Universidade Federal do Rio Grande do Norte

Instituto Metrópole Digital

DIM0135 - Processamento de Linguagem Natural

# Part-of-Speech Tagger: Bidirectional LSTM

Com embeddings pré-treinados

João Vitor Venceslau Coelho

Natal/RN

2022

## Introdução

Este relatório tem como objetivo apresentar a abordagem utilizada para a tarefa de **Part-of-Speech Tagging**, utilizando modelos baseados em **Redes Neurais Recorrentes (RNNs)**. Em específico, um modelo criado utilizando a biblioteca **TensorFlow** e outro utilizando a biblioteca **Flair**, que utiliza o **PyTorch** internamente.Ambos os modelos utilizam uma camada de **LSTM Bidirecional (Bidirectional Long Short-Term Memory)** na sua arquitetura. Serão apresentadas as diferenças no pré-processamento para ambos os modelos e os resultados obtidos, que são brevemente discutidos. O Corpus selecionado foi o Penn Treebank, com as seções de 0 a 18 para treino, 19 a 21 para a etapa de validação/desenvolvimento e as seções de 22 a 24 para os testes finais.

## Primeiro Modelo

Utilizando o TensorFlow com embeddings pré-treinados do modelo NNLM possuindo 50 dimensões

### Pré-processamento

Os dados foram lidos e armazenados em um array de sentenças, onde cada sentença é armazenada como um array de tokens, então todas as sentenças com mais de 100 tokens foram removidas dos conjuntos de treino, validação e teste, mantendo apenas as que possuem até 100 tokens. Foi então aplicado um processo de tokenização, atribuindo a cada token um número de identificação e para tokens desconhecidos é atribuído o número de identificação de um token especial o **<unk>**. Em seguida, os arrays de sentenças são completados com padding após o último token de cada sentença, para que todas tenham o mesmo tamanho de 100 tokens, o número identificador do pad é 0, um valor não utilizado no passo da tokenização realizado anteriormente.

### Embeddings

Os embeddings escolhidos possuem 50 dimensões e estão disponíveis neste [link](https://tfhub.dev/google/nnlm-en-dim50/2) como um modelo/camada do TensorFlow, é um embedding baseado em tokens treinado no corpus de Notícias do Google em Inglês, utilizando uma rede neural Feed-Foward.

Cada token do vocabulário dos dados de treino foi aplicado no modelo para obter os embeddings do token, o resultado foi armazenado na matriz de inicialização de embeddings da camada de embeddings do modelo.

### Arquitetura

A arquitetura consiste na camada inicial de embeddings, que com base no identificador de cada token, retorna o embedding do mesmo, seguida de uma camada bidirecional de LSTM com 256 unidades e por fim a camada final com softmax para obter o resultado do PoS tagging.

### Treinamento

Como parâmetros de treinamento, foi utilizado o otimizador Adamax com learning rate de 0.001, a métrica de loss foi a Categorical Cross Entropy e as métricas de avaliação escolhidas foram a Accuracy, Precision e Recall. O modelo foi treinado por 5 epochs em batchs de 32 e para reduzir o tempo de treinamento foi utilizada a GPU do Colab, totalizando 7 minutos e 43 segundos de treino.

### Resultados obtidos no conjunto de teste

| **tag** | **precision** | **recall** | **f1-score** | **support** |
| --- | --- | --- | --- | --- |
| pad | 1,000 | 1,000 | 1,000 | 416580 |
| nn | 0,000 | 0,000 | 0,000 | 17792 |
| in | 0,000 | 0,000 | 0,000 | 13472 |
| nnp | 0,000 | 0,000 | 0,000 | 13145 |
| dt | 0,000 | 0,000 | 0,000 | 11162 |
| jj | 0,000 | 0,000 | 0,000 | 8198 |
| nns | 0,000 | 0,000 | 0,000 | 8047 |
| , | 0,000 | 0,000 | 0,000 | 6865 |
| . | 0,000 | 0,000 | 0,000 | 5379 |
| cd | 0,000 | 0,000 | 0,000 | 4816 |
| rb | 0,000 | 0,000 | 0,000 | 4401 |
| vbd | 0,000 | 0,000 | 0,000 | 4560 |
| vb | 0,000 | 0,000 | 0,000 | 3563 |
| cc | 0,000 | 0,000 | 0,000 | 3239 |
| to | 0,000 | 0,000 | 0,000 | 2905 |
| vbz | 0,000 | 0,000 | 0,000 | 2639 |
| vbn | 0,000 | 0,000 | 0,000 | 2707 |
| prp | 0,870 | 1,000 | 0,930 | 2203 |
| vbg | 0,000 | 0,000 | 0,000 | 1929 |
| vbp | 0,000 | 0,000 | 0,000 | 1565 |
| md | 0,000 | 0,000 | 0,000 | 1266 |
| pos | 0,000 | 0,000 | 0,000 | 1276 |
| prp$ | 0,000 | 0,000 | 0,000 | 1068 |
| $ | 0,000 | 0,000 | 0,000 | 940 |
| `` | 0,000 | 0,000 | 0,000 | 1072 |
| '' | 0,000 | 0,000 | 0,000 | 1043 |
| : | 0,000 | 0,000 | 0,000 | 746 |
| wdt | 0,000 | 0,000 | 0,000 | 584 |
| jjr | 0,000 | 0,000 | 0,000 | 423 |
| rp | 0,000 | 0,000 | 0,000 | 397 |
| nnps | 0,000 | 0,000 | 0,000 | 170 |
| wp | 0,000 | 0,000 | 0,000 | 283 |
| wrb | 0,000 | 0,000 | 0,000 | 304 |
| jjs | 0,000 | 0,000 | 0,000 | 267 |
| rbr | 0,000 | 0,000 | 0,000 | 271 |
| -rrb- | 0,000 | 0,000 | 0,000 | 181 |
| -lrb- | 0,000 | 0,000 | 0,000 | 180 |
| ex | 0,000 | 0,000 | 0,000 | 126 |
| rbs | 0,000 | 0,000 | 0,000 | 69 |
| pdt | 0,000 | 0,000 | 0,000 | 44 |
| fw | 0,000 | 0,000 | 0,000 | 28 |
| wp$ | 0,000 | 0,000 | 0,000 | 37 |
| # | 0,000 | 0,000 | 0,000 | 15 |
| uh | 0,000 | 0,000 | 0,000 | 17 |
| sym | 0,000 | 0,000 | 0,000 | 11 |
| ls | 0,000 | 0,000 | 0,000 | 15 |
| macro\_avg | 0,040 | 0,040 | 0,040 | 546000 |
| weighted\_avg | 0,770 | 0,770 | 0,770 | 546000 |

Com acurácia total de 0.7670 e F1-Score macro e ponderado de 0,040 e 0,770. Se o pad for considerado como tag, o que é incorreto.

Porém vale destacar que os resultados obtidos durante a validação são totalmente diferentes do que ocorreu durante o teste:

| **tag** | **precision** | **recall** | **f1-score** | **support** |
| --- | --- | --- | --- | --- |
| pad | 1,000 | 1,000 | 1,000 | 420932 |
| nn | 0,920 | 0,910 | 0,920 | 18538 |
| in | 0,990 | 0,970 | 0,980 | 13651 |
| nnp | 0,820 | 0,900 | 0,860 | 13183 |
| dt | 0,990 | 0,990 | 0,990 | 11230 |
| jj | 0,870 | 0,820 | 0,840 | 8094 |
| nns | 0,970 | 0,930 | 0,950 | 8043 |
| , | 1,000 | 1,000 | 1,000 | 7087 |
| . | 1,000 | 1,000 | 1,000 | 5468 |
| cd | 0,910 | 0,960 | 0,940 | 5223 |
| rb | 0,880 | 0,920 | 0,900 | 4171 |
| vbd | 0,930 | 0,970 | 0,950 | 4621 |
| vb | 0,950 | 0,940 | 0,950 | 3503 |
| cc | 1,000 | 0,990 | 0,990 | 3380 |
| to | 1,000 | 1,000 | 1,000 | 3074 |
| vbz | 0,960 | 0,950 | 0,950 | 2815 |
| vbn | 0,900 | 0,840 | 0,870 | 2825 |
| prp | 0,990 | 0,990 | 0,990 | 2386 |
| vbg | 0,890 | 0,820 | 0,850 | 1956 |
| vbp | 0,940 | 0,910 | 0,930 | 1486 |
| md | 0,990 | 1,000 | 0,990 | 1293 |
| pos | 0,990 | 1,000 | 0,990 | 1241 |
| prp$ | 1,000 | 1,000 | 1,000 | 1184 |
| $ | 1,000 | 1,000 | 1,000 | 1144 |
| `` | 1,000 | 1,000 | 1,000 | 1022 |
| '' | 0,990 | 1,000 | 1,000 | 991 |
| : | 1,000 | 0,990 | 1,000 | 639 |
| wdt | 0,950 | 0,970 | 0,960 | 545 |
| jjr | 0,840 | 0,820 | 0,830 | 445 |
| rp | 0,750 | 0,690 | 0,720 | 363 |
| nnps | 0,610 | 0,340 | 0,440 | 412 |
| wp | 0,990 | 0,990 | 0,990 | 319 |
| wrb | 1,000 | 0,990 | 1,000 | 270 |
| jjs | 0,940 | 0,850 | 0,900 | 262 |
| rbr | 0,730 | 0,650 | 0,690 | 229 |
| -rrb- | 1,000 | 1,000 | 1,000 | 181 |
| -lrb- | 1,000 | 1,000 | 1,000 | 181 |
| ex | 0,970 | 0,990 | 0,980 | 118 |
| rbs | 0,790 | 0,750 | 0,770 | 51 |
| pdt | 0,910 | 0,640 | 0,750 | 64 |
| fw | 0,000 | 0,000 | 0,000 | 14 |
| wp$ | 1,000 | 1,000 | 1,000 | 16 |
| # | 1,000 | 0,970 | 0,980 | 31 |
| uh | 0,000 | 0,000 | 0,000 | 13 |
| sym | 0,000 | 0,000 | 0,000 | 4 |
| ls | 0,000 | 0,000 | 0,000 | 2 |
| macro\_avg | 0,860 | 0,840 | 0,840 | 552700 |
| weighted\_avg | 0,990 | 0,990 | 0,990 | 552700 |

Com acurácia total de 0.9853 e F1-Score macro e ponderado de 0,840 e 0,990.

## Segundo Modelo

Utilizando o Flair com embeddings pré-treinados do modelo FastText em conjunto de embeddings de caracteres e de BytePair

### Pré-processamento

Os dados são transformados no formato específico que é aceito pelo Flair e escritos em três arquivos: treino, validação e teste. Em seguida os três arquivos são lidos em um objeto Corpus, específico da biblioteca, assim não se faz necessário realizar manualmente os passos de tokenização das sentenças, apenas indicar qual o tipo de tag a ser utilizada, no caso, as tags de Part-of-Speech obtidas do Penn Treebank.

### Embeddings

Os embeddings escolhidos são fornecidos pela própria biblioteca, concentrando diversos embeddings de outros trabalhos, e facilitando a combinação entre eles por meio do que chamam de StackedEmbeddings, assim, foram escolhidos os seguintes embeddings:

[Character Embeddings](https://github.com/flairNLP/flair/blob/master/resources/docs/embeddings/CHARACTER_EMBEDDINGS.md), [FastText Embeddings](https://github.com/flairNLP/flair/blob/master/resources/docs/embeddings/FASTTEXT_EMBEDDINGS.md) e [BytePair Embeddings](https://github.com/flairNLP/flair/blob/master/resources/docs/embeddings/BYTE_PAIR_EMBEDDINGS.md).

### Arquitetura

A arquitetura utilizada foi a padrão sugerida pela biblioteca, iniciando com a camada de embeddings criada, seguida por uma camada de word dropout com taxa de 0.05 e outra camada de dropout fixo com 0.5 de taxa. Então é colocada uma simples camada linear seguida da camada de LSTM bidirecional com 256 unidades, por fim, como última etapa, temos outra camada Linear para fornecer o resultado final do PoS tagging.

### Treinamento

Como parâmetros de treinamento, foi utilizado um learning rate de 0.1 por 5 epochs e com batchs de 32. Para reduzir o tempo de treinamento foi utilizada a GPU do Colab, totalizando 1 hora 10 minutos e 43 segundos de treino.

### Resultados obtidos no conjunto de teste

| **tag** | **precision** | **recall** | **f1-score** | **support** |
| --- | --- | --- | --- | --- |
| NN | 0,9153 | 0,9597 | 0,9370 | 17834 |
| IN | 0,9598 | 0,9648 | 0,9623 | 13492 |
| NNP | 0,9608 | 0,9609 | 0,9608 | 13177 |
| DT | 0,9777 | 0,9868 | 0,9822 | 11183 |
| NNS | 0,9749 | 0,9730 | 0,9739 | 8061 |
| JJ | 0,8646 | 0,7658 | 0,8122 | 8215 |
| , | 1,0000 | 0,9996 | 0,9998 | 6876 |
| , | 0,9983 | 0,9965 | 0,9974 | 5381 |
| CD | 0,9764 | 0,9847 | 0,9805 | 4823 |
| RB | 0,7435 | 0,8754 | 0,8041 | 4405 |
| VBD | 0,9735 | 0,9029 | 0,9369 | 4561 |
| VB | 0,9320 | 0,9479 | 0,9399 | 3573 |
| CC | 0,9837 | 0,9846 | 0,9842 | 3250 |
| VBN | 0,7394 | 0,9184 | 0,8192 | 2707 |
| TO | 0,9983 | 0,9997 | 0,9990 | 2913 |
| VBZ | 0,9877 | 0,9731 | 0,9803 | 2639 |
| PRP | 0,9647 | 0,9782 | 0,9714 | 2205 |
| VBG | 0,9426 | 0,8158 | 0,8747 | 1933 |
| VBP | 0,9293 | 0,8735 | 0,9005 | 1565 |
| POS | 0,9768 | 0,9890 | 0,9829 | 1276 |
| MD | 0,9832 | 0,9716 | 0,9774 | 1267 |
| `` | 0,9963 | 1,0000 | 0,9981 | 1074 |
| PRP$ | 0,9906 | 0,9916 | 0,9911 | 1068 |
| '' | 0,9924 | 0,9971 | 0,9947 | 1045 |
| $ | 0,9989 | 0,9968 | 0,9979 | 943 |
| : | 0,9854 | 0,9854 | 0,9854 | 752 |
| WDT | 0,9185 | 0,9264 | 0,9224 | 584 |
| JJR | 0,6511 | 0,4941 | 0,5618 | 423 |
| RP | 0,8500 | 0,5139 | 0,6405 | 397 |
| WP | 0,9854 | 0,7173 | 0,8303 | 283 |
| WRB | 0,9245 | 0,4836 | 0,6350 | 304 |
| RBR | 0,6205 | 0,3801 | 0,4714 | 271 |
| -RRB- | 0,8462 | 0,8824 | 0,8639 | 187 |
| -LRB- | 0,8778 | 0,8495 | 0,8634 | 186 |
| NNPS | 0,2116 | 0,2353 | 0,2228 | 170 |
| JJS | 0,0000 | 0,0000 | 0,0000 | 267 |
| EX | 0,9065 | 1,0000 | 0,9509 | 126 |
| RBS | 0,9200 | 0,6667 | 0,7731 | 69 |
| WP$ | 1,0000 | 0,7297 | 0,8437 | 37 |
| PDT | 0,7500 | 0,1364 | 0,2308 | 44 |
| FW | 0,0000 | 0,0000 | 0,0000 | 30 |
| # | 1,0000 | 0,9333 | 0,9655 | 15 |
| LS | 1,0000 | 0,6000 | 0,7500 | 15 |
| SYM | 1,0000 | 0,8182 | 0,9000 | 11 |
| UH | 1,0000 | 0,1176 | 0,2105 | 17 |
| macro\_avg | 0,8713 | 0,7839 | 0,8084 | 129654 |
| weighted\_avg | 0,9396 | 0,9404 | 0,9387 | 129654 |

Com acurácia total de 0.9404 e F1-Score macro e ponderado de 0.8084 e 0.9387.

## Discussões dos Resultados

Infelizmente ocorreram problemas na execução do modelo 1 no conjunto de teste, invalidando a sua análise mais detalhada, pois as tags preditas foram praticamente apenas o pad, que nem sequer deveria ser considerado uma tag para o modelo, porém devido a escolhas de pré-processamento e modelagem do problema, ocorreu esse bug. Já no conjunto de validação as predições das tags de cada sentença ocorreram de forma mais aceitável, mesmo sendo percebido um overfitting ao comparar com o tagging obtido durante o treino.

O modelo 2 obteve bons resultados em geral, porém houve muito menos liberdade para configuração do mesmo, sendo que a arquitetura praticamente não foi modificada, sendo utilizada a padrão da biblioteca Flair, apenas a camada de embeddings que permite uma maior personalização. O tempo de treinamento foi muito mais longo do que o do modelo 1, enquanto o modelo 1 treinou em menos de 10 minutos, o segundo modelo levou mais de 1 hora para completar o treinamento, sendo que ambos foram executados durante o mesmo número de épocas e com o mesmo tamanho de batch.

Com relação ao resultado do modelo 2, as tags com piores métricas foram: FW, JJS, NNPS, RBR, UH e PDT, todas com F1-Score menor que 0.5, em seguida temos as tags: LS, WP$, RBS, WRB, WP, RP, JJR, VBN, RB e JJ, todas com F1-Score maior que 0.5 e menor que 0.85, enquanto que as demais tags tiveram um F1-Score acima de 0.85, muitas próximas de 0.95.

## Considerações Finais

Foram apresentadas duas abordagens para o problema de PoS Tagging utilizando de Redes Neurais Recorrentes (RNR), mais especificamente com o uso de camadas Long Short-Term Memory Bidirecionais (LSTM Bidirecional). Foi brevemente explicitado como o pré-processamento em cada modelo foi efetuado, quais embeddings foram utilizados, a organização da arquitetura de cada modelo, comentando quais camadas foram utilizadas, quais parâmetros foram aplicados para efetuar o treinamento, e por fim, quais os resultados obtidos no conjunto de teste.

O modelo 1 apresentou problemas ao ser avaliado no conjunto de teste, onde praticamente apenas o pad estava sendo classificado, o que pode ser devido a forma como os dados foram modelados e pré-processados, pois o pad acabou sendo considerado como uma tag para o modelo, o que é incorreto, esse comportamento acaba invalidando outras análises no modelo 1, porém o mesmo não ocorreu no conjunto de validação, onde mesmo sendo percebido um overfitting ao comparar com os resultados do conjunto de treino, ainda houveram tags diferentes do pad sendo atribuídas de forma considerável.

O modelo 2 apresentou bons resultados, porém a custa de um longo tempo de execução e pouca liberdade de configuração da arquitetura e parâmetros do modelo, porém essas mesmas restrições na liberdade garantiram uma execução sem erros de pré-processamento e/ou modelagem, pois o pad não foi incluído como tag, e a etapa de escolha e configuração dos embeddings foi grandemente facilitada, permitindo a composição de 3 tipos de embedding sem dificuldades.

Os notebooks utilizados podem ser acessados aqui: [LSTM](https://drive.google.com/file/d/1IUBW7p1_x354TGrd2T9XWv6Xk0PKe4P6/view?usp=sharing) e [Flair](https://www.kaggle.com/johnvithor/pln-u3-flair/notebook)

## Referências

Tutorial sobre RNNs para PoS Tagging no TensorFlow:

* <https://towardsdatascience.com/pos-tagging-using-rnn-7f08a522f849>

Tutoriais sobre o uso do Flair e seu funcionamento:

* <https://github.com/flairNLP/flair/tree/master/resources/docs>

TensorFlowHub e os diversos modelos de embeddings disponibilizados:

* <https://tfhub.dev/s?module-type=text-embedding>

Capítulo 6 do livro Speech and Language Processing de Daniel Jurafsky & James H. Martin:

* <https://web.stanford.edu/~jurafsky/slp3/6.pdf>