

# Relatório EP 2

Flávio Nakasato Cação

7373076

## a) Sobre as configurações e os hiperparâmetros do modelo.

Como base de dados, foi utilizado o conjunto 'b2w-10k.csv' fornecido pela B2W, com 10 mil registros. Esta base foi embaralhada e dividida em três grupos: treinamento, validação e teste (este último utilizado apenas no final), nas proporções 65%, 10% e 25%, respectivamente. As curvas, losses e acurácias mostradas a seguir foram geradas sobre o conjunto de validação. Para os *embeddings*, tomamos o de 50 dimensões 'cbow\_s50.txt' do projeto NILC. Limitamos o tamanho do vocabulário a 2000.

Foi realizado um *Gridsearch* sobre 4 parâmetros, com 70 épocas por rodada, nas variações abaixo:

- **Rede:** Unidirecional e Bidirecional
- **Dropout:** 0.00, 0.25 e 0.50
- **Tammax** (para o tamanho das palavras iniciais): 50 e 300
- **Tam** (para o tamanho do batch): 32 e 256

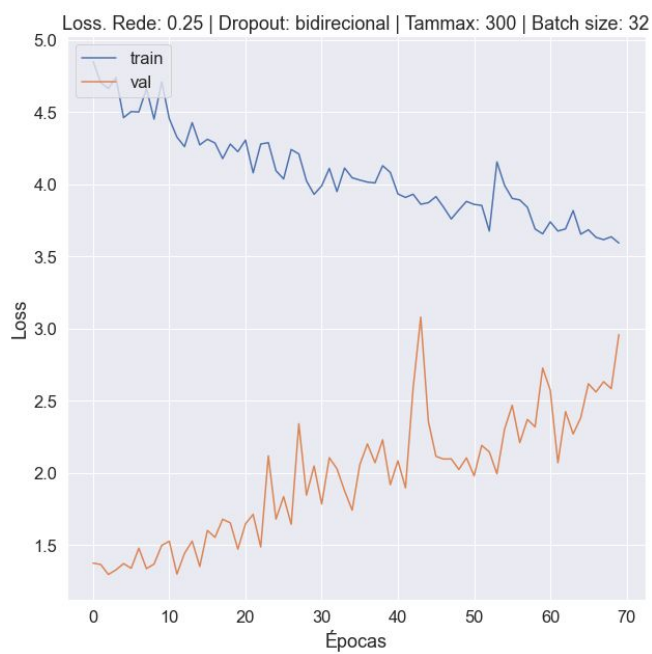
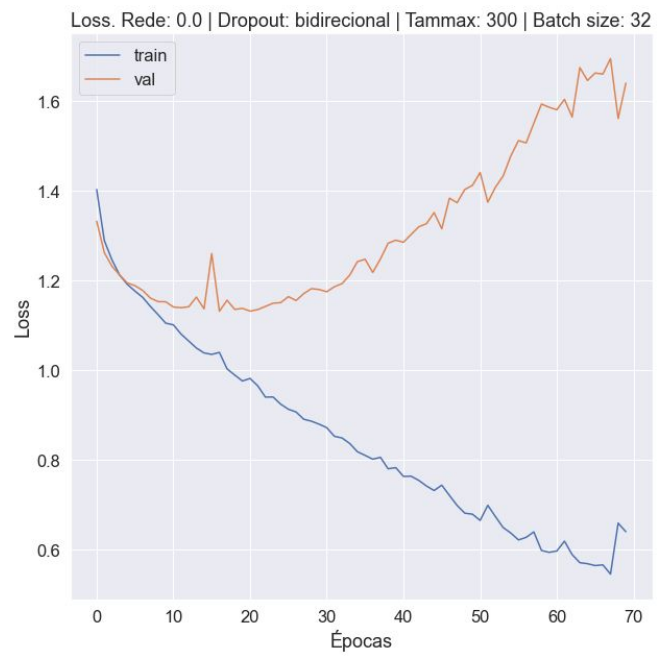
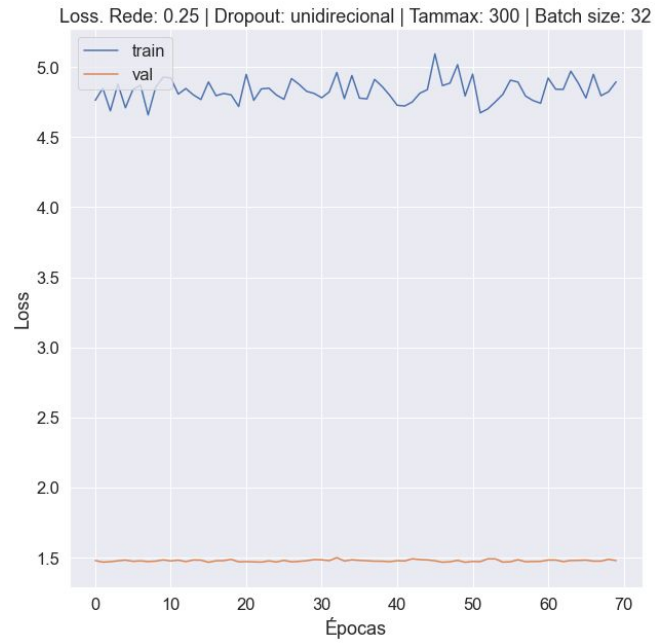
Em todos os casos usamos 32 unidades de LSTMs, otimizadores *Adam*, *loss* 'sparse\_categorical\_crossentropy' e a métrica-alvo foi a acurácia.

## b) Gráficos de validação e épocas até sobreajuste.

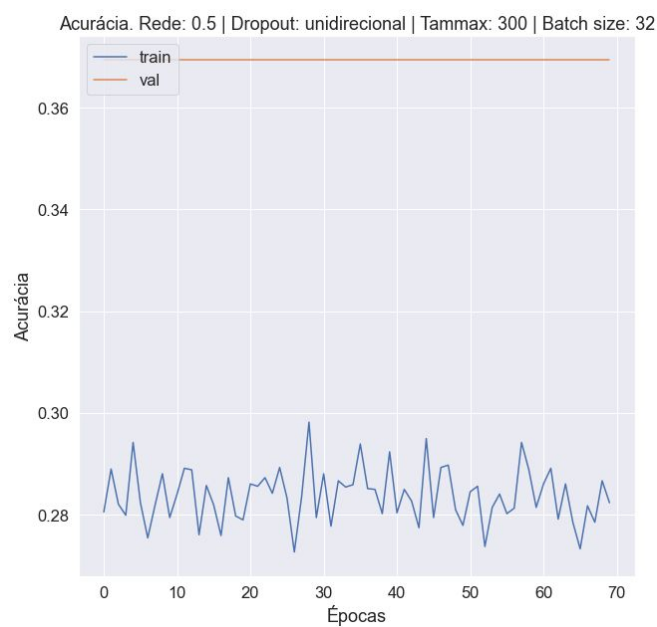
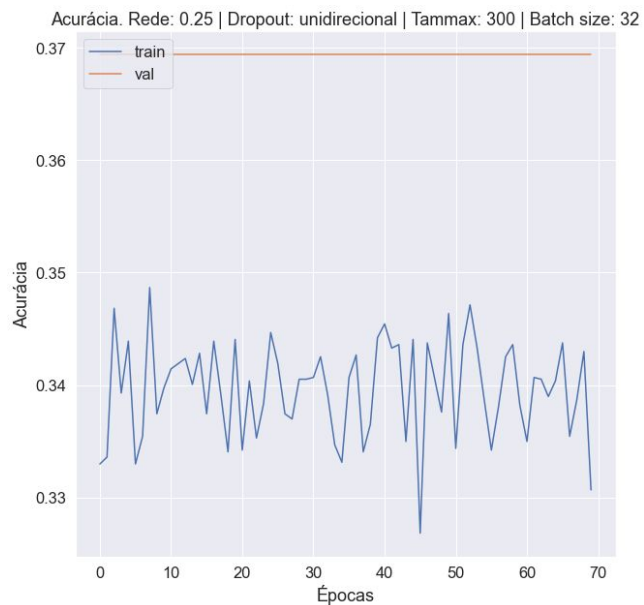
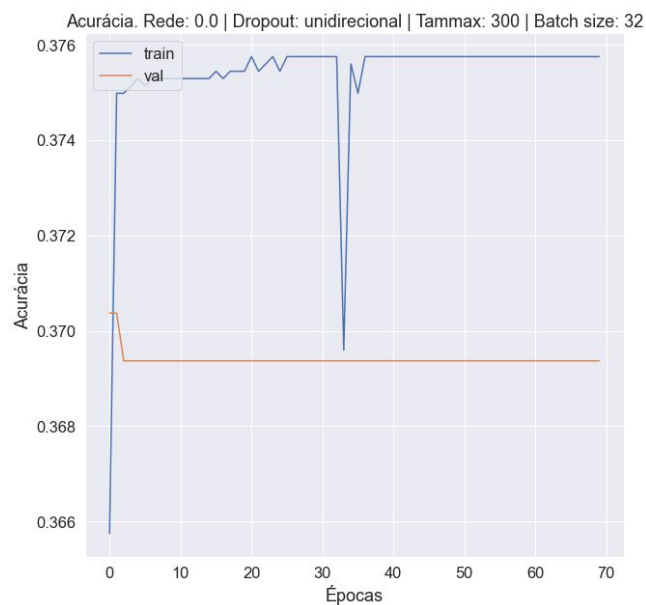
Gráficos de *loss* e acurácia abaixo gerados sobre o conjunto de validação, com **tam=300** e **tammax=32**, que se mostraram um pouco melhores na etapa de *Gridsearch*. Nota: há typo de inversão nos valores "Rede" e "Dropout" nos títulos dos gráficos; eles são consistentes com a curva, apenas estão trocados.

Foram tomadas 70 épocas no total para cada curva, mas pode-se observar que nos melhores casos abaixo, começa a haver sobreajuste a partir de **~17 épocas aproximadamente** (melhor modelo: Rede: bidirecional | Dropout: 0.0 | Tammax: 300 | Batch size: 32). Na maioria dos modelos, principalmente os unidirecionais, não parece ter havido aprendizagem real, com as curvas de validação estáticas praticamente.

## Losses



## Acurácias



### c) Tabela com acurácias no teste.

Tabela com as acurácias no conjunto de teste após 17 épocas, com tam=300 e tammax=32. Em negrito, o melhor modelo.

Rede	Dropout	Acurácia no teste
Unidirecional	0.00	0.376
	0.25	0.376
	0.50	0.375
<b>Bidirecional</b>	<b>0.00</b>	<b>0.510</b>
	0.25	0.500
	0.50	0.300

Os resultados acima foram obtidos rodando as avaliações dos modelos no conjunto de teste imediatamente após o treinamento. Quando salvamos todos os parâmetros do modelo usando os métodos do Keras (`modelo.save('modelo_final')`) e depois o carregamos (`tf.keras.models.load_model('modelo_final')`), mesmo mantendo sementes no Numpy e no Tensorflow, os resultados caem para cerca 0.33, no mesmo conjunto de teste. Ainda não foi compreendido disto, mas cremos que se deva a alguma particularidade computacional da biblioteca ou IDE usadas, então reportamos os valores acima, sobre o mesmo conjunto.

### d) Discussão.

Pudemos observar que redes unidirecionais têm considerável dificuldade em aprender sob as nossas especificações de problema; seus desempenhos são muitos baixos e assim se mantêm sob diversas combinações de hiperparâmetros.

Obtivemos, no entanto, um resultado substancialmente mais elevado usando redes bidirecionais, como seria esperado. É importante notar que a camada de dropout não trouxe a este problema grandes melhorias de acurácia e consideramos o melhor modelo - sob pequena margem, diga-se - aquele cuja taxa de dropout é nula (em negrito na tabela da seção anterior).

Um efeito interessante do Dropout é que as curvas de validação em geral são melhores (menores erros, maiores acurácias) do que aquelas obtidas no treino. Nossa hipótese é que isso ocorre porque, no último caso, a rede está, por construção, com diversos neurônios desativados; isto é, estamos tornando artificialmente mais difícil para a rede a tarefa de prever. Na validação, no entanto, a rede usufrui de toda a sua capacidade para a mesma tarefa, resultando no observado desempenho superior.