

Treinamento de Modelo de Processamento de Linguagem Natural com técnica de ‘Transfer Learning’ para classificação de interações em loja virtual

Jonathas dos Santos Lucas Vinícius de Lima Assis

21 de agosto de 2023

Sumário

1	Introdução	4
2	Preparação dos dados	5
2.1	Dados para o modelo de previsão genérico	5
2.2	Dados para o modelo de previsão de interações comerciais . . .	5
2.3	Dados para o modelo de classificação de interações comerciais	6
3	Treinamento do modelo	7
3.1	Treinando o modelo de previsão genérico	7
3.2	Treinando modelo de previsão de interações comerciais	8
3.3	Treinando o modelo de classificação de interações comerciais .	9
4	Considerações finais	9
5	Anexos	10

Lista de Figuras

1	Processo de treinamento do classificador de interações comerciais	4
2	Estrutura de pastas do <i>dataset</i> de interações comerciais	6

Lista de Tabelas

1	Tabela de métricas. Treinamento do modelo genérico.	8
2	Tabela de métricas. Treinamento do modelo especializado . . .	8

1 Introdução

O propósito deste projeto consistiu na criação de um modelo básico de processamento de linguagem natural (PLN) capaz de analisar as sentenças emitidas por clientes durante interações com vendedores. O objetivo foi categorizar se os clientes desejavam:

- receber a lista de produtos disponíveis;
- efetuar uma compra;
- solicitar um reembolso;
- iniciar uma interação;
- ou encerrar o diálogo.

Para alcançar esse objetivo, adotamos uma estratégia conhecida como *transfer learning*, na qual o conhecimento adquirido em uma tarefa é aplicado para melhorar o desempenho em uma tarefa relacionada. No nosso caso, seguimos esse procedimento ao inicialmente treinar um modelo de previsão de forma genérica, utilizando um conjunto de dados livre para uso. Posteriormente, refinamos esse modelo para se adequar às interações comerciais específicas do nosso cenário. Para concretizar esse processo, estruturamos o projeto em várias etapas, abrangendo desde a preparação dos dados e o treinamento do modelo genérico até o ajuste preciso para a situação comercial em questão. Isso culminou na atribuição de rótulos (*labels*) necessários para a criação do classificador.

Veja o esquema abaixo:

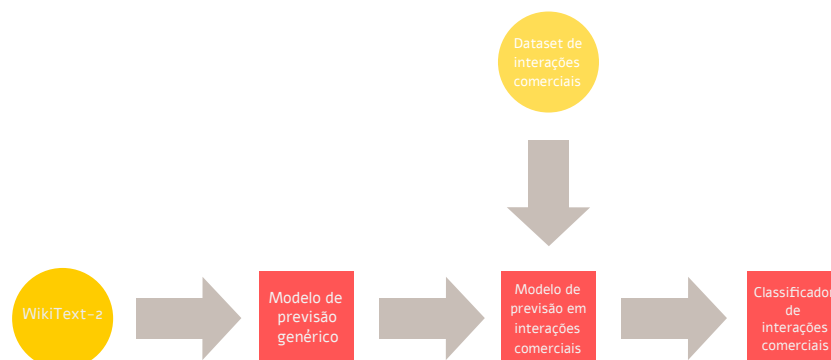


Figura 1: Processo de treinamento do classificador de interações comerciais

2 Preparação dos dados

A primeira parte da criação do classificador é a preparação dos dados. Esta etapa tem uma grande relevância, visto que é por meio desses conjuntos de informações que o modelo será capaz de ‘aprender’ e assim repetir o padrão em futuras entradas.

Para este projeto, foi necessário preparar dois conjuntos de dados diferentes. O primeiro foi direcionado para o desenvolvimento do modelo genérico, cujo propósito é prever a próxima palavra de uma sentença qualquer. Já o segundo foi utilizado na adequação do modelo anterior ao contexto de interações comerciais da perspectiva do cliente, e na criação das *labels* para o classificador.

Cada um desses *datasets* possuem características e recursos próprios para que o produto final atinja as expectativas.

2.1 Dados para o modelo de previsão genérico

O primeiro modelo a ser criado foi o de previsão genérica. Sobre seu funcionamento explico mais detalhadamente na seção 3.1. Nesta etapa, precisávamos de um *dataset* bastante robusto, visto que o vocabulário do modelo final seria formado a partir dele, assim como sua ‘compreensão’ de funcionamento da língua, ou seja, se o *dataset* fosse muito pequeno, teríamos dificuldades em criar um produto final eficiente.

Por isso, nos utilizamos de um *dataset* já existente, conhecido como WikiText-2. Ele é composto por cerca de 100 milhões de *tokens*¹ extraídos de artigos verificados da wikipedia, o que nos proporcionou um vocabulário bem amplo.

2.2 Dados para o modelo de previsão de interações comerciais

Nesta fase, adotamos nosso próprio conjunto de dados. Sua escala não precisou ser extensa, pois tanto os pesos quanto o vocabulário do modelo de previsão genérica serviriam como base para a criação do novo modelo. O conteúdo do conjunto se resume a frases de clientes em várias situações e contextos durante suas interações com vendedores. No processo de elaboração deste conjunto, nos esforçamos para simular uma ampla gama de perfis de clientes, visando conferir maior abrangência ao modelo.

¹No contexto de PLN, uma sentença é dividida em unidades de texto (*tokens*) para posterior processamento

Para a construção do conjunto de dados, criamos inicialmente um conjunto de critérios que delimitasse, de maneira simples, os tipos de clientes desejados, bem como as várias situações nas quais poderiam se encontrar. Utilizamos essas restrições como entrada para a interação com o Chat-GPT, combinando-as com as instruções adequadas. Com base nas respostas obtidas, conseguimos formar nosso conjunto completo de dados. É relevante destacar que ajustes nas restrições foram necessários para gerar as sentenças das diversas categorias. Abaixo, apresentamos alguns exemplos:

1. O pedido pode ser formal ou informal;
18. O cliente pode pedir mais de uma unidade de um mesmo produto;
24. O cliente pode ser indeciso ou não;

Veja a lista completa na página 10.

2.3 Dados para o modelo de classificação de interações comerciais

Nesta etapa, utilizamos o mesmo conjunto de dados empregado na fase anterior. No entanto, desta vez nos utilizamos da estrutura de pastas do nosso *dataset* para informar ao classificador quais frases pertenciam a cada categoria. Isso foi possível pois cada uma das classes, relacionadas na seção 1, possui uma pasta própria com seus respectivos conjuntos de texto. Essa organização específica foi adotada para orientar o modelo na associação das frases a cada classe.

O sistema de pastas utilizado é bastante simplório, como podemos ver abaixo, mas eficiente para o propósito:

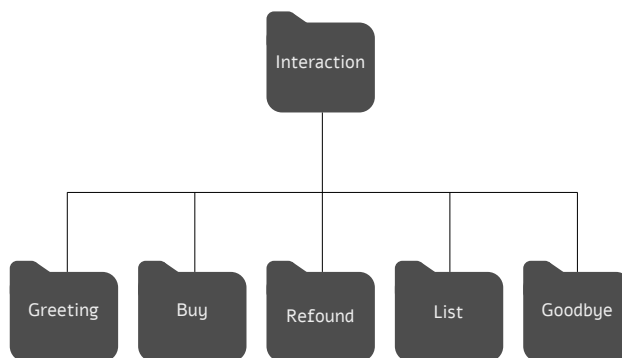


Figura 2: Estrutura de pastas do *dataset* de interações comerciais

3 Treinamento do modelo

Após realizar todas as preparações, passamos então para o treinamento do modelo propriamente dito. Da mesma forma que na preparação dos dados, podemos separar este processo em três diferentes partes: o treinamento do modelo genérico, o ajuste fino para o modelo específico, e a criação do classificador.

3.1 Treinando o modelo de previsão genérico

A etapa inicial envolveu a criação do modelo genérico. Sua principal habilidade é antecipar a próxima palavra em uma sentença qualquer, permitindo inclusive a geração de textos a partir de uma expressão inicial. A utilidade do pré-treinamento de um modelo genérico pode não ser imediatamente óbvia, mas ao examinarmos mais a fundo, torna-se claro que esse processo simplifica muito nosso trabalho.

A principal vantagem desse passo é que ele elimina a necessidade de lidar diretamente com um grande conjunto de dados específico para o nosso contexto. Para compreender isso de maneira mais clara: atualmente, é bastante desafiador encontrar um conjunto de dados que atenda nossas necessidades de forma satisfatória e que seja suficientemente amplo. Por outro lado, é muito mais fácil encontrar conjuntos de dados amplos e genéricos.

Dessa forma, podemos treinar inicialmente um modelo com um conjunto de dados abrangente, permitindo que o modelo compreenda a linguagem em sua estrutura geral e absorva seu vocabulário. Em seguida, ajustamos o modelo para um contexto específico utilizando um conjunto de dados menor. Esta estratégia é chamada de *transfer learning*.

Como mencionado na seção 2.1, utilizamos o WikiText-2 como fonte das sentenças genéricas devido à sua disponibilidade, facilidade de processamento e dimensão de vocabulário. Enquanto isso, para a etapa de especialização, utilizamos um conjunto de dados menor, de autoria própria, como dissemos na seção 2.2.

Por fim, em relação ao processo de treinamento propriamente dito, seguimos um procedimento simples. Utilizamos uma taxa de aprendizado de $3e^{-3}$, esse valor foi obtido por meio de uma função incorporada do *fastai*², que gera um gráfico relacionando a taxa de aprendizado com a perda, permitindo-nos escolher um valor dentro da faixa ideal.

A versão mais recente do modelo, no momento em que este relatório está sendo escrito, foi treinada ao longo de sete épocas. Essa quantidade

²Biblioteca de alto nível, em python, destinada ao trabalho com inteligência artificial, construída sobre o pyTorch

foi selecionada porque, embora a perda durante o treinamento estivesse consistentemente diminuindo com as iterações, a acurácia começou a declinar. Dessa forma, decidimos não treinar por muitas épocas. A tabela a seguir apresenta as métricas alcançadas:

Epoch	Train loss	Valid loss	Accuracy	Perplexity
0	3.226153	3.202698	0.407666	24.598808
1	3.037235	3.104787	0.409239	22.304470
2	2.874897	3.112789	0.407964	22.483662
3	2.743517	3.127358	0.406895	22.813631
4	2.701494	3.135628	0.406182	23.003088
5	2.662250	3.139539	0.405848	23.093229
6	2.652601	3.140287	0.405707	23.110491

Tabela 1: Tabela de métricas. Treinamento do modelo genérico.

3.2 Treinando modelo de previsão de interações comerciais

Aqui é o ponto em que utilizamos o nosso próprio *dataset*. Após carregar os dados de interações comerciais, os associamos aos pesos do modelo previamente treinado por meio de seu *encoder*³. Assim, iniciamos o processo de ajuste fino. A taxa de aprendizado aplicada foi de $2e^{-3}$, valor próximo ao do treinamento anterior. Uma vez que estamos realizando apenas um refinamento, o treinamento ocorreu ao longo de somente cinco épocas. Essa escolha foi guiada pelo mesmo motivo: embora a perda diminuísse consistentemente ao longo das iterações, a acurácia começava a apresentar declínio. Para mais detalhes, consulte a tabela de métricas a seguir:

Epoch	Train loss	Valid loss	Accuracy	Perplexity
0	2.623832	3.148892	0.404847	23.310215
1	2.652626	3.163655	0.403420	23.656910
2	2.595777	3.173937	0.402769	23.901402
3	2.555416	3.178530	0.402386	24.011419
4	2.536613	3.179537	0.402311	24.035624

Tabela 2: Tabela de métricas. Treinamento do modelo especializado

³Parte de uma rede neural que exclui apenas sua última camada

3.3 Treinando o modelo de classificação de interações comerciais

Essa fase, em termos de processamento, foi a mais ágil, já que envolveu apenas a inclusão das *labels* e um ajuste final. Nessa etapa, além de carregarmos os dados e o *encoder* do modelo anterior, estabelecemos um critério de classificação para a criação das *labels*. No nosso caso, conforme delineado na seção 2.3, a própria estrutura de diretórios do conjunto de dados foi utilizada, pois as pastas que armazenam os arquivos de texto tem como nome as classes correspondentes.

Durante esse procedimento, a biblioteca *fastai* foi de muita ajuda. Além de executar uma busca recursiva entre os diretórios, o que significa que os dados não precisavam estar todos na mesma pasta, a biblioteca também disponibiliza uma função interna para rotular as informações a partir dos nomes das pastas em que estão organizadas.

Por fim, realizamos o ajuste fino em múltiplas etapas, compreendendo cerca de 5 épocas distintas, cada uma com um conjunto diferente de camadas iniciais do modelo prévio congeladas (passo importante no *transfer learning*). A taxa de aprendizado utilizada variou, mantendo-se, contudo, na ordem de magnitude de e^{-3} .

Com a conclusão desse processo, nosso classificador estava pronto para ser exportado.

4 Considerações finais

O objetivo estabelecido para este projeto foi alcançado com sucesso. O modelo de classificação desenvolvido demonstrou uma eficácia satisfatória. É notável destacar que essa performance foi obtida mesmo considerando a utilização de *datasets* pequenos. Por exemplo, o WikiText-2 é uma versão reduzida de um *dataset* conhecido como WikiText-103, e o conjunto de dados que nós geramos contém menos de 2000 frases ao todo. Assim, existe a perspectiva de aprimorar ainda mais a eficiência do modelo ao explorar versões ampliadas dos *datasets* mencionados.

De maneira geral, a versão final do modelo não apresentou comportamentos inesperados, exceto em alguns casos particulares em que a resposta do modelo variava drasticamente ao mudarmos o produto solicitado de um item presente na lista ⁴ para um que não estava incluído. Esse fenômeno era mais evidente em frases menos convencionais relacionadas à ordem de compra.

⁴No primeiro conjunto de restrições gerado, foi fornecido uma lista de produtos que poderiam ser pedidos, baseada no projeto que utilizaria o classificador

Nessas situações, o modelo demonstrava uma inclinação para interpretar o pedido como uma solicitação de reembolso caso o produto requisitado não constasse na lista. Identificamos que essa tendência se originava do fato de que, ao contrário do conjunto de dados de ordens de compra, o conjunto de frases sobre reembolsos continha muitas sentenças nas quais produtos não eram mencionados. Assim, quando um produto não reconhecido aparecia, o modelo julgava improvável que a frase denotasse um desejo de compra. Para solucionar essa questão de forma eficaz, percebemos que bastava substituir todas as menções aos nomes dos produtos por alguma *tag* genérica, neutralizando assim essa tendência indesejada.

No que se refere à aprendizagem, este projeto, embora simples em sua abordagem, proporcionou uma compreensão mais sólida do processo de desenvolvimento de modelos de inteligência artificial, especialmente aqueles voltados para o processamento de linguagem natural.

Dentre as várias técnicas empregadas, uma em particular se destaca por sua relevância futura: a abordagem de *transfer learning*. Ao realizar este projeto, ficou claro o quão desafiador pode ser encontrar um conjunto de dados que atenda plenamente às necessidades de uma aplicação específica. Entretanto, a utilização dessa técnica revelou-se notavelmente eficaz. Por meio dela, conseguimos treinar um modelo com um conjunto de dados genérico e, em seguida, ajustá-lo utilizando um conjunto de dados menor e mais direcionado ao nosso contexto, como demonstrado neste projeto.

5 Anexos

As restrições a seguir foram originalmente criadas para o conjunto de dados de pedidos de compra. As delimitações aplicadas aos demais grupos derivaram de ajustes sutis realizados a partir deste:

1. O pedido pode ser formal ou informal;
2. O cliente pode ser objetivo ou prolixo;
3. O cliente pode ser britânico ou estadunidense;
4. O cliente pode usar marcas linguísticas de qualquer estado dos Estados Unidos;
5. O cliente pode usar marcas linguísticas de qualquer estado da Inglaterra;

6. O cliente pode ser idoso, criança, adolescente, adulto, ou seja, pode possuir qualquer idade;
7. O cliente pode pertencer a qualquer classe social;
8. O cliente pode apresentar pequenos erros na fala ou não;
9. O cliente pode se utilizar de gírias de qualquer grupo social, como skatistas, surfistas, nerds, noveleiras, bookstans, roqueiros, engenheiros, matemáticos, professores, pedreiros, jovens, etc;
10. O cliente pode se utilizar de abreviações da linguagem;
11. O cliente pode se utilizar de variantes digitais da linguagem;
12. O cliente pode se encontrar com qualquer tipo de humor, exemplo: feliz, triste, eufórico, apático, etc;
13. O cliente pode agir desrespeitosamente com vendedor;
14. O cliente pode agir respeitosamente com o vendedor;
15. O cliente pode ou não ter algum vínculo familiar ou de amizade com o vendedor;
16. Pode haver mais de um vendedor;
17. O cliente pode pedir que um produto seja entregue em sua mão ou colocado em algum recipiente de compras;
18. O cliente pode pedir mais de uma unidade de um mesmo produto;
19. O cliente pode ser homem ou mulher;
20. O cliente pode agir com diferentes níveis de intimidade para com o(s) vendedor(es), podendo ir de mais reservado, passando por alguém que age livremente na frente deles, e indo até alguém que tenta seduzir os vendedores;
21. O cliente pode incluir um pedido desconto;
22. O pedido pode estar incluído no meio de outra conversa, seja em relação a outros produtos, o clima, a vida dos vendedores, etc;
23. O cliente pode dar algum apelido ao(s) vendedor(es) ou não;
24. O cliente pode ser indeciso ou não;

25. O cliente pode ter problemas de fala ou não;
26. Você deve criar as falas levando em consideração essas cláusulas tanto de forma individual quanto fazendo combinações delas.