modèles instruits (dev)

Modèle	EMR
bloomz-560m	0.0737
bloomz-3b	0.1442
bloomz-7b1	0.1602
bloomz-7b1-mt	0.1762
flan-t5-xxl-11b	0.1794
flan-ul2-20b	0.1570
tk-instruct-3b-def	0.1346
tk-instruct-11b-def	0.1826
oasst-sft-1-pythia-12b	0.0705
opt-iml-1.3b	0.0673
opt-iml-30b	0.1442
galactica-125m	0.0128
galactica-1.2b	0.0192
galactica-6.7b	0.0352
pmc-llama-7b	0.0224

- Perfs liées à la taille des modèles dans une famille
- Meilleur modèle zéro-shot aussi bon que BERT supervisé ($\simeq 0.18$, Labrak et al. 2022)
- Effet du type d'instructions pas très clair
- Modèles spécialisés pas forcément meilleurs

Résultats : modèles affinés (dev)

Modèle	Taille	Affinage	EMR
llama	7B	-	0.0576
llama	7B	alpaca	0.1217
llama	7B	alpaca-fr	0.1185
llama	7B	deft	0.1378
llama	13B	-	0.0769
llama	13B	alpaca	0.1474
llama	13B	deft	0.1730
llama	30B	-	0.1442
llama	30B	alpaca	0.1923
llama	30B	deft	0.2467
llama	65B	-	0.1730
llama (fp16)	65B	-	0.2179
llama	65B	deft	0.3044

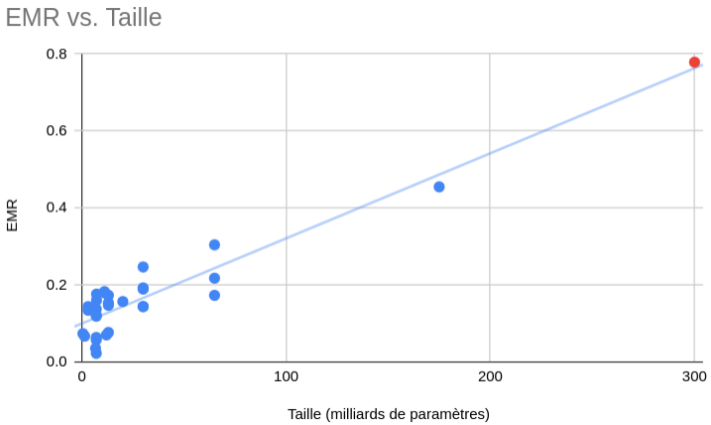
- Instructions génériques $<$ affinage DEFT
- Peu d'effet de la traduction (à confirmer)
- Perfs liées à la taille des modèles
- Effet significatif de la quantification

Résultats : modèles commerciaux

Fournisseur	Modèle	EMR
cohere	command-xlarge-beta	0.1057
ai21	j1-jumbo	0.0833
openai	code-cushman-001	0.1121
openai	code-davinci-002	0.3108
openai	text-curie-001	0.1217
openai	text-davinci-003	0.2884
openai	gpt-3.5-turbo-0301 “ChatGPT"	0.4551
openai	gpt-4-0314	0.7788

- Modèle appris sur du code source aussi bon que modèle affiné par nos soins
- Performances liées à la capacité des modèles à suivre les instructions
- Coût raisonnable mais non-nul : $\simeq 2$ USD pour traiter le test avec GPT-4
- Impossible de tirer des conclusion scientifiques sans description fiable des modèles
  - Certains modèles ne sont même plus accessibles pour répliquer les résultats

Effet de la taille, conclusion



Résultats dans la campagne (test)

		Tâche principale		Tâche annexe	
Nom du système	**Repro.**	**Hamming**	**EMR**	**F1-Score**	**Accuracy**
LIS/llama-65b-lora	✓	52.94	33.76	42.42	68.65
LIS/llama-30b-lora	✓	47.43	27.81	35.26	65.92
LIS/llama-13b-lora	✓	35.93	17.85	34.52	65.11
LIS/gpt-3.5-turbo-0301_prompt0		64.75	46.95	47.51	68.17
LIS/gpt-4-0314_prompt0		85.17	72.83	71.57	79.58

- Résultats stables par rapport au dev
- Tâche annexe dérivée directement de la tâche principale
  - Différence inexpliquée entre ChatGPT et GPT-4

Les modèles ont-ils mémorisé le test ?

- Memorization Effects Levenshtein Detector, MELD [Nori et al, 2023, arXiv :2303.13375v2]
  - Générer la 2e moitier des questions avec une température de 0
  - Calculer un score de mémorisation à partir de l'alignement de Levenshtein entre reference et texte généré

> Parmi les termes suivants quels sont ceux pouvant caractériser la précision d'une méthode :
> (a) La fidélité.
> (b) L'exactitude.
> (c) La reproductibilité.
> REF : (d) La sensibilité. (e) La répétabilité.
> GEN : (d) La sensibilité. (e) La spécificité.
> DISTANCE : 0.15

Model	$\% < 0.05$	**Dist. moyenne**
ChatGPT	2.25	0.4266
GPT-4	1.92	0.4756

Plan

Conclusion

- Les LLM sont une option viable pour la réponse à des QCM dans le domaine médical
  - Performances liées à la taille des modèles
  - Modèles non instruits $<$ instructions génériques $<$ affinage LORA
  - Code source disponible (un peu brouillon) : `https://gitlab.lis-lab.fr/benoit.favre/deft2023-llm`
- Mais cadre expérimental difficile à construire
  - Pas de garantie que les données n'ont pas déjà été vues à l'entraînement
  - Ressources nécessaires pour apprentissage et inférence
- Reste à explorer
  - Few-shots pour mieux imposer la tâche aux modèles non affinés
  - Langue française dans les LLM
  - Lien entre jeux d'instructions et performances
  - Introduction explicite de connaissances externes