UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Um Estudo sobre Sistemas de Recomendação baseados em *Transformers*

João Gabriel Coutinho

Monografia - MBA em Inteligência Artificial e Big Data

ICMC USP SÃO CARLOS

João Gabriel Coutinho

Um Estudo sobre Sistemas de Recomendação baseados em *Transformers*

Monografia apresentada ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Ricardo Marcacini

Versão original

São Carlos

2022

João Gabriel Coutinho

Um Estudo sobre Sistemas de Recomendação baseados em *Transformers*

Monograph presented to the Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, as part of the requirements for obtaining the title of Specialist in Artificial Intelligence and Big Data.

Concentration area: Artificial Intelligence

Advisor: Prof. Dr. Ricardo Marcacini

Original version

São Carlos

2022

RESUMO

Coutinho, João Gabriel **Um Estudo sobre Sistemas de Recomendação baseados em *Transformers***.2022. 33p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2022.

Este trabalho apresenta um estudo acerca da utilização da arquitetura Transformer em algoritmos de recomendação. Essa arquitetura tem o diferencial de incorporar informações de sequência de acesso dos usuários de plataformas online aos itens digitais (mídia, arquivos, produtos, dentre outros). Em muitas aplicações, essas informações ajudam a estimar o próximo item de interesse do usuário e suas preferências. Para tanto foi feito um estudo da literatura acerca de sistemas de recomendação, especialmente sobre o uso recente de modelos Transformer para essa tarefa. Este trabalho descreve o desenvolvimento de quatro diferentes arquiteturas, com diferentes configurações e tamanhos de sequência para um dataset de recomendação de filmes, com o objetivo observar a desempenho e o comportamento dessa abordagem em diferentes configurações. Além disso, foi realizada uma comparação experimental com métodos de referência (baseline) que são tradicionalmente utilizados, demonstrando que a proposta de uso de Transformers é competitiva.

Palavras-chave: Arquitetura Transformer. Algoritmos de Recomendação.

ABSTRACT

Coutinho, João Gabriel **Um Estudo sobre Sistemas de Recomendação baseados em *Transformers***.2022. 33p. Monograph (MBA in Artificial Intelligence and Big Data) - Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2022.

This work presents a study about the use of the Transformer architecture in recommendation algorithms. This architecture incorporates sequential information to its items. In many applications, this information helps to estimate the user's next item of interest and preferences. Therefore, a study of the literature about recommender systems was carried out, especially about the recent use of Transformer models for this task. This work describes the development of four different architectures, with different configurations and sequence sizes for a film recommendation dataset, in order to observe the performance and behavior of this approach in different configurations. In addition, an experimental comparison was carried out with reference methods (baseline) that are traditionally used, demonstrating that the proposal to use Transformers is competitive.

Keywords: Transformer Architecture. Recommender Systems.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

O consumo de dados na última década atingiu um crescimento significativo e com isso várias plataformas baseadas em conteúdo, seja esse conteúdo produtos, imagens, vídeos, enfrentam o problema de excesso de conteúdo. Sites como Amazon contam com milhões de produtos, plataformas como Instagram contam com milhões de fotos de diferentes categorias, e aplicativos como Tik Tok possuem um grande volume de vídeos a serem consumidos por seus usuários. No entanto, cada usuário é diferente naquilo que consome e para o usuário comum seria extremamente cansativo analisar e filtrar todo o conteúdo oferecido para extrair aquilo que lhe atrai (FAYYAZ *et al.*, 2020).

Sistemas de Recomendação tem o objetivo de solucionar esse problema fornecendo ao usuário uma experiência personalizada, visando oferecer itens (produtos ou serviços) mais próximos aos seus interesses. O sistema faz isso através da predição de utilidade de um item para certo usuário baseando sua decisão a partir de interações históricas de usuários e itens (AGGARWAL *et al.*, 2016). O projeto em questão pretende estudar Sistemas de Recomendação baseados em Transformers (VASWANI *et al.*, 2017), uma arquitetura proposta em 2017 e que permitiu avanço significativo em diferentes áreas de pequisa, como NLP (Processamento de linguagem natural) e Visão Computacional.

Também é importante ressaltar que, com o desenvolvimento da área de Deep Learning, os modelos que utilizam redes neurais se mantêm como um dos estados-da-arte. Porém, essa abordagem ainda tem espaço de melhoria, pois muitas propostas recentes ignoram um fator importante: A sequência de ações do usuário. Um exemplo desse sequenciamento de ações seria um usuário que compra uma capinha de celular após ter comprado o celular.

O presente projeto estuda uma nova abordagem que incorpora os sinais sequenciais de usuários, inspirado na arquitetura de Transformers e sua habilidade em capturar as dependências posicionais. O modelo que inspirou esse estudo foi proposto inicialmente para a plataforma Alibaba (CHEN *et al.*, 2019), visando capturar as dependências entre itens selecionados por um usuário em sequência, explorando melhorias que esse novo método traz para o ramo de sistemas de recomendação. Para exemplificar o uso de Transformers em sistemas de recomendação, um método proposto pela empresa Alibaba consegue incorporar diferentes atributos do usuário e itens em sua recomendação, além de considerar a sequência em que os itens são consumidos pelo usuário. Essa estratégia também foi avaliada com sucesso em estudos mais recentes (MOREIRA *et al.*, 2021).

Também é importante ressaltar que sistemas de recomendação baseados em Transformers podem ser treinados para determinados domínios de aplicação. Nesse projeto, há

interesse em explorar tais sistemas considerando o contexto de uma rede social focada em conteúdo de jogos eletrônicos. A relação sequencial é importante para o cenário de recomendação em redes sociais para jogos, pois o usuário costuma explorar diferentes itens relacionados e jogos similares para consumir um novo conteúdo e interagir com a plataforma.

Uma segunda motivação para o estudo do método seria a análise de viabilidade da implementação de sistemas de recomendação baseados em Transformers considerando uma rede social criada por uma startup americana, a qual o autor do presente projeto está associado. Um sistema de recomendação apropriado para esse domínio de aplicação permitirá melhor engajamento dos usuários e, consequentemente, maior uso da plataforma.

1.1 Objetivos

O objetivo geral deste projeto é desenvolver um estudo sobre sistema de recomendação baseado na arquitetura Transformers para o domínio redes sociais de conteúdo de jogos. A ideia geral é avaliar o quanto a informação sequencial de navegação dos usuários impactam na qualidade da recomendação, em comparação com sistemas de recomendação mais tradicionais baseados em filtros colaborativos ou conteúdo. As seguintes questões de pesquisa foram elaboradas e serviram como base do projeto a ser conduzido:

Q1) Como incorporar o comportamento de acesso sequencial dos usuários no sistema de recomendação?

Q2) Qual a arquitetura baseada em Transformers (números de neurônios e camadas) que melhor as recomendações?

Q3) Quais os tamanhos de sequências que melhoram as recomendações?

A metodologia de pesquisa deste projeto será baseada em três grandes etapas: coleta e pré-processamento de bases de dados para recomendação; (2) extração de padrões via treinamento de modelos de recomendação baseado em Transformers; e (3) pós-processamento para medir a qualidade do modelo estudado em comparação com outros sistemas de recomendação tradicionais da literatura.

2 FUNDAMENTOS E TRABALHOS RELACIONADOS

2.1 Sistemas de Recomendação

A internet e os serviços web modernos têm aumentado nas últimas décadas, e isso tem gerado um excesso de informação (FAYYAZ *et al.*, 2020). Pode ser um desafio para os usuários filtrar todas essas informações de forma que apenas o conteúdo relevante a ele seja servido, para um usuário comum, navegar por todas as possibilidades pode ser desmotivador, podendo causar uma sobrecarga de informações. Muitas empresas de comércio eletrônico online recomendam produtos a seus usuários utilizando sistemas de recomendação (RESNICK; VARIAN, 1997; JANNACH *et al.*, 2010; BOBADILLA *et al.*, 2013; ZHANG; LU; JIN, 2021). Oferecendo milhões de produtos em uma plataforma, esses sistemas de recomendação visam resolver o problema de sobrecarga de informações enquanto personalizam a experiência do usuário, fornecendo recomendações precisas e personalizadas de itens/produtos aos usuários de acordo com suas preferências (BOBADILLA *et al.*, 2013; ZHANG; LU; JIN, 2021).

Um sistema de recomendação (RS) visa prever se um item seria útil para um usuário com base em informações fornecidas (AGGARWAL *et al.*, 2016). O uso desses sistemas tem crescido constantemente nos últimos anos, onde são usados em empresas de varejo e comércio eletrônico como eBay e Amazon. Essas empresas adquirem dados de usuários em massa e adaptam os RSs para atender às necessidades dos usuários e dos negócios. RSs são amplamente utilizados em e-commerce e varejo além de também serem utilizados em muitos outros setores, como saúde, transporte e agricultura.

2.2 Recomendações Baseadas em Conteúdo

As recomendações baseadas em conteúdo tentam construir um perfil de usuário para prever classificações em itens não vistos (PAZZANI; BILLSUS, 2007; BOBADILLA *et al.*, 2013). Por exemplo, métodos baseados em conteúdo bem-sucedidos utilizam tags e palavras-chave, gerados por meio da interação dos usuários com itens comprados e comentários (AGGARWAL *et al.*, 2016). Medir a utilidade da recomendação baseada em conteúdo é comumente calculado usando funções heurísticas, como a medida de similaridade de cosseno entre vetores de "*features*" (características ou atributos) do item e do usuário. A recomendação baseada em conteúdo pode ser empregada em muitos casos, onde os valores dos *features* podem ser facilmente extraídos, porém em casos em que os valores dos *features* devem ser inseridos manualmente essa abordagem não é indicada. Isso pode ser gerenciável para pequenos conjuntos de dados, mas quando milhares de novos produtos são adicionados diariamente, essa tarefa é impraticável.

A recomendação baseada em conteúdo geralmente não requer dados de outros usuários, pois as recomendações previstas são específicas do usuário. Além disso, ao contrário da recomendação colaborativa, a recomendação baseada em conteúdo não apresenta problemas de "*cold start*", ou seja, novos itens ou produtos são sugeridos antes mesmo que uma lista substancial de usuários atribua uma classificação (AGGARWAL *et al.*, 2016).

Por outro lado, a recomendação baseada em conteúdo tem algumas desvantagens. Por exemplo, temos o fato de que se não houver informações suficientes no conteúdo para diferenciar os produtos com precisão, a recomendação não será adequada. Na prática, essas técnicas requerem conhecimento de domínio intensivo e oferecem um grau limitado de novidade na recomendação, pois devem corresponder às características dos perfis e itens.

2.3 Recomendações Colaborativas

As Recomendações Colaborativas avaliam os produtos usando as classificações dos usuários (explícitas ou implícitas) a partir de dados históricos (WANG; YUE-XIN; CHUN-YING, 2019). As técnicas de recomendações colaborativas são classificadas em recomendações baseadas em item e recomendações baseadas em usuário. As técnicas baseadas no usuário passam por dois estágios principais para prever as classificações dos itens para um usuário específico. O primeiro estágio localiza usuários semelhantes ao usuário de destino, conforme iterações históricas na base de itens. A segunda etapa obtém taxas de usuários semelhantes ao usuário ativo, usando-as para produzir recomendações. Existem muitas medidas de algoritmos de recomendações colaborativas que calculam as semelhanças entre os usuários. As medidas de similaridade comumente utilizadas na literatura incluem diferença média quadrática, correlação de Pearson, similaridade de cosseno, correlação de Spearman e similaridade de cosseno ajustado (ZHANG; LU; JIN, 2021).

As recomendações colaborativas são comumente selecionadas para RSs e por não exigirem conhecimento de domínio. Outra vantagem das recomendações colaborativas é que elas geram modelos que ajudam os usuários a descobrir novos interesses. Ainda, são um ótimo ponto de partida para outros RSs, pois comumente requer apenas a matriz de classificação R para desenvolver um modelo de fatoração, em que R é uma matriz bidimensional de n usuários e m itens; cada entrada nesta matriz, rij representa a classificação fornecida do usuário i ao item j. Apesar de ser favorável em muitos aspectos, as recomendações colaborativas também apresentam várias desvantagens, como o problema da "cold-start".

2.4 Métodos Híbridos

Sistemas híbridos são a combinação de duas ou mais técnicas de recomendação para atingir uma performance melhor. O objetivo principal desse método é eliminar as

desvantagens oferecidas pelos métodos quando utilizados individualmente.

Um método híbrido comum envolve combinar sistemas de recomendação baseado em conteúdo e em filtragem colaborativa. A combinação pode focar nos itens que são recomendados por ambos os métodos. Outra alternativa é executar inicialmente as recomendações baseada em filtragem colaborativa para, em seguida, refinar a lista de itens recomendados por meio de filtragem baseada em conteúdo.

Um dos desafios de métodos híbridos é de fato em como combinar diferentes características. Recentemente, métodos baseados em redes neurais têm sido utilizados nessa estratégia (BATMAZ *et al.*, 2019), pois permitem aprender uma representação intermediária considerando diferentes tipos de informação, como dados do perfil dos usuários, características dos itens, acesso dos usuários aos itens, entre outros. A seguir é apresentada brevemente uma arquitetura de rede neural que tem sido empregada com sucesso em sistemas de recomendação e que é de interesse do escopo deste projeto.

2.5 Redes Neurais baseadas em Transformers

O Transformer (ou transformadores) (VASWANI *et al.*, 2017) em Processamento de Linguagem Natural é uma arquitetura que visa resolver tarefas *sequence-to-sequence* (como os dados textuais) enquanto lida com dependências de longo alcance com facilidade. Assim como as Redes Neurais Recorrentes (RNNs), os transformadores são projetados para lidar com dados de entrada sequenciais, como linguagem natural, para tarefas como tradução e resumo de texto. No entanto, ao contrário das RNNs, os Transformers não exigem que os dados sequenciais sejam processados em ordem.

Desse modo, os transformadores rapidamente se tornaram um modelo popular para problemas de PLN, substituindo modelos de rede neural recorrente, como a LSTM (Long-Short Term Memory), que era limitada para paralelização de sua execução (WANG; ZHOU; JIANG, 2020). Como o modelo Transformer facilita a paralelização durante o treinamento, ele permite o treinamento em conjuntos de dados maiores do que era possível antes de ser introduzido.

No contexto desse trabalho, o objetivo é investigar Transformers para obtenção de uma representação intermediária dos dados, denominada de embedding ou espaço latente (GOODFELLOW; BENGIO; COURVILLE, 2016). Para tal, inicialmente o processamento de um Transformer é chamada de tokenização. A tokenização é uma maneira de separar um pedaço de texto em unidades menores chamadas tokens. Aqui, os tokens podem ser palavras, caracteres ou subpalavras. Como exemplo, a frase “recommendation system is a subclass of information filtering system”, possuem os seguintes tokens: “recommendation”, “system”, “is”, “a”, “subclass”, “of”, “information”, “filtering”, “system”.

Portanto, em vez de uma sequência de elementos, agora temos um conjunto.

Conjuntos são uma coleção de elementos distintos, onde a disposição dos elementos no conjunto não importa. Os tokens, no entanto, estão fortemente correlacionados entre si. Uma das funções da arquitetura Transformers é encontrar uma representação vetorial densa para os textos. Abordagens anteriores (e.g. word2vec) (WANG; ZHOU; JIANG, 2020), obtém uma representação estática para cada token, mas Transformers obtém uma representação dinâmica, conforme o contexto do token em uma sentença, por exemplo. Dessa forma, é possível gerar uma *embedding* para palavras e sentenças. Outra característica relevante de Transformers é o mecanismo de Auto-Atenção, que permite encontrar correlações entre diferentes tokens de entrada, indicando a estrutura sintática e contextual da frase.

Embora a arquitetura Transformers tenha se tornado popular para tarefas de processamento de linguagem natural, como o modelo BERT baseado em Transformers (DEVLIN *et al.*, 2019), recentemente também tem sido empregada outros tipos de dados e aplicações, conforme discutido na próxima seção.

2.6 Sistemas de Recomendação baseados em Transformers

Muitas tarefas exploram informações de sequência dos dados para solução de problemas, como o processamento de linguagem natural. Nesse sentido, a arquitetura Transformers é uma solução popular, pois considera a informação de sequência e permite processar grandes volumes de dados por realizar o processo de forma paralela (VASWANI *et al.*, 2017). No contexto deste projeto, a arquitetura Transformers será avaliada em sistemas de recomendação, como uma forma de capturar informação ordem das ações do usuário, como navegação nos itens.

Nesse sentido, um estudo que merece destaque é apresentado por Chen *et al.* (2019). Neste estudo os pesquisadores propõem o uso de um modelo utilizando Transformers com objetivo de utilizar os os dados sequenciais dos usuários em sua recomendação. Ao contrário dos métodos anteriores, esse método leva em consideração a ordem das ações do usuário, por exemplo, quando o usuário clica em um produto ou faz uma compra, em que também é considerado todo o histórico de navegação até esta ação. Os autores assumem que a ordem dos eventos passados é importante para se prever os futuros cliques de um usuário, por exemplo, um usuário que comprou um celular tende a querer adquirir uma capa protetora para o aparelho.

A vantagem chave dos Sistemas de Recomendação baseados em Transformers é capturar as dependências entre seus itens, sejam esses itens palavras ou produtos de um site *e-commerce* como proposto pelos pesquisadores (CHEN *et al.*, 2019). Além disso, obtém uma representação de embedding para itens e para os usuários, permitindo assim calcular proximidade entre eles.

Os resultados experimentais apresentados pelos autores indicam que o uso de Trans-

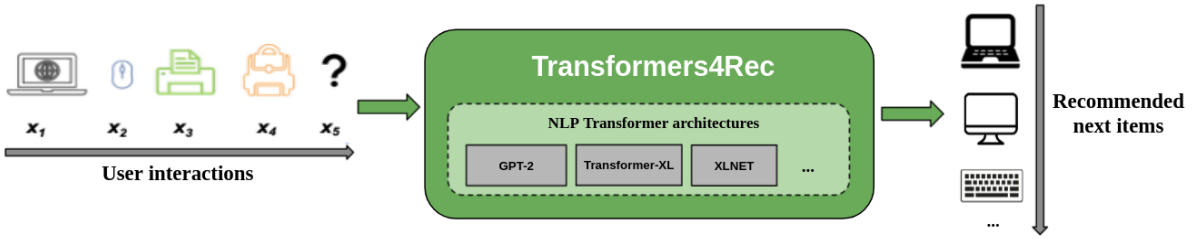

Figura 1 – Visão geral da arquitetura Transformers4Rec. Fonte: (MOREIRA *et al.*, 2021)

formers apresentou resultados superiores em relação aos métodos tradicionais, avaliados no e-commerce da plataforma Alibaba.

Resultados similares também foram obtidos no trabalho de (MOREIRA *et al.*, 2021), que propuseram o método Transformers4Rec para a tarefa de predizer o próximo *click* de um usuário. Ao obter bons resultados nesta tarefa, o modelo então pode ser utilizado para recomendar itens para novos usuários conforme seu padrão de navegação, conforme ilustrado na Figura 1.

Esses resultados motivaram a investigação dessa arquitetura para sistemas de recomendação focados em redes sociais com intuito de efetuar a recomendação de conteúdo para usuários, baseado principalmente no conteúdo em que o usuário tenha consumido sequencialmente antes disso.

3 DESENVOLVIMENTO

Os Sistemas de Recomendação (RSs) têm sido a aplicação mais popular na indústria na última década, e os métodos baseados em deep learning foram amplamente implantados em RSs industriais nos últimos cinco anos. Apesar de seu sucesso, essa abordagem é intrinsecamente insatisfatória porque ignora um tipo de sinal altamente essencial na prática, o sinal sequencial subjacente às sequências de comportamento dos usuários, ou seja, a ordem em que os usuários clicaram nos objetos, algo importante para prever os cliques do usuário no futuro. Ao longo deste capítulo será dissertado o funcionamento prático da incorporação de informações sequenciais em sistemas de recomendação e o funcionamento da arquitetura transformers para implementação dessas informações, por fim será exposto o experimento feito utilizando essa técnica, quais dados foram utilizados, os critérios de avaliação e os resultados obtidos.

3.1 Incorporando informações de sequência em sistemas de recomendação

Os sistemas de recomendação sequencial funcionam para representar e analisar sequencialmente a entrada do usuário ao longo do tempo. A interação com a entrada do usuário é baseada principalmente no sequenciamento. Isso significa, por exemplo, que quando um voo é reservado, a hospedagem e o táxi para a viagem são reservados simultaneamente. Esses dados são mantidos em ordem cronológica. O sistema oferecerá sugestões para reserva de hotel ou quarto se outra pessoa agendar um voo e um táxi. A popularidade de um item entre os usuários também flutua ao longo do tempo. Por exemplo, parece que mais pessoas estão comprando fones de ouvido sem fio e algumas pessoas desejam substituir seus telefones anualmente. Esses padrões afetam a popularidade de vários telefones, fones de ouvido e outros dispositivos. Esse perfil dinâmico é de grande importância para o perfil preciso de um usuário ou item para recomendações mais precisas e eles só podem ser capturados apenas por sistemas de recomendação Sequenciais. Em sistemas de recomendação tradicionais, como Filtragem Colaborativa e Filtragem Baseada em Conteúdo, as interações dos itens são estáticas e capturam apenas as preferências gerais do usuário. Mas no sistema de recomendação Sequencial, a interação do item é uma sequência dinâmica.

3.2 Arquitetura *Transformers*

Na recomendação sequencial, seja $U = \{u_1, u_2, ..., u_{|U|}\}$ um conjunto de usuários, $V = \{v_1, v_2, ..., v_{|V|}\}$ um conjunto de itens e $S_u = [v_1(u), ..., v_t(u), ..., v_n(u)]$ a sequência de interação em ordem cronológica para o usuário $u \in U$, onde $v_t(u) \in V$ é o item em que o usuário u interagiu no momento t e n_u é o comprimento da sequência de interação. Dado

o histórico de interação S_u, a análise sequencial tem como objetivo estimar o item que o usuário irá interagir com no passo de tempo $n_u + 1$, bem como informações de preferência sobre esse item, como o *rating*. Pode ser formalizado como modelagem da probabilidade sobre todos os itens possíveis para o usuário u na etapa de tempo $n_u + 1$, conforme definido na Equação 3.1:

$$p(v^{(u)}_{n_u+1} = v|S_u) \tag{3.1}$$

O desafio em como aproximar a função da probabilidade pode ser solucionado pela arquitetura Transformers, que visa aprender embeddings dos itens, mas respeitando-se a posição dos itens na sequência. Essas embeddings são posteriormente passados por camadas densas de neurônios que, por sua vez, é enviada para uma camada de saída final. Essa camada de saída, em geral, visa prever um *rating* do último item da sequência. Assim, o modelo pode ser usado tanto para estimar quais itens o usuário pode ter interesse para uma dada sequência, bem como estimar o *rating* desse item.

No estudo deste presente trabalho, a arquitetura Transformers foi estudada para prever *ratings* de filmes com o intuito de recomendar quais filmes os usuários possivelmente gostariam baseados nos *ratings* informados anteriormente e a sequência dos mesmos. A arquitetura do experimento pode ser visualmente representada da arquitetura geral ilustrada na Figura 2.

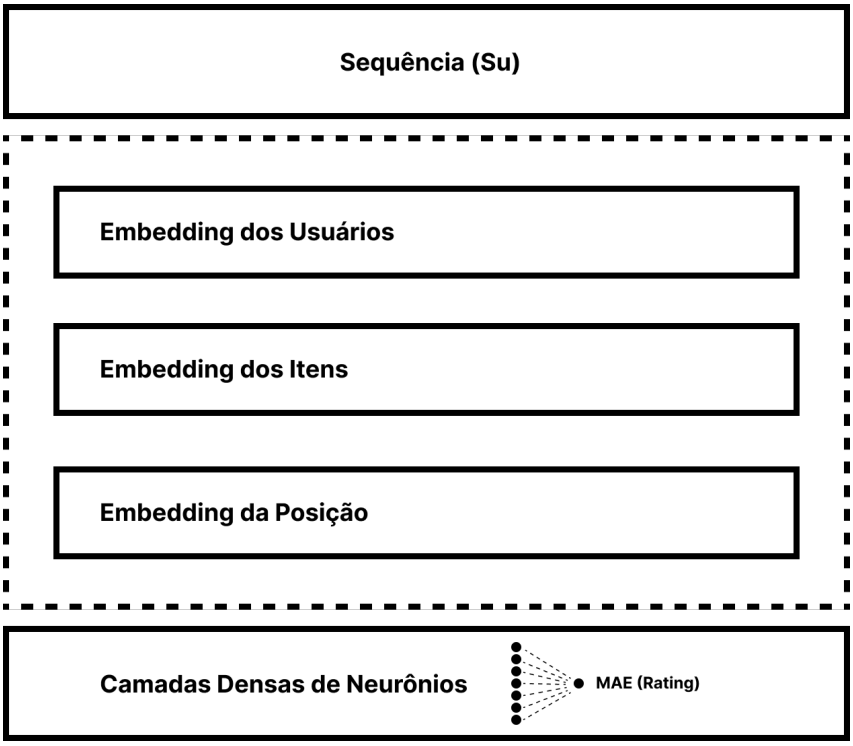

Figura 2 – Visão geral das principais estruturas obtidas por meio da arquitetura Transformers.

Como *input* temos a sequência de itens informada por um usuário, sendo ela uma lista de itens representados pelo usuário. O timestamp (momento em que o filme foi avaliado) é utilizado para determinar a sequência. A classe da instância a ser utilizada é o *rating* do último elemento da sequência, em um valor de 1 até 5. O objetivo da rede neural é estimar esse valor e é a utilizada a função MAE como função de *loss*.

Os dados do *input* passam então por um processo de embedding, na qual a arquitetura Transformers incorpora naturalmente a posição dos itens. O embedding dos usuários e itens criam uma representação vetorial, representada na Figura 3, da relação entre o usuário e um item (filme) onde os usuários se concentram mais próximos dos itens que os mesmo já tenham avaliado positivamente.

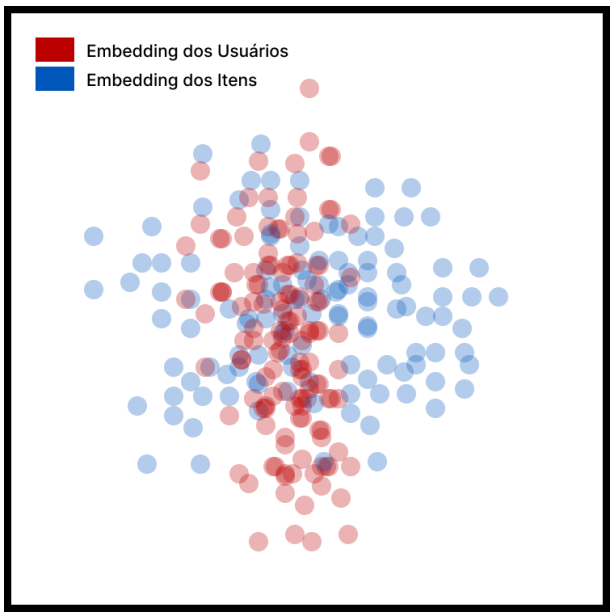

Figura 3 – Ilustração de como as embeddings de usuários e itens aprendido pela arquitetura Transformers podem ser projetadas num espaço bidimensional para analisar similaridade entre usuários e itens conforme histórico.

Vale ressaltar que o embedding de posição é uma das características principais da Transformers para essa aplicação. Essa embedding permite incluir informação de sequência na representação do problema. Informações posicionais são adicionadas ao modelo explicitamente para reter as informações sobre a ordem dos filmes avaliados pelo usuário, ou seja, a codificação posicional é o esquema pelo qual o conhecimento da ordem dos objetos em uma sequência é mantido.

3.3 Dataset

O dataset escolhido para análise experimental foi um dataset público chamado “MovieLens 1 M"onde foram coletados os inputs de mais de 6.000 usuários do site MovieLens onde os mesmos avaliavam os filmes oferecidos, no total, aproximadamente 4 mil filmes

foram julgados gerando um total de mais de 1.000.000 de avaliações para o estudo. Cada avaliação é dada por "star ratings" que variam de 1 a 5 e acompanham o timestamp de quando a mesma foi gerada, assim proporcionando a habilidade de gerar a sequência de avaliações de um usuário, que foi um dos motivos principais da escolha deste dataset, além disso foi levado em consideração o fato do dataset ser similar a vários outros datasets quanto a estrutura de seus dados (usuários, itens, ratings), por fim se levou em consideração também o fato do dataset em questão ter semelhanças com a base de dados da empresa do autor deste trabalho assim possibilitando a execução de estudo similar com a base de dados da empresa e possivelmente uma implementação de algoritmos de recomendação para servirem os usuários da mesma.

3.4 Critérios de Avaliação

O critério de avaliação utilizado durante o estudo foi o *mean absolute error (MAE)* que é uma medida de erros entre observações pareadas que expressam o mesmo fenômeno. O MAE é calculado como a soma dos erros absolutos dividido pelo tamanho da amostra.

Utilizando o critério MAE e possível medir quão distante a predição está do rating original. Dessa forma pode-se avaliar dentre uma variedade de parâmetros chave (neste estudo foi utilizado o tamanho da sequência) mais promissores, ou seja, quais valores dos parâmetros chave geram resultados que mais se aproximam do rating original.

4 RESULTADOS EXPERIMENTAIS

Para a realização da avaliação experimental foram utilizadas diferentes arquiteturas de sistemas de recomendação baseado em Transformers, tendo seus resultados comparados. As arquiteturas utilizadas se diferenciam principalmente na quantidade de parâmetros que cada uma contém, dessa forma foi possível avaliar o impacto que a quantidade de parâmetros tem nos resultados encontrados. Como referência, os resultados também foram comparados com um método baseline baseado em Filtragem Colaborativa.

A seguir são apresentadas as quatro arquiteturas implementadas e estudadas neste projeto, que são variações da arquitetura geral ilustrada na Figura 2. É importante observar que a principal alteração entre as quatro arquiteturas está no tamanho da sequência de entrada e nas camadas densas utilizadas logo após o bloco Transformers (ver Figura 2).

- Tiny: esta é a arquitetura mais simples implementada, na qual as embeddings de saída do bloco de Transformers são diretamente conectadas na camada de saída para predição do rating. Esta arquitetura possui 260.525 parâmetros ajustáveis com tamanho de sequência igual a 8.

- Small: nesta arquitetura, as embeddings do bloco Transformers são conectadas por uma camada densa de 16 neurônios, com ativação ReLU, antes da camada de saída para predição do rating. O número de parâmetros ajustáveis é de 315.211 considerando um tamanho de sequência igual a 8.

- Base: nesta arquitetura foram utilizadas duas camadas densas, com ativação ReLU, de tamanhos 128 e 64, respectivamente, logo após o bloco Transformers e antes da camada de saída para predição de rating. A arquitetura possui 441.779 parâmetros ajustáveis considerando um tamanho de sequência igual a 8.

- Large: por fim, nesta arquitetura são adicionadas três camadas densas logo após o bloco de Transformers, de tamanhos 1024, 512 e 256, respectivamente, com função de ativação ReLU. Após a última camada densa (256), é adicionada a camada de saída para predição de rating. O número de parâmetros ajustáveis da rede é de 1.662.854 considerando um tamanho de sequência igual a 8.

Para todas as arquiteturas implementadas, foi utilizada uma taxa de aprendizado em 0.01, com 30 épocas. A implementação foi realizada por meio do Framework Keras e partir de adaptações do código base disponibilizado em:

https://keras.io/examples/structured_data/movielens_recommendations_transformers/.

Para cada arquitetura e tamanho de sequência foram executadas 5 variações de sampling do dataset. A medida MAE foi utilizada nos valores não normalizados de ratings, ou seja, o erro entre o rating real (de 1 até 5) e o rating predito pode ser acima de 1.

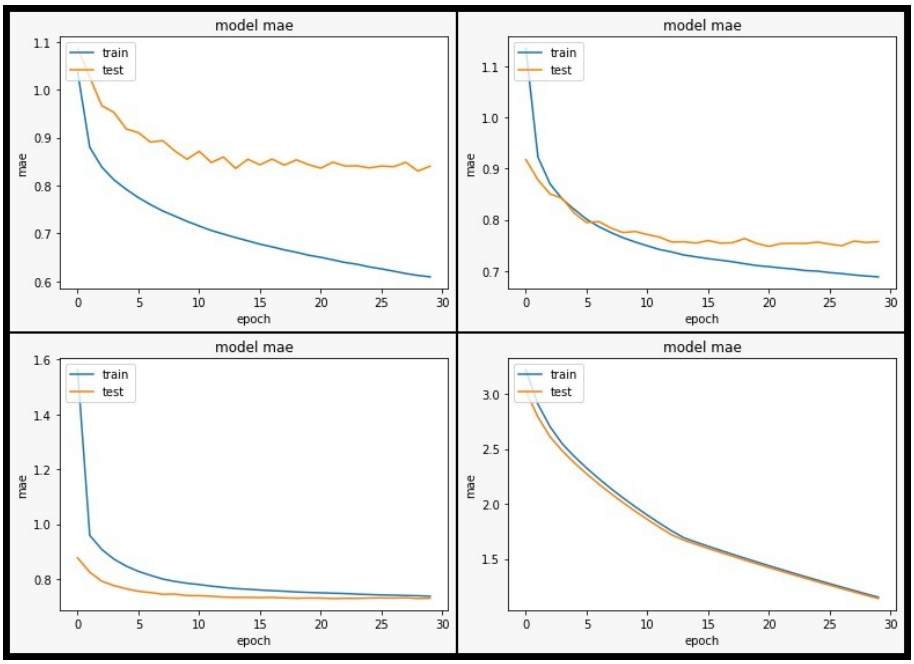

Figura 4 – Ilustração do comportamento das curvas de treinamento para uma diferentes arquiteturas utilizando uma tamanho de sequência igual a 8.

Na Tabela 1 é apresentada uma visão geral da medida MAE para cada arquitetura e tamanho da sequência, considerando valores médios e seus respectivos desvios padrão.

Tabela 1 – Valor médio da medida MAE para cada arquitetura em cada tamanho de sequência.

Arquitetura	2	4	6	8
Tiny	1,112 ± 0,010	1,128 ± 0,008	**0,756 ± 0,008**	1,146 ± 0,007
Small	0,766 ± 0,006	0,749 ± 0,007	0,744 ± 0,006	**0,737 ± 0,007**
Base	0,784 ± 0,008	**0,753 ± 0,005**	0,757 ± 0,005	0,765 ± 0,007
Large	**0,821 ± 0,006**	0,848 ± 0,006	0,855 ± 0,007	0,848 ± 0,006

Observe que cada uma das arquiteturas foram testadas com diferentes tamanhos de sequências variando de 2 a 8 itens. Analisando os resultados obtidos para cada uma dessas arquiteturas com cada uma das variações de tamanho de sequência pode-se observar que sequências muito pequenas são contra-indicadas pois obteve resultados piores em comparação a tamanhos de sequência maiores. Essa análise faz sentido considerando que as arquiteturas foram implementadas para processamento de sequência de itens.

Por outro lado, as e sequência maiores são mais difíceis de serem obtidas uma vez que os usuários em média não interagem com muitos elementos em sequência durante o uso, esse tipo de comportamento é geralmente encontrado apenas dentre os usuários que

mais utilizam uma aplicação. Assim, embora permitissem captar mais relações entre os itens, na prática pode ocorrer uma redução do conjunto de treinamento.

De forma geral, os resultados expostos fornecem evidências de sequência de tamanho 4 e 6 são, em geral, mais promissoras em reduzir o MAE.

Para uma comparação com um método de referência, como previamente mencionado, cada uma das arquiteturas propostas foram comparadas com os resultados obtidos por arquitetura baseline. Nesse caso, dado um tamanho de sequência e respectivo conjunto de treino e teste gerado, os usuários e itens desses conjuntos foram considerados para lidar com um problema de recomendação de itens baseado em Filtragem Colaborativa. Para tal, foi utilizado uma versão baseada em redes neurais denominada RecommenderNet. Na Figura 5 é possível observar como cada um dos tipos de arquitetura utilizada se compara com a arquitetura baseline, considerando o tamanho de sequência igual a 4.

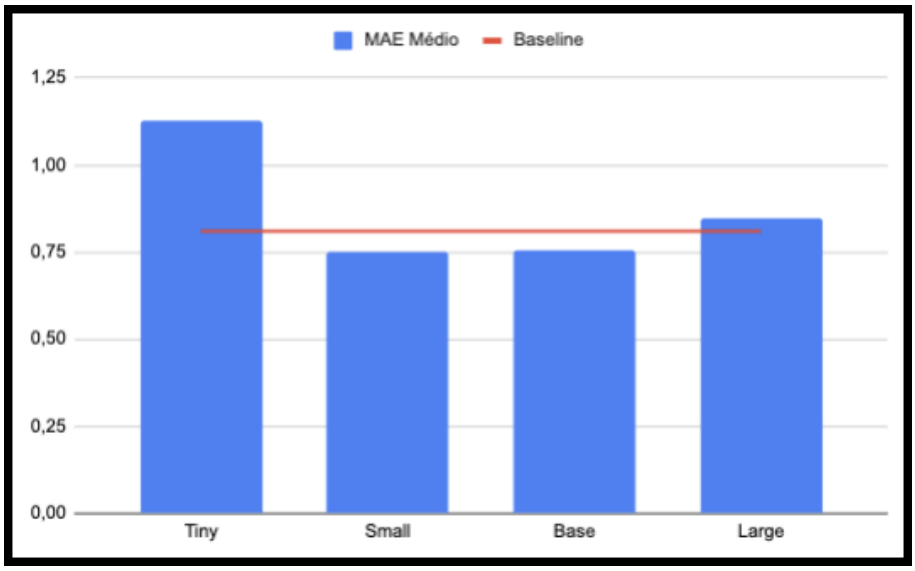

Figura 5 – Ilustração da comparação do valor médio obtido pelas arquiteturas e os resultados obtidos pelo baseline para um mesmo tamanho de sequência.

Os resultados encontrados mostram que a utilização das arquiteturas expostas em aplicações reais seria uma proposta competitiva caso a quantidade de dados fosse grande o suficiente para extração de sequência de acesso dos usuários aos itens.

Foi possível observar que uma possível aplicação realista para os modelos utilizados durante os experimentos seria em redes sociais onde os usuários interagem com o conteúdo os avaliando de alguma forma. Um exemplo disso seria a rede social Capitis, na qual o autor trabalha. Nesta rede social os conteúdos servidos aos usuários são fotos e vídeos de jogos eletrônicos onde o usuário pode avaliá-los com um rating. Assim, a avaliação dos itens se assemelha bastante com a avaliação dos itens nos dados utilizados durante a coleta dos resultados experimentais.

5 CONCLUSÃO

5.1 Visão das Contribuições e Resultados

Neste projeto foi apresentado um estudo sobre a recente arquitetura Transformers para o problema de recomendação de itens. Como contribuição e resultados, podemos destacar:

- Enquanto a maioria dos métodos tradicionais ignoram informação de sequência de acesso aos itens por um usuário, esse estudo fornece evidências de que tais informações podem ser úteis para melhorar a recomendação.
- A partir de uma arquitetura geral, foram propostas 4 variações com diferentes números de parâmetros ajustáveis. Para cada variação, também foi analisada diferentes tamanhos de sequências. Observou-se que utilizar sequências maiores pode obter melhores resultados, porém, em aplicações reais irá limitar a construção de um conjunto de treinamento.
- O código-base das arquiteturas utilizadas foi desenvolvido em Python, no ambiente Google Colab, e está disponível para a comunidade em um repositório GitHub, incluindo a estratégia para pré-processamento do dataset.

5.2 Limitações do Trabalho

A primeira limitação encontrada durante o trabalho foi a quantidade de datasets utilizados nos experimentos, uma vez que apenas um domínio foi utilizado. Para obter resultados melhores e mais precisos é recomendado utilizar mais dataset com uma maior variedade de domínios. No entanto, para isso é necessário buscar por datasets que possuam ao menos informação de timestamp de acesso dos usuários aos itens para geração das sequências de acesso.

A segunda limitação encontrada foi a necessidade da utilização de dados com tamanhos de sequência maiores para obtenção de melhores resultados. Dados com grande tamanho de sequência são mais difíceis de obter uma vez que usuários normalmente não interagem com tantos itens durante uma sessão de uso. Ainda, vale destacar que não foram consideradas uma análise da distância temporal de acesso entre um item e outro.

A terceira e última limitação encontrada foi a dependência na utilização de GPU para treinar modelos como os utilizados durante o trabalho. Em geral o uso de GPUs têm custo maior e podem limitar as aplicações desses modelos, principalmente em cenários que precisam fornecer recomendações em tempo real.

5.3 Direções para Trabalhos Futuros

Para trabalhos futuros, adequa-se a utilização de um número maior e mais variados de datasets, vez que, seria possível assim, a criação de um modelo mais generalizado e abrangente. Nesse sentido, é esperada a aplicação das arquiteturas usadas no presente estudo, em casos reais, por exemplo, estudar a aplicabilidade desta arquitetura em algoritmos de recomendação de músicas ou conteúdo em redes sociais na qual o autor do presente trabalho está atuando.

Outra direção para trabalhos futuros é explorar as embeddings geradas por este de arquitetura, uma vez que tanto usuários quanto itens são representados em um mesmo espaço vetorial. Assim, é possível projetar essas embeddings em baixa dimensionalidade para inspeção visual dos dados, bem como o emprego de outros métodos de recomendação baseado em medidas de proximidade.

REFERÊNCIAS

AGGARWAL, C. C. *et al.* **Recommender systems**. [*S.l.: s.n.*]: Springer, 2016. v. 1.

BATMAZ, Z. *et al.* A review on deep learning for recommender systems: challenges and remedies. **Artificial Intelligence Review**, Springer, v. 52, n. 1, p. 1–37, 2019.

BOBADILLA, J. *et al.* Recommender systems survey. **Knowledge-based systems**, Elsevier, v. 46, p. 109–132, 2013.

CHEN, Q. *et al.* Behavior sequence transformer for e-commerce recommendation in alibaba. *In*: **Proceedings of the 1st International Workshop on Deep Learning Practice for High-Dimensional Sparse Data**. [*S.l.: s.n.*], 2019. p. 1–4.

DEVLIN, J. *et al.* Bert: Pre-training of deep bidirectional transformers for language understanding. *In*: **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**. [*S.l.: s.n.*], 2019. p. 4171–4186.

FAYYAZ, Z. *et al.* Recommendation systems: Algorithms, challenges, metrics, and business opportunities. **applied sciences**, Multidisciplinary Digital Publishing Institute, v. 10, n. 21, p. 7748, 2020.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. [*S.l.: s.n.*]: MIT Press, 2016. http://www.deeplearningbook.org.

JANNACH, D. *et al.* **Recommender systems: an introduction**. [*S.l.: s.n.*]: Cambridge University Press, 2010.

MOREIRA, G. de S. P. *et al.* Transformers4rec: Bridging the gap between nlp and sequential/session-based recommendation. *In*: **Fifteenth ACM Conference on Recommender Systems**. [*S.l.: s.n.*], 2021. p. 143–153.

PAZZANI, M. J.; BILLSUS, D. Content-based recommendation systems. *In*: **The adaptive web**. [*S.l.: s.n.*]: Springer, 2007. p. 325–341.

RESNICK, P.; VARIAN, H. R. Recommender systems. **Communications of the ACM**, ACM New York, NY, USA, v. 40, n. 3, p. 56–58, 1997.

VASWANI, A. *et al.* Attention is all you need. *In*: **Advances in neural information processing systems**. [*S.l.: s.n.*], 2017. p. 5998–6008.

WANG, J.; YUE-XIN, L.; CHUN-YING, W. Survey of recommendation based on collaborative filtering. *In*: IOP PUBLISHING. **Journal of Physics: Conference Series**. [*S.l.: s.n.*], 2019. v. 1314, n. 1.

WANG, S.; ZHOU, W.; JIANG, C. A survey of word embeddings based on deep learning. **Computing**, Springer, v. 102, n. 3, p. 717–740, 2020.

ZHANG, Q.; LU, J.; JIN, Y. Artificial intelligence in recommender systems. **Complex & Intelligent Systems**, Springer, v. 7, n. 1, p. 439–457, 2021.