Database-driven Chatbot

Eduardo Neves
Universidade de Coimbra
email@example.com

Abstract

No panorama atual vemos em várias plataformas algoritmos de comunicação com pessoas. Entre muitas aplicações, estes podem ser encontrados em vários *websites* banais. A sua utilidade no mundo empresarial viu até crescer mecanismos de construção personalizada de *Chatbots*. Neste trabalho planeia-se desenvolver os mecanismos para a construção de um *Chatbot AI* a partir de *Machine Learning*.

1 Introdução

A Chatbot is

2 Dados e Abordagem

Os dados para um *Chatbot* baseiam-se principalmente em sequências de interações entre partes. Os *datasets* podem variar entre pergunta e resposta, a e-mails trocados ou conversas em plataformas online entre indivíduos. Para este trabalho foi usado um conjunto de dados baseado em em legendas de filmes para português.

2.1 Conjunto de dados

O pacote de dados usado foi tirado do projeto ”OPUS ... the open parallel corpus” [OPUS, 2022], um ’corpus’ de testos traduzidos da internet baseada em produtos ’open source’. Este conjunto é atualmente ditribuío pela plataforma ”opensubtitles” [opensubtitles, 2022], de onde foi retirado o conteúdo.

Como o foco do projeto é um algoritmo em portguês, retirou-se o dataset ”pt”, que coleciona mais de 500000 ficheiros de legendas de filmes até o ano de 2017. Separados por anos, alguns filmes contam com várias versões do mesmo filme. O seu tratamento é explicado na secção seguinte.

2.2 Tratamento dos dados

Com o conjunto de dados inicial com redundâncias e pouco estruturado, procurou-se correr um pequena organização à informação. A organização do fluxo de entrada para treino é feita aquando a inicialização do processo.

Figure 1: Formato original do *dataset* para um filme aleatório

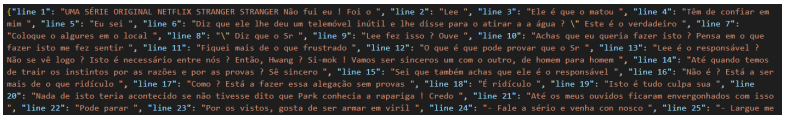

Figure 2: Formato alterado de um filme aleatório do conjunto

Redução de redundâncias

Como referido, algumas versões do mesmo filme são dispostas e organizadas em conjunto. Ao remover este entrave, pode-se remover uma camada na diretoria e agrupar apenas por ano. Apesar desta diferenciação não ser necessária poderá ser útil em comparações ortográficas ou até numa maior confiança de traduções mais recentes. Para selecionar o ficheiro mais relevante apenas se retirou o ficheiro com menor volume de espaço no disco para ’performance’.

Alteração do formato

O *dataset* original continha apenas ficheiros em .xml, com muitos elementos, como tempos e personagens, desnecessárias a este trabalho. Outra especificação sem relevância é a separação por palavras das falas de cada personagem. Simplificou-se então para um formato .json, onde cada entrada é a fala de uma personagem. Manteve-se a separação dos ficheiros por filmes.

Redução de amostragem

Para além da redução na alteração do formato, reduziu-se também a quantidade de dados usados para o programa. Utilizou-se apenas filmes a partir de 2000, o que resultou em 37851 filmes, com uma média de 455 falas de personagens. Este ainda assim é um número muito grande de filmes, mas manteve-se para manuntenção da utilidade do conjunto, além de uma maior escolha entre filmes de váriоa anos para possível teste.

Preparação para treino
Na língua portuguesa deparamo-nos com vários acentos e outros símobolos como os traços (como exemplo "ensinote"). Quanto aos acentos, a sua mantenção é imperativa para boa leitura e para remoção de confusões entre palavras (como por exemplo "estás" e "estas"). Por outro lado, os traços foram retirados, já que a flexão do pronomes quanto ao número e género é demasiado vasta para garantir um vocabulário conciso por parte do modelo. Outros símbolos de síntaxe e gramaticais foram retirados aquando a preparação para treino.

2.3 Modelo

O modelo base foi retirado da plataforma Github, do repositório Seq2Seq-Chatbot [Sojasingarayar, 2020]. Este modelo usa o módulo *tensorflow* para os processos de ML.

No trabalho citado foi também usado um *dataset* de filmes, o 'Cornell Movie Dialog Corpus' [Danescu-Niculescu-Mizil and Lee, 2011]. Este dispunha de 304713 entradas para treino num ficheiro .txt também disponibilizado.

LSTM
A 'Long Short Term Memory' (LSTM) é uma rede neuronal recorrente capaz de aprender a dependência em problemas de predição. Este é útil em tradução, reconhecimento de fala e, direcionado para este tópico, simulação de conversação humana. Mais sobre este tipo de redes, elas têm um estado interno que consegue representar informação em contexto [Bengio *et al.*, 1994].

Seq2seq
A metodologia 'sequence-to-sequence' (seq2seq) é a peça-chave do trabalho. A metodologia utiliza modelos de 'Machine Learning' (ML) para obeter um sequência como entrada, num domínio, e convertê-la para uma representação noutro domínio.

O *seq2seq* baseia-se num bloco *encoder* que lê a série de entrada a partir de um vetor de dimensionalidade fixa e um bloco *decoder* que extrai a frase. Ambos estes blocos são células LSTM e são treinados ao mesmo tempo [Sutskever *et al.*, 2014]. Entre eles há um vetor de contexto que encapsula todo o sentido da frase.

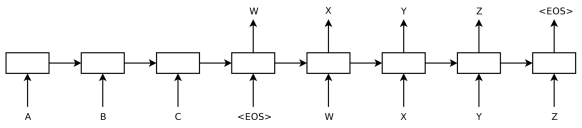

Figure 3: Ilustração da metodologia. os primeiros quatro retêngulos referem-se à camada *encoder* enquanto o resto pertence à *decoder* [Sutskever *et al.*, 2014]

No momento inicial, quando é fornecida uma sequência ao modelo, a frase terá de caber no vetor fixo, portanto leva um *padding*. À sequência acrescenta-se a componente "<PAD>". Ao passar pelo vetor, o *decoder* inicia com o comando "<START>", ou "<GO>" como usado neste trabalho, para iniciar a produção da saída. A sequência é completa com a expressão "<EOS>" que simboliza o final do *output*. Para expressões que o algoritmo não reconhece lança-se o comando "<UNK>".

'Attention'
Em ML esta técnica tem o objetivo de reproduzir atenção cognitiva (comportamento humano). o efeito traduz-se em realçar algumas partes da entrada, enquanto diminui outras, mudando o foco da informação dada. Este método é também sensível ao contexto, algo que é tido em conta no treino da máquina.

3 Experimentação e Métricas

3.1 ROUGE

'Recall-Oriented Understudy for Gisting Evaluation' é um conjunto de métricas espcializadas em avaliar *machine translation*. Estas métricas focam-se em quanto as palavras (ou *n-grams* no input se assemelham às previstas pelo modelo.

Neste trabalho foca-se no ROUGE-L que se baseia na subsequência mais longa em comum entre o *output* e a referência. Uma vantagem do uso desta métrica é que não é necessário ter correspondências consecutivas, mas correspondências em sequência [Lin and Och, 2004]. A implementação é feita com recurso ao módulo "rouge" do *python*.

4 Resultados

5 Problemas e Resoluções

5.1 Conjunto de dados

Alguns problemas foram encontrados no uso da informação.

Em primeiro lugar, muitos ficheiros continham referências aos tradutores como parte das legendas. Como estas variavam em formato e local era difícil prever e retirar estas informações com sucesso.

Algumas traduções eram também desvirtuadas do português Portugal, tais como o uso de expressões como "em a", em vez de "na" em vários locais. A fonte oferecia também um pacote de português Brasil, portanto estas nuances não se esperariam no conjunto usado. Pequenas alterações como esta seriam de difícil execução sem uma análise extensiva dos dados.

Resolução
Apesar destes problemas, usou-se este 'dataset' pela sua vastidão e facilidade de acesso. Acrescendo a dificuldade de materiais para a língua desejada, esta foi a melhor opção encontrada.

5.2 Modelo

A escolha do código foi influenciada na metodologia seq2seq (aconselhada pelo professor). A pesquisa foi então restringida para tal.

O uso desta versão 1.14 do *tensorflow* é desatualizada, pois o pacote está já na versão 2 a partir de 2019. Esta veio com uma mudança na organização e, portanto, na sintaxe a usar para construir o modelo. Obriga, também, ao uso de uma versão anterior do python (3.6 ou menor).

Este modelo traz então algumas agravantes, em especial o tempo de compilação. Para 500 filmes com 100 *epochs*, o

tempo estimado era de cerca de 13 horas, todo o processo de treino. Com estes números o modelo leva a um esforço computacional aquém do esperado, tornando-se até inexequível em termos de desenvolvimento do trabalho.

Guardar os pesos ocupa muito espaço!!!!?

6 Conclusions

References

[Bengio *et al.*, 1994] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[Danescu-Niculescu-Mizil and Lee, 2011] Cristian Danescu-Niculescu-Mizil and Lillian Lee. Chameleons in imagined conversations: A new approach to understanding coordination of linguistic style in dialogs. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics, ACL 2011*, 2011.

[Lin and Och, 2004] Chin-Yew Lin and FJ Och. Looking for a few good metrics: Rouge and its evaluation. In *Ntcir workshop*, 2004.

[opensubtitles, 2022] opensubtitles. opensubtitles.org, 2022.

[OPUS, 2022] OPUS. OPUS the open parallel corpus, 2022.

[Sojasingarayar, 2020] Abonia Sojasingarayar. Seq2seq-chatbot. https://github.com/Abonia1/Seq2Seq-Chatbot, 2020.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27, 2014.