**Relatório: Sistema de Perguntas e Respostas sobre Dados Tabulares**

Trabalho realizado pelo aluno Ananias Corrêa para a disciplina de Inteligência Artificial do Instituto de Computação da Universidade Federal Fluminense (UFF)

1. Introdução

O projeto é sobre a tarefa de Question Answering (QA) com foco em dados estruturados, especificamente perguntas sobre países, capitais, populações e localização geográfica. Para tal, dois modelos foram treinados utilizando tanto a abordagem clássica de machine learning Decision Tree (com TF-IDF) quanto a abordagem moderna Transformers (utilizando TAPAS como modelo pré-treinado). O objetivo do trabalho é treinar os modelos para que consigam responder automaticamente perguntas sobre informações de países no geral, formuladas em linguagem natural, e assim analisar suas diferenças.

As perguntas de teste foram salvas em um arquivo .parquet, variando entre:

* Perguntas diretas: “Qual é a capital da Alemanha?”
* Perguntas booleanas: “O Brasil tem mais de 30 estados?”
* Perguntas quantitativas: “Quantos países estão na Ásia?”
* Perguntas de listas: “Quais países estão na Europa?”

Cada modelo lidou com maneiras diferentes de pré-processamento e de representação de conhecimento. As respostas finais de cada pergunta são escritas nos arquivos “predictionsD.txt”, para as predições do modelo utilizando Decision Tree, e “predictionsT.txt”, para as predições do modelo utilizando Transformers.

2. Pré-processamento

**2.1.Abordagem clássica: TF-IDF + Decision Tree**

Como base de conhecimento, o modelo utilizando Decision Tree + TF-IDF recebeu exemplos de diversas perguntas sobre países de diferentes níveis de complexidade com suas respectivas respostas, tais como “Qual é a capital do Canadá?” e “Ottawa” como resposta. A tabela de dados é representada como um DataFrame do Pandas.

**2.2.Abordagem moderna: Transformers com TAPAS**

A tabela de conhecimento é também representada como um DataFrame do Pandas, contendo as colunas País, Capital, População, Faixa Populacional, População maior que 100 milhões (Sim/Não) e Continente, ou seja, cada país possui tais dados atrelados a ele.

Foi empregado o modelo pré-treinado google/tapas-large-finetuned-wtq, especializado em responder perguntas sobre tabelas diretamente, sem necessidade de vetorização manual.

Como o TAPAS é um modelo pré-treinado com dados e perguntas em inglês, foi optado por representar o conhecimento com dados em inglês, como no exemplo do Brasil: [Country: Brazil, Capital: Brasília, Population: 213 millions, Population range: >200M, Population over 100 million: Yes, Continent: America].

3. Estratégias utilizadas

**3.1.Abordagem clássica: TF-IDF + Decision Tree**

***Vetorização:*** As perguntas utilizadas para predição são vetorizadas com o TfidfVectorizer (n-gramas de até 2 palavras), resultando em representações esparsas.

***Stopwords:***  Foram adicionadas stopwords (palavras como “de”, “o”, “a” que não carregam um significado importante à frase) a serem removidas das perguntas para reduzir ruídos nos dados e consequentemente melhorar a performance do modelo.

***Heurísticas:*** Devido à limitação do modelo em inferir contexto, uma função heurística pós-processa as respostas para traduzir booleanos para “Sim” ou “Não”, resolver perguntas como “capital de que país?” usando um dicionário invertido e lidar com listas representadas como strings.

**3.2.Abordagem moderna: Transformers com TAPAS**

***Pipeline:*** A tabela do Pandas foi passada diretamente ao modelo via TapasTokenizer, as perguntas foram codificadas e inferidas simultaneamente e o modelo retornou coordenadas das células com a resposta.

***Tradução:*** Os dados tabulares e as perguntas para predição foram traduzidas para inglês, visto que o modelo lida melhor com essa linguagem.

***Heurísticas:*** Mesmo com TAPAS, a saída bruta frequentemente não está no formato esperado. Assim, aplicou-se heurísticas como: corrigir perguntas booleanas mal interpretadas, calcular pontuação total por continente e normalizar listas/categorias.

4. Avaliação dos resultados

Analisando e testando os modelos é possível notar suas principais vantagens e desvantagens, tais como suas diferenças.

**4.1.Abordagem clássica: TF-IDF + Decision Tree**

**Vantagens:**

* Rápido e interpretável
* Responde bem as perguntas literais e diretas

**Desvantagens:**

* Não generaliza bem a novas formulações
* Erra perguntas mais complexas ou inferenciais

**Exemplos de respostas:**

Pergunta: Qual é a capital da Alemanha?

Resposta: Berlim

Pergunta: Qual é a população da França?

Resposta: 67 milhões

Pergunta: A Argentina está na Europa?

Resposta: Não

Pergunta: Liste populações maiores que 80 milhões.

Resposta: Não sei

**4.2.Abordagem moderna: Transformers com TAPAS**

**Vantagens:**

* Generaliza bem sem treinamento adicional
* Interpreta bem contextos e relações entre as colunas
* Possui uma taxa maior de acertos

**Desvantagens:**

* Mais lento para gerar respostas finais
* Necessita pós-processamento para respostas numéricas e listas
* Dá respostas “Não sei” mais frequentemente à perguntas ambíguas

**Exemplos de respostas:**

Question: What is the capital of Germany?

Answer: Berlin

Question: What is the capital of Argentina?

Answer: Buenos Aires

Question: Which countries have population over 100 million?

Answer: Brazil, United States, China, India, Russia, Japan, Mexico, Indonesia, Nigeria, Philippines, Pakistan, Ethiopia

Question: Does Brazil have a population over 500 million?

Answer: Unknown

5. Trabalhos relacionados

A tarefa de QA sobre dados tabulares é abordada por diversos trabalhos na literatura, tais como:

**Pasupat & Liang (2015)** - Propuseram a tarefa WikiTableQuestions e métodos baseados em parsing semântico.

**Herzig et al. (2020)** - Introduziram o modelo TAPAS, permitindo QA sobre tabelas diretamente com Transformers.

**TaBERT (2020)** - Modela tabelas e perguntas conjuntamente, usando BERT especializado em estruturas tabulares.

**Totto Dataset (2020)** - Ampla base de dados para aprendizado supervisionado em tabelas com explicações.

Esses trabalhos demonstram que abordagens baseadas em Transformers superam modelos clássicos, especialmente em tarefas abertas, com vocabulário livre e perguntas não estruturadas.