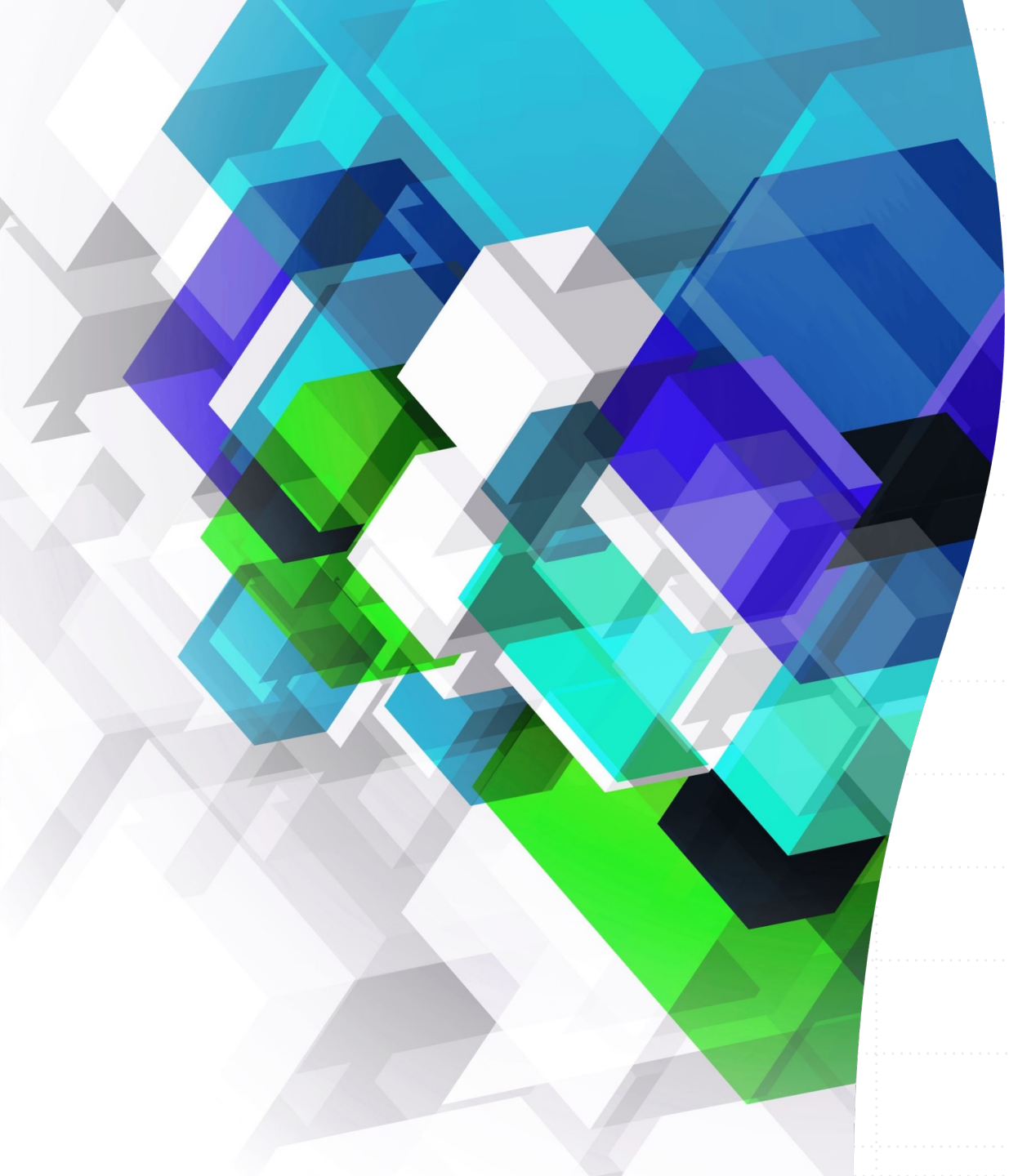

Aprendizagem por Reforço com Modelos de Linguagem de Grande Porte (LLMs) - REMEMBERER

Igor Nascimento – ivln@cin.ufpe.br

Monique Monteiro – mlblm@cin.ufpe.br

IN1087 – Tomada de Decisão sob Incerteza

Centro de Informática - UFPE

Narrativa

Um framework de agente tendo um *Large Language Model* (LLM) como o agente que aprende a tomar decisões melhores com base na memória de experiências de episódios passados do processo de aprendizagem por reforço. O objetivo é atualizar a memória de experiências com os dados de observações, ações e recompensas para que o agente LLM possa tomar melhores decisões futuras.

Elementos centrais

Métricas

O sistema deve retornar a máxima expectância das estimativas dos valores baseado na última observação, ação, recompensa e a nova observação. Nos exemplos descritos no artigo, métricas mais concretas incluem a taxa de sucesso em dois *benchmarks* recentes: navegação por um site de comércio eletrônico (WebShop), onde o objetivo é comprar um determinado produto especificado como entrada, e navegação por um site de perguntas e respostas (WikiHow), onde o objetivo é chegar à página que contém a resposta correta para uma dada pergunta.

Decisões

Precisa-se decidir qual é a melhor ação a ser tomada com base nas informações da memória de experiências do sistema. Seguindo os exemplos fornecidos nos benchmarks realizados, as decisões referem-se à descrição textual da opção de navegação a ser tomada pelo usuário (no caso, o agente), como selecionar um link, realizar uma busca, retornar à página inicial, confirmar uma compra etc.

Fontes de incerteza

Uma das fontes de incerteza é o resultado das observações do ambiente nas futuras tomadas de decisão. Assim, a cada etapa do seu processo de decisão, o LLM obtém uma observação do ambiente, no caso concreto, a descrição textual dos elementos da tela no qual ela está navegando, em ambos os exemplos.

Modelo matemático

Variáveis de estado

A variável de estado pode ser modelada como a memória de experiências resultantes das interações com o ambiente até o momento t. Assim, podemos representar cada estado St como sendo:

$$S_t = (O_t, A_t, Q_t)$$

A variável de estado inicial é dada por:

$$S_0 = (O_0, A_0, Q_0)$$

Variáveis de decisão

- A variável de decisão xt é a ação que o LLM vai executar dependendo da tarefa a ser realizada.
- refere-se à sequência de tokens predita pelo modelo.
- a técnica de amostragem a partir dessa probabilidade é variável e muitas vezes configurável.
- pode ser adotada a heurística de sempre prever o token de maior probabilidade (greedy-decoding), mas também podem ser adotadas variações como beam search, top-k, top-p, temperatura, etc.

Informações exógenas

A observação é resultante da ação executada no passo t1, e conforme citado anteriormente, tal ação não é determinística, podemos assumir que a observação também não é determinística, vindo a constituir portanto um elemento de incerteza. Sendo assim, vamos modificar ligeiramente a notação para adaptá-la à do framework de Warren Powell, adotando a terminologia Wt+1:

$$W_{t+1} = o_{t+1}$$

Função de transição

Durante a interação com o ambiente, novas experiências são adicionadas à memória de experiências. Logo, dada a tupla (ot, at, rt, ot+1, contendo a última observação, a ação executada, a recompensa associada e a nova observação, temos as seguintes atualizações:

$$O_{t+1} = O_t \cup \{o_t\}$$

$$A_{t+1} = A_t \cup \{a_t\}$$

$$Q_{t+1} = Q_t \cup \{Q'(g, o_t, a_t)\}$$

$$Q'(g, o_t, a_t) = r_t + \gamma \max_a Q(g, o_{t+1}, a)$$

Função de transição

caso não exista um registro associado a (g, ot, at) na memória. Caso contrário, o valor Q estimado será atualizado por meio do algoritmo Q-Learning:

$$Q'(g, o_t, a_t) \leftarrow (1 - \alpha)Q(g, o_t, a_t) + \alpha Q'(g, o_t, a_t)$$

sendo alpha = 1/N, e N denota o número de vezes em que esse valor é atualizado.

Função objetivo

WebShop:

$$\max_{\theta} F^{\pi}(\theta) = \mathbb{E}_{S_0}\mathbb{E}_{W^1,\ldots,W^N|S_0}\mathbb{E}_{\hat{W}|S_0} Sim(y, \hat{o}^N)$$

WikiHow:

$$\max_{\theta} F^{\pi}(\theta) = \mathbb{E}_{S_0}\mathbb{E}_{W^1,\ldots,W^T|S_0}\left\{\sum_{t=0}^{T} Sim(y_{t+1}, s_{t+1})\right\}$$

Modelagem da Incerteza

- Segundo Powell:
  - Observações de campo → método *data driven* (observação direta da saída)
  - No caso concreto: simuladores WebShop e WikiHow (“*environments*”)

- Fontes de incerteza:
  - Decisões tomadas pelo LLM (sequências de tokens previstas de forma auto-regressiva)
  - Modelo de Markov: prob. de cada token dada a prob. dos tokens anteriores
  - Cada saída (decisão) do LLM → atualiza memória de experimentos

Projetando políticas

- Em cada passo <estado, observação>, o sistema recupera:
  - Conjunto de experiências:
    - N tuplas (observação, ação, recompensa)
    - Subconjunto com M < N tuplas repassado para o LLM
      - Engenharia de prompts (fow-shot learning)
      - Exemplos de ações recomendadas e não recomendadas
- Política:
  - Atualização dos valores Q para cada tupla (tarefa, observação, ação) (Q-Learning)
  - Estratégia: *Value Function Approximation*
    - $X^*(S_t|\theta) = \underset{\pi}{\operatorname{argmax}} F^{\pi}(\theta)$
  - Ressalva: política não determinística escolhida pelo LLM

Status da implementação

- Modelagem teórica de acordo com o framework de Powell ✅
- Instalação do ambiente WebShop (reduzido) ✅
  - Problemas técnicos para executar o benchmark
  - Treinamento dos baselines em andamento ✅
  - Issue aberta para os desenvolvedores do WebShop
    - https://github.com/princeton-nlp/WebShop/issues/29
- Plano B: ambiente Mobile-Env (WikiHow)
- Reimplementação utilizando pomdp_py
- Uso do LLM (GPT 3.5/4 ou Llama 2)

Variações

RL para LLMs

- Open AI ChatGPT (GPT 3.5/4):
  - PPO: Proximal Policy Optimization Algorithms: 2018!
  - Algoritmo para Deep Reinforcement Learning
  - PPO → RLHF (Reinforcement Learning with Human Feedback)

- RLHF
  - 1a. Etapa: geração de prompts + respostas ideais → treinamento supervisionado (instruções)
  - 2a. Etapa: prompts + respostas do LLM –> ranqueamento por humanos
  - 3a. Etapa: treinamento de um modelo de recompensa (ex.: PPO)

Alternativas em 2023

- ReAct: prompts + bases externas
- Alternativas a RLHF: RRHF, RAFT, etc.
- RRTF (Rank Responses fo Align Test&Teacher Feedback)) (Ex.: PanGu-Coder2)
  - Variação que substitui o humano por testes automatizados
  - Treinamento supervisionado p/ entropia cruzada
- AutoGen
  - Orquestração de LLM workflows
  - Conversações multi-agentes
    - LLMs podem conversar e colaborar entre si
    - Uso de entradas dos usuários (engenharia de prompts)