

Comparação dos modelos BERT e Snips para compreensão de linguagem natural para a assistente virtual ADA*

Leonardo Costa Santos¹, Antonio Deusany de Carvalho Junior¹, Alfredo Goldman¹

¹Instituto de Matemática e Estatística - IME - USP
Rua do Matão, 1010 – São Paulo – SP – Brazil

leonardo_c_santos@usp.br, {dj, gold}@ime.usp.br

Abstract. *Natural Language Understanding is an important area of Natural Language Processing, concerned with understanding and extracting semantic information from text or speech. It is an essential component of any modern virtual assistant being responsible for identifying the intent of the user and the objects of their speech. In this work we compared two models, BERT and Snips, for natural language processing. These models were tested with the goal of being integrated into the A.D.A. virtual assistant, performing the tasks of intent classification and slot filling. As preliminary results, we obtained high precision in both tasks with the tested solutions.*

Resumo. *Compreensão de linguagem natural, é uma subárea importante de Processamento de Linguagem Natural em Inteligência Artificial, que busca entender e extrair informação semântica de um texto ou fala. Ela é um componente essencial de qualquer assistente virtual moderna, sendo responsável por identificar a intenção do usuário e os objetos de sua fala. Sendo assim, este trabalho apresenta uma comparação de dois modelos, sendo estes o BERT e Snips, para compreensão de linguagem natural. Tais modelos foram testados com o intuito de serem integrados à assistente virtual A.D.A., realizando as tarefas de classificação de intenção e slot filling. Como resultados preliminares, conseguimos obter alta precisão em ambas as tarefas com as soluções testadas.*

1. Introdução

A A.D.A é uma assistente virtual distribuída baseada em software livre, projetada para compreender comandos em português e interagir com dispositivos da área de Internet das Coisas (IoT, do inglês *Internet of Things*) [Freire et al. 2020]. Para que a A.D.A. possa executar os comandos do usuário é necessário o uso de um componente de compreensão de linguagem natural para traduzir a fala do usuário para uma linguagem ou representação formal, permitindo que os comandos do usuário sejam interpretados e executados pela assistente.

Para isso, foram testadas e comparadas duas soluções para compreensão de linguagem natural bem utilizadas da literatura: um modelo que utiliza a representação de linguagem contextual do BERT em português, e a biblioteca snips_nlu da Snips Voice Platform. Elas realizam as tarefas de *intent classification* e *slot filling*, isto é, identificar o

*Este projeto foi financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e pelo grupo de extensão CodeLab uclab.xyz/site

tipo de ação desejado pelo usuário e os termos da frase que correspondam à entidades ou objetos relevantes a essa ação. Ambas as soluções já atingiram resultados de *state-of-the-art* nessas tarefas em inglês [Coucke et al. 2018, Chen et al. 2019].

Conseguimos obter precisão acima de 90% em ambas as tarefas com as soluções testadas. Assim, tanto a `snips_nlu` quanto o BERT são adequados para serem integrados à A.D.A. como será apresentado no decorrer deste artigo. Além disso, nas próximas seções vamos descrever os temas relacionados, a metodologia utilizada para realizar a comparação dos modelos, e esses resultados obtidos até então.

2. Referencial teórico

Processamento de Linguagem Natural (NLP, do inglês *Natural Language Processing*) é a área da Inteligência Artificial que busca analisar e entender linguagem natural, i.e. aquelas linguagens faladas por seres humanos e que, por terem evoluído naturalmente, não podem ser representadas por um conjunto de regras. Isto se dá em contraste com linguagens formais, que são definidas por conjuntos de regras e portanto podem ser processadas por algoritmos [Jurafsky and Martin 2009].

Uma subárea importante de NLP é a Compreensão de Linguagem Natural (NLU, do inglês *Natural Language Understanding*), que busca converter um texto ou fala em linguagem natural para uma linguagem formal. Isto permite que um computador extraia informações de um texto que podem ser usadas para tarefas como *machine translation*, busca de informações em texto ou para implementação de interfaces de voz. Com base nas tecnologias citadas acima, foram testadas algumas soluções da área que podem ser utilizadas para modelagem de dados e interação em assistentes virtuais. Tais soluções são descritas a seguir.

2.1. Snips-NLU

A primeira solução testada foi a biblioteca de compreensão de linguagem natural Snips-NLU, criada como parte da Snips Voice Platform, uma ferramenta criada para realizar compreensão de fala em microprocessadores de dispositivos de IoT [Coucke et al. 2018]. A biblioteca possui suporte para várias línguas, inclusive português brasileiro ¹.

A *engine* da biblioteca utiliza dois modelos para classificação de intenção e extração de *slots*: um modelo determinístico e um probabilístico. O modelo determinístico garante a eficácia em sentenças que são parte do dataset de treinamento através da geração de representações de padrões com os quais as entradas são comparadas. O modelo probabilístico possui duas partes: um modelo de regressão logística, que realiza a classificação da intenção da fala do usuário; e um conjunto de *linear-chain conditional random fields* (CRFs), cada um treinado para reconhecer os *slots* de uma intenção.

2.2. BERT (*Biredirectional Encoder Representations from Transformers*)

BERT é um modelo de representação de linguagem criado para pré-treinamento de representações contextuais bidirecionais [Devlin et al. 2018]. Uma representação contextual de linguagem é uma representação vetorial que preserva a informação semântica

¹Linguagens suportadas pela Snips: <https://snips-nlu.readthedocs.io/en/latest/languages.html>

do vocabulário de um corpus. O BERT gera representações para cada *token* da entrada, e também uma representação da sequência de entrada como um todo para uso em tarefas de *sequence classification*.

3. Metodologia

Para a realização dos testes, foi implementado um modelo baseado no BERT que realiza classificação de intenção e *slot filling* de maneira conjunta, com base no trabalho de [Chen et al. 2019]. A performance deste modelo é comparada com a da biblioteca *snips_nlu*, sendo ambos treinados com o mesmo *dataset*. O *dataset* criado para o treinamento consiste de 5060 frases, apresentando 4 intenções no contexto de uma *smarthouse*, e contém 4 tipos de entidades para *slot filling*.

A biblioteca *snips-NLU* está disponível em Python e fornece funções para instanciar, treinar e utilizar o modelo, que retorna um dicionário com a intenção e os *slots* já processados.

O modelo BERT utilizado foi o BERTimbau [Souza et al. 2020], implementado com Pytorch, que está disponível através da biblioteca Transformers da HuggingFace [Wolf et al. 2020]. Para classificação de intenção, a saída do token de classificação do BERT é usada como entrada para um modelo linear com softmax. Para *slot filling*, foi utilizada uma camada linear e uma CRF, utilizando a saída por token do BERT.

Os conjuntos de dados de treinamento e teste contêm, respectivamente, 4048 e 1012 frases. As métricas utilizadas para avaliação são a precisão da classificação de intenção e a pontuação F1 da classificação de slots. O modelo com BERT foi treinado por 3 *epochs*, em *batches* de 16 frases, com uma *learning rate* de $1e-3$.

4. Resultados e discussão

As métricas utilizadas para medir a performance dos modelos nas tarefas testadas foram a precisão (*accuracy*) da classificação de intenção, que é o percentual de frases classificadas corretamente, e a pontuação F1 da classificação de slots, que é a média geométrica das métricas de *precision* e *recall*. A partir dos testes realizados, obtivemos os seguintes resultados que são apresentados na Tabela 1. Nestes testes, os dois modelos conseguiram performances acima de 90% em ambas as tarefas. Em classificação de intenção o modelo com BERT obteve uma precisão menor (94.76%) que a biblioteca *snips_nlu* (100%). Já em *slot filling* o BERT obteve F1 de quase 100%, comparados aos 96.74% da *snips_nlu*.

Modelo	Intenção	Slots
BERT + CRF	94.76	99.89
<i>snips_nlu</i>	100.0	96.74

Tabela 1. Performance do modelos com base na precisão da classificação de intenção (%) e no F1 para os slots (%)

Para integração com a A.D.A., ambas as soluções obtiveram bons resultados nas tarefas testadas. O modelo com BERT pode ser mais adequado para comandos menos previsíveis ou com estrutura mais variável, como busca online, e também pode ser utilizado para uma quantidade maior de tarefas, o que o faz um candidato melhor para uso com a A.D.A., considerando um futuro aumento no número de habilidades da assistente.

Já a `snips_nlu`, por ter sido criada para funcionar *offline* em sistemas embarcados, apresenta um uso menor de memória e executa mais rapidamente que o modelo com BERT. Porém, a biblioteca é feita para lidar apenas com as tarefas apresentadas, o que limita as opções de uso.

5. Conclusão

Foram testadas e comparadas duas soluções de compreensão de linguagem natural, que realizam as tarefas de classificação de intenção e *slot filling*, com o objetivo de escolher uma solução para ser integrada à assistente virtual A.D.A.. Os testes mostram que tanto a biblioteca `snips_nlu` quanto o modelo que utiliza o BERT obtêm bons resultados e, portanto, são boas opções para serem utilizadas com a A.D.A. dependendo do contexto. O BERT é mais adequado para comandos menos previsíveis, enquanto a `snips_nlu` consome menos memória e é executada mais rapidamente.

Como trabalhos futuros, pretendemos melhorar os resultados com o BERT porque ele é um modelo que generaliza melhor e pode ser usado para acrescentar mais capacidades à A.D.A. em novas versões da mesma. Também buscaremos acrescentar novas intenções possíveis e tipos de *slots* para tornar a assistente mais robusta e interativa.

Referências

- Chen, Q., Zhuo, Z., and Wang, W. (2019). Bert for joint intent classification and slot filling. arXiv. eprint 1902.10909.
- Coucke, A., Saade, A., Ball, A., Bluche, T., Caulier, A., Leroy, D., Doumouro, C., Gisselbrecht, T., Caltagirone, F., Lavril, T., et al. (2018). Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *arXiv preprint arXiv:1805.10190*, pages 12–16.
- Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Freire, F., Rosa, T., Feulo, G., Elmadjian, C., Cordeiro, R., Moura, S., Andrade, A., de Omena, L. A., Vicente, A., Marques, F., Sheffer, A., Hideki, O., Nascimento, P., Cordeiro, D., and Goldman, A. (2020). Toward Development of A.D.A. – Advanced Distributed Assistant. In *Anais do XXI Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 203–214, Porto Alegre, RS, Brasil. SBC.
- Jurafsky, D. and Martin, J. H. (2009). *Speech and Language Processing (2Nd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- Souza, F., Nogueira, R., and Lotufo, R. (2020). BERTimbau: pretrained BERT models for Brazilian Portuguese. In *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*.
- Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.