Deep Learning for Automatic Image Captioning

Maynara N. Scoparo, Adriane B. S. Serapião

[1]Universidade Estadual Paulista (UNESP)
Rio Claro – SP – Brasil

maynara.scoparo@unesp.br, adriane.serapiao@unesp.br

Abstract. *Automatic image caption generation is a task that consists of deciphering an image and describing it in natural language sentences. It combines Natural Language Processing and Computer Vision to generate captions. Recently, Deep Learning methods are achieving very promising results for the captions generation problem. This present work proposed, based on the NIC (Neural Image Captioning) model, a combination of convolutional neural networks over images and recurrent neural network over sentences, aligning them with a structured objective of creating the textual description of images. The results have shown that the proposed neural model was able to learn the language model upon image content, producing accurated descriptions in most images.*

1. Introdução

Algumas das ações que requerem habilidades cognitivas humanas, como classificar e descrever imagens, reconhecer fala, detectar objetos são extremamente complicadas de serem replicadas por máquinas. Hoje, a área de Inteligência Artificial investiga formas de viabilizar a reprodução dessas características intrinsecamente humanas. Um dos principais desafios da área é geração automática de legendas de imagens (*image captioning* ou *image labeling*, em inglês), que consiste na descrição textual de imagens que seja compreensível por humanos e seja coerente com as ações, objetos e cenas apresentadas na imagem.

A importância de se desenvolver modelos capazes de gerar descrição automática de imagens é evidente e diversos motivos fundamentam esse fato. Por exemplo, as descrições das imagens podem ser utilizadas para a tarefa de indexação automática de imagens. A indexação de imagens é importante para a recuperação de imagens baseada em conteúdo (em inglês, *Content-Based Image Retrieval* (CBIR)) e, portanto, pode ser aplicada a muitas áreas, incluindo biomedicina, comércio, educação, bibliotecas e pesquisa na web. Pode-se citar também o uso da tarefa em plataformas de mídias sociais, com o intuito de inferir, a partir da imagem, onde o usuário está (praia, café etc) [Hossain et al. 2019]. Outro exemplo seria produzir explicações sobre o que acontece em um vídeo, quadro a quadro, já que um quadro é uma imagem estática, indicando cada cena, o que poderia ser um grande auxílio para pessoas com deficiência visual.

O processo de gerar legendas de imagens requer um entendimento visual (extração de características da imagem) e o processamento de linguagem (gerar descrições plausíveis, sintaticamente e semanticamente corretas). Na literatura há diversos modelos propostos para gerar descrição de imagens. Entretanto, devido à natureza complexa do processo e a necessidade de lidar com um volume extenso de informações, muitas abordagens limitam a expressividade das legendas. Por isso, abordagens com o uso de técnicas

de *Deep Learning* (DL) [Goodfellow et al. 2016] (que permitem a manipulação de volumes extensos de dados), mais especificamente técnicas utilizadas na área de Computação visual e Processamento de Linguagem Natural, têm se mostrado eficientes para a tarefa e trouxeram modelos que são estado-da-arte [Bai and An 2018].

Esses modelos, considerados estado da arte em geração de legendas de imagens, possuem de forma superficial, dois aspectos principais: uma arquitetura de rede neural capaz de extrair características da imagem e outra que seja capaz de processar linguagem para gerar legendas. Para o primeiro aspecto prevê-se o uso de uma rede neural convolucional (CNN, do inglês *Convolutional Neural Networks*), que simula o funcionamento do córtex humano [LeCun et al. 1998], de forma a tornar possível o reconhecimento de imagens por parte do computador. Já para o segundo aspecto, é necessário o uso de redes neurais recorrentes (RNN, do inglês *Recurrent Neural Network*) [Bengio et al. 1993], que é capaz de prever a sequência de palavras na legenda baseadas na descrição já gerada.

As CNNs são utilizadas como subrede para extrair as características das imagens que alimentarão o modelo de treinamento. Muitos modelos pré-treinados em classificação de imagens podem ser usados, assim com modelos híbridos, onde o modelo pré-treinado pode ser ajustado ao problema. Para o modelo de linguagem, as RNN permitem a recorrência das entradas, de forma que palavras já geradas na sequência possam realimentar a entrada para prever as próximas palavras. Nesse último caso, as memórias longas de curto prazo (LSTMs, do inglês *Long Short-Term Memory*) [Hochreiter and Schmidhuber 1997] têm se mostrado mais eficientes, pois possuem um conjunto de portas (*gates*) que permitem manter informações mais tempo em 'memória', o que é fundamental na geração adequada das descrições.

No presente trabalho foi desenvolvido um modelo de geração de descrição automática de imagens baseado na proposta de [Vinyals et al. 2015], na qual é descrito um modelo de rede neural *encoder-decoder* (traduzido "codificador-decodificador"), em que o *encoder* é uma CNN que codifica uma imagem para um vetor de tamanho fixo e depois passa para o *decoder*, que é uma RNN, realizar a decodificação para uma descrição em linguagem natural. Nesse arquitetura, a saída da última camada escondida da CNN é utilizada como a entrada da RNN.

O artigo está dividido da seguinte forma. A Seção 2 apresenta os principais trabalhos relacionados ao tema. Na Seção 3 é abordada a metodologia de desenvolvimento da proposta, considerando o pré-processamento dos dados, a construção do modelo, a forma como a inferência de legendas é realizada, os parâmetros adotados no treinamento e as métricas utilizadas para avaliação. Na Seção 4 são apresentados os resultados obtidos e, por fim, na Seção 5 apresentam-se as conclusões do trabalho.

2. Trabalhos correlatos

As abordagens com técnicas de DL dominaram a área de geração de legendas. O método de [Vinyals et al. 2015] e de [Karpathy and Fei-Fei 2015] usam a combinação de CNN com RNN, porém, de formas diferentes. No primeiro, a imagem passa por um processo de extração de características através da CNN, retornando a saída da última camada, que é um vetor de tamanho fixo. Já no segundo, é utilizada uma CNN que detecta objetos na imagem e representa-os como um conjunto de vetores. Então, para modelar a relação com a imagem, as palavras são representadas como vetores de mesmas dimensões ocupadas

pelas regiões da imagem através de uma RNN. Para preservar o contexto da palavra, sua posição no vetor representa sua posição dentro da sentença. Tais modelos apenas introduzem características de imagem no primeiro passo de tempo e não exploram uma representação "fatorada". Em contraste, há o modelo proposto por [Donahue et al. 2015], que estabelece a inserção de características de imagens a cada passo de tempo, de forma que os dados visuais que alimentam unidades LSTM podem variar com o tempo.

No método proposto por [Xu et al. 2015], a descrição do conteúdo da imagem tem por objetivo dar foco a partes da imagem durante o processo de gerar a legenda. Para isso, é utilizada uma CNN para extrair os vetores de características da imagem. Contudo, ao invés de uma arquitetura com uma camada totalmente conectada ao fim, foi utilizada a ideia de manter uma camada de convolução para extrair os vetores correspondentes a cada parte da imagem. Dessa forma, o decodificador (LSTM) pode selecionar um subconjunto de características da imagem, permitindo que se alcance o objetivo de dar atenção apenas em partes da imagem por vez.

Outros trabalhos visam abordar limitações específicas de modelos de legendagem baseados na combinação de arquiteturas convolucionais e recorrentes. [Hendricks et al. 2016] propôs um modelo que não se baseia apenas nos pares imagem-sentença contidos no conjunto de treinamento para gerar legendas. Essa habilidade é adquirida devido a um classificador lexical existente no modelo, que permite um treinamento totalmente independente em dados de imagens e dados meramente textuais, dessa forma é possível descrever novos objetos em contextos distintos. Já o modelo proposto por [Mao et al. 2015] visa introduzir a capacidade de a rede neural adicionar o significado semântico de novas palavras ao seu vocabulário, para que sejam usadas posteriormente na descrição de imagens.

O presente trabalho teve por objetivo incorporar as ideias propostas por [Vinyals et al. 2015] para construção do modelo de descrição de legendas, onde as imagens têm suas características extraídas na forma de um vetor de tamanho fixo por uma CNN. Assim, essas características alimentam um modelo *decoder* (RNN) juntamente com a informação textual gerada a cada passo de tempo, esperando-se que a saída seja a próxima palavra da sentença.

3. Arquitetura do modelo proposto

O modelo aqui usado baseou-se no modelo NIC (*Neural Image Captioning*), proposto por [Vinyals et al. 2015], no qual as características da imagem são extraídas por um modelo pré-treinado e inseridas no modelo de geração automática de legendas em um primeiro passo. Assim, os dados visuais são processados com os dados textuais por LSTMs, com o intento de gerar a descrição em linguagem natural para a imagem.

O modelo NIC usa para a CNN o modelo pré-treinado Inception V3 [Szegedy et al. 2016]. Essa arquitetura tem como principal característica convoluções em paralelo, com o propósito de sanar o problema de decidir entre qual configuração de convolução é melhor para cada situação. Por exemplo, pode-se colocar convoluções de 3×3 e 5×5 em paralelo em um nível da rede neural e fazer com que o modelo desenvolvido decida pela melhor. O modelo de CNN usado aqui é a VGGNet [Simonyan and Zisserman 2014], criado pelo Visual Geometry Group (VGG) da Universidade de Oxford, pré-treinada no conjunto ImageNet.

Para processamento das legendas, foi preciso aplicar uma RNN capaz de manter em "memória" dados processados nos passos anteriores. Isso é fundamental, visto que a natureza do problema requer o uso das palavras já geradas para gerar a próxima. Todavia, um aprimoramento da RNN, a LSTM, proposta por [Hochreiter and Schmidhuber 1997], fornece a opção de manter informação em memória por mais tempo e, por isso, foi adotada no trabalho. Além disso, é capaz de resolver problemas de gradientes explosivos e gradientes que se anulam, encontrados no uso de RNN.

Na presente proposta utilizou-se o *merge model*, proposto por [Tanti et al. 2017], no qual há uma combinação entre a imagem codificada e a descrição de texto gerada até o momento para gerar a próxima palavra da sequência. Dessa maneira, pretende-se que a LSTM lide com informações puramente textuais. O modelo *inject model*, eleito no NIC, considera que as informações visuais (vetores de características das imagens) são "injetados" nas LSTMs. Assim, é codificado um vetor com informações visuais e textuais com o porpósito de prever as próximas palavras da sequência. Esses dois pontos sobre a CNN e sobre a RNN são as principais diferenças de nossa abordagem com o modelo NIC.

A Figura 1 ilustra o modelo desenvolvido no presente trabalho. O modelo textual, que recebe as legendas vetorizadas como entrada, possui uma camada de *embedding*, a qual é utilizada para criar modelo de incorporação de palavras, onde as palavras do vocabulário dos dados textuais são codificadas para vetores de valores contínuos. Essa camada pode ser usada para carregar modelos pré-treinados, mas aqui foi utilizada como parte do modelo de geração automática de legendas com inicialização aleatória dos pesos. A entrada da camada é o número de palavras no vocabulário; a saída foi estabelecida com o valor de 256, que define o tamanho do vetor que representa cada palavra. Por último, foi necessário estabelecer o tamanho da entrada, no nosso caso, é o comprimento da maior legenda encontrada nas legendas de referência. Além disso, tanto no modelo textual como no modelo visual é aplicada uma camada de *dropout* de 50%, com a intenção de reduzir o *overfitting*. As informações textuais são processadas por uma LSTM e as imagens por uma camada densa. Logo após, as informações são agrupadas em uma camada de *merge* e então processadas por camadas densas. Nas camadas densas foi utilizada a função de ativação *softmax* na saída e a ReLU nas demais. A saída é um vetor com o tamanho do vocabulário, em que cada posição é a probabilidade de uma determinada palavra ser a próxima do vocabulário. Nas camadas de ativação também fora aplicadas a regularização L2, de forma a prevenir *overfitting*.

4. Metodologia

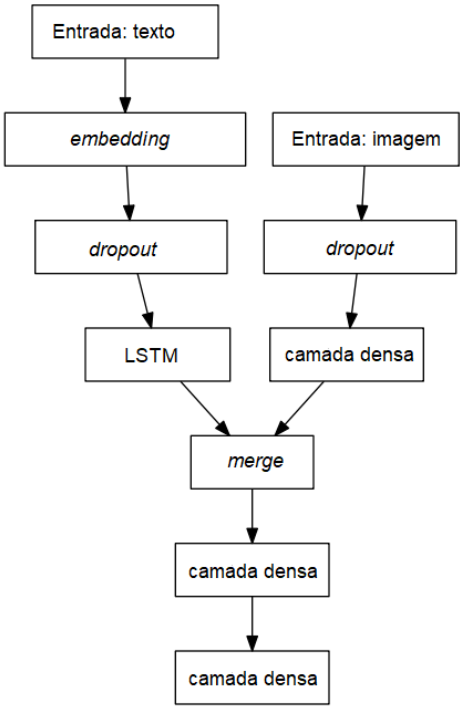
Nesta seção apresenta-se a metodologia utilizada para desenvolvimento de um modelo de descrição automática de imagens, o que inclui o modelo de rede neural construída, a forma que foi estabelecido o treinamento da mesma, a estratégia abordada para inferir as legendas, bem como as métricas usadas para avaliação.

4.1. Conjunto de dados

Para a avaliação do modelo foram adotados os seguintes conjuntos de imagens:

- Flickr8K: consiste em uma coleção de 8 mil imagens descritas. extraídas do flickr.com [Hodosh et al. 2013]. Esse conjunto foi dividido em 6000 imagens no conjunto de treino, 1000 imagens no conjunto de validação e 1000 imagens no conjunto de teste. Além disso, cada imagem possui 5 legendas de referência.

Figura 1. Arquitetura do modelo proposto para geração de legendas



- Flickr30K: é uma extensão do conjunto anterior, contendo uma coleção de 31381 imagens descrita, também extraídas do flickr.com. Esse conjunto foi dividido em: 25381 imagens no conjunto de treino, 3000 imagens no conjunto de validação e 3000 imagens no conjunto de teste. Cada imagem possui 5 legendas de referência.

4.2. Treinamento do modelo

Para o treinamento é necessário realizar o pré-processamento dos dados e organizá-los da forma como é requerido pelo modelo de rede neural construído. Do mesmo modo, é importante estabelecer de que forma esses dados alimentam a rede e definir quais parâmetros serão usados no processo de treinamento, como número de épocas, penalizações etc.

4.2.1. Pré-processamento

Visando preparar os dados corretamente para alimentar a rede neural, os dados visuais e textuais foram pré-processados. Primeiramente, foi preciso redimensioná-los para o tamanho de 256×256, como é requerido pela VGGNet. Dessa forma, a última camada de ativação da CNN foi extraída de maneira que a saída fosse apenas as características da imagem, sem a classificação. Assim, a saída passou a ser vetores fixos com 4096 posições contendo valores numéricos, que representam justamente as características da imagem, necessárias para o treinamento do modelo de geração automática de legendas.

Em seguida, os dados textuais passaram por um processo de "limpeza", de forma que pontuações, números, símbolos foram retirados, assim como todas letras foram transformadas em minúsculas e palavras formadas por caracteres únicos também foram eliminadas. Como consequência, obteve-se um vocabulário com 7579 palavras.

4.2.2. Gerador de dados

Na fase de treinamento é necessário estruturar os dados da forma adequada para treinar a rede neural. A abordagem utilizada no trabalho foi produzir vetores como mostra a Tabela 1. A ideia é que a rede seja alimentada com as características da imagem (X_{img}) combinadas com a legenda gerada em determinado passo de tempo (X_{text}), esperando-se que a saída seja a próxima palavra da sequência (Y_{text}). O *token* ‘ $< start >$ ‘ sinaliza o início, o fim é sinalizado por ‘ $< end >$ ‘ ou se o tamanho máximo da legenda é atingido.

Tabela 1. Exemplificação da estrutura de dados utilizada como entrada da rede neural

Passo de tempo	X_{img}	X_{text}	Y_{text}
0	características da imagem	<start>	young
1	características da imagem	<start>young	boy
2	características da imagem	<start>young boy	runs
3	características da imagem	<start>young boy runs	across
4	características da imagem	<start>young boy runs across	the
5	características da imagem	<start>young boy runs across the	street
6	características da imagem	<start>young boy runs across the street	<end>

Foi utilizado um gerador de dados para estruturar corretamente os dados para o treinamento. Assim, a cada passo de tempo, a rede é alimentada por uma imagem e suas respectivas descrições. Portanto, cada época é dada por n passos, sendo n a quantidade de imagens no conjunto de treinamento.

4.3. Inferência

As saídas obtidas no processo de geração de legenda de uma imagem são distribuições de probabilidade sobre as palavras do vocabulário do conjunto de dados, portanto, é necessário que haja um algoritmo de decodificação que decida pela melhor palavra a cada passo de tempo, de forma a inferir a descrição que melhor se adequa à imagem de entrada. Existem diversos métodos possíveis, mas no presente trabalho foi adotado o método *Beam Search*, que é uma heurística capaz de estruturar as frases através do reconhecimento do conteúdo mais relevante e é frequentemente usada em sistemas de tradução.

O *Beam Search* possui um parâmetro, o *beam width*, que indica quantas palavras têm que ser avaliadas antes de chegar à conclusão da melhor resposta. Por exemplo, se o *beam width* for estabelecido como 3, no primeiro passo as 3 palavras com maior probabilidade são armazenadas e, então, nos próximos passos, são calculadas as probabilidades de cada uma das 3 palavras escolhidas anteriormente.

No modelo proposto no trabalho foi avaliado o uso da *Beam Search* com *beam width* iguais a 1, 3, 5 e 20. Com o valor 1, o *Beam Search* funciona da mesma maneira que o algoritmo *Greedy Search*, que significa, em essência, apenas decidir pela palavra com a maior probabilidade a cada passo de tempo.

4.4. Métricas de avaliação

Na avaliação do quão coerente as legendas são geradas é necessário a adoção de métricas de avaliação de saídas textuais, utilizadas no processamento de linguagem natural. As

métricas usadas foram BLEU (*BilLingual Evaluation Understudy*) [Papineni et al. 2002], METEOR (*Metric for machine Translation Evaluation*) [Denkowski and Lavie 2014], CIDEr (*Consensus-based Image Description Evaluation*) [Vedantam et al. 2015] e ROUGE-L (*Recall Oriented Understudy for Gisting Evaluation*) [Lin 2004].

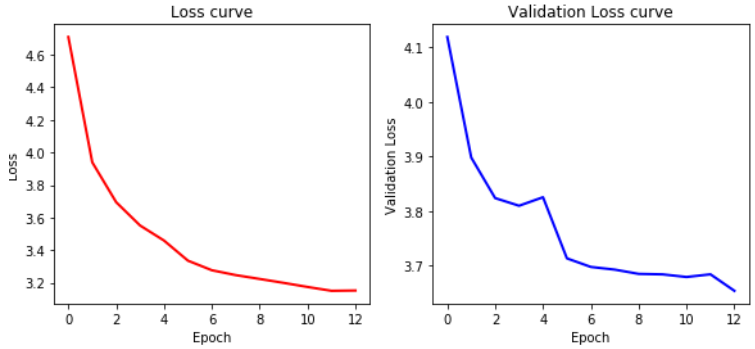
O critério utilizado pelas métricas para avaliação do resultados principia na ideia de comparar o quão próxima a descrição produzida se encontra da descrição de referência, baseado na precisão e revocação, assim como ocorre em validações de problemas de classificação. Dado que no Processamento de Linguagem Natural outros fatores devem ser levados em conta, como por exemplo, a coerência e significado do texto gerado, as métricas diferem em quais critérios são utilizados para computar a precisão e a revocação.

5. Resultados

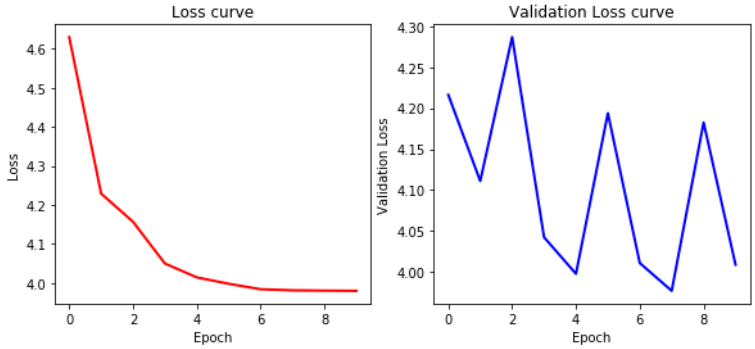
Na Figura 2 é possível visualizar a perda tanto no treinamento *(loss)* quanto no passo de validação *(validation loss)*, com os conjuntos de imagens a) Flickr8K e b) Flickr30K. Percebe-se a influência da inserção de penalidades de redução da taxa de aprendizado (fator 0,01). Cada época que não indicava melhoria no resultado era seguida de penalização. Na curva de *validation loss* é possível ver essa oscilação no decorrer das épocas. Para o cálculo das métricas foram considerados os pesos obtidos nas épocas que obtiveram menores valores de *validation loss*.

Figura 2. Curva de treinamento dos conjunto de imagens para (*loss* e *validation loss*)

a) Conjunto de imagens Flickr8K



b) Conjunto de imagens Flickr30K



Os resultados das métricas BLEU, METEOR, CIDEr e ROUGE-L foram computados para inferência com o *Beam Search*, com *beam width* igual a 1, 3, 5 e 20. Os resultados obtidos são apresentados na Tabela 2.

Tabela 2. Resultados obtidos com o modelo de geração automática de legendas desenvolvido com os conjuntos Flickr8k e Flickr30k

Conjunto	*Beam width*	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
Flickr8K	1	0,400	0,400	0,134	0,076	0,205	0,407	0,263
	3	0,528	0,308	0,194	0,118	0,223	0,456	0,327
	5	0,517	0,300	0,190	0,118	0,220	0,451	0,317
	20	0,449	0,236	0,147	0.090	0,200	0,409	0,212
Flickr30K	1	0,409	0,202	0,116	0,060	0,175	0,365	0,077
	3	0,443	0,217	0,125	0,066	0,179	0,382	0,090
	5	0,362	0,165	0,102	0,064	0,172	0,367	0,123
	20	0,333	0,120	0,060	0,026	0,147	0,314	0,055

Os resultados obtidos com *beam width* igual a 3 foram superiores em quase todas as métricas (exceto BLEU-2 para Flickr8K) aos resultados obtidos com o parâmetro igual a 1, 5 e 20. Esperava-se que o resultado pudesse ter ganhos pelo aumento no parâmetro *beam width*, considerando que mais palavras são avaliadas antes de se atingir o resultado final. Apesar disso, o aumento do *beam width* para 5 e 20 não resultou em valores mais satisfatórios do que aqueles obtidos com o parâmetro igual a 3. Esse comportamento foi observado no treinamento dos dois conjuntos.

Com o conjunto que contém quantidades menores de informações, o Flickr8K, é possível notar que os resultados foram satisfatórios, principalmente ao que se diz respeito aos valores obtidos com o *beam width* igual a 3, o qual apresentou os melhores resultados das métricas.

No trabalho de [Vinyals et al. 2015] foram obtidos valores de BLEU-1 iguais a 0,63 e 0,66 para os conjuntos Flickr8k e Flickr30k, respectivamente, utilizando *beam width* igual a 20 para inferência. Apesar dos resultados obtidos serem inferiores ao trabalho de referência, ainda não é suficiente analisar o desempenho do modelo apenas comparando valores de BLEU-1, considerando a deficiência da métrica em fornecer informações sobre a qualidade das descrições geradas. Por isso, para se ter uma visão mais detalhada das legendas resultantes, os valores computados de BLEU-1 foram separados nos seguintes intervalos: legendas com BLEU-1 $>$ 0,7; legendas com BLEU-1 entre 0,2 e 0,5; legendas com BLEU-1 entre 0,5 e 0,7 e legendas com BLEU-1 $<$ 0,2. As Figuras 3 e 4, representadas por gráficos de setores, apresentam a proporção de distribuição dos valores de BLEU-1 nos intervalos para as legendas produzidas para as imagens dos conjuntos de dados. Dessa forma, é possível analisar qualitativamente as legendas, sem apostar exclusivamente nos valores apontados pelas métricas.

Percebe-se que com ambos os conjuntos de dados, a maioria das legendas foram pontuadas entre 0,2 e 0,5. Os resultados mais satisfatórios, encontrados com BLEU-1 $>$ 0,7 e no intervalo entre 0,5 e 0,7 são encontrados com *beam width*=1, isso porque, apesar de ser comumente utilizada, a métrica BLEU-1 possui pontos fracos. Por exemplo, frases que possuem palavras repetidas que estão contidas na frase de referência são altamente pontuadas, porém, não deveriam, visto que a coerência da frase também deve ser levada em consideração. O uso das métricas indica uma ideia de desempenho do modelo pelas palavras escolhidas.

As métricas utilizadas diferem na sua forma de avaliação, como já citado anteriormente. Porém, não se mostram totalmente confiáveis para avaliar descrições, visto que

Figura 3. Distribuição dos resultados obtidos com o Flickr8K em cada intervalo estabelecido para a métrica BLEU-1

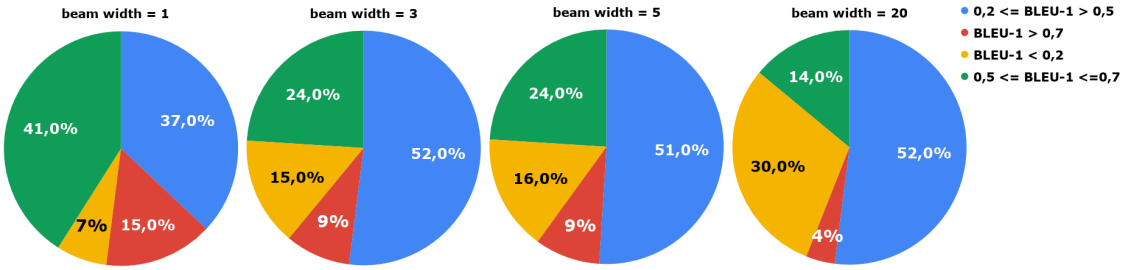
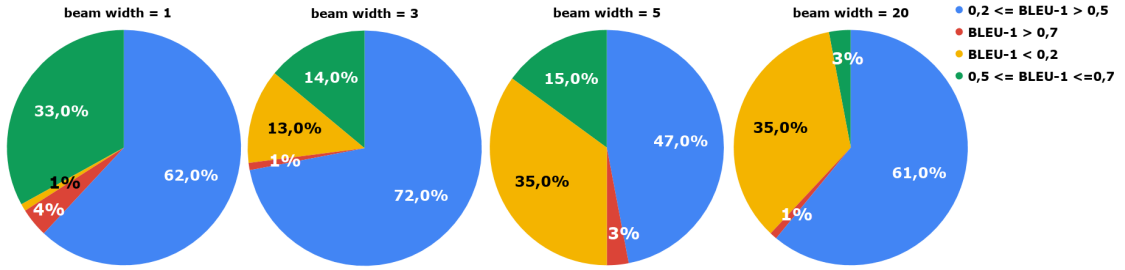


Figura 4. Distribuição dos resultados obtidos com o Flickr30K em cada intervalo estabelecido para a métrica BLEU-1

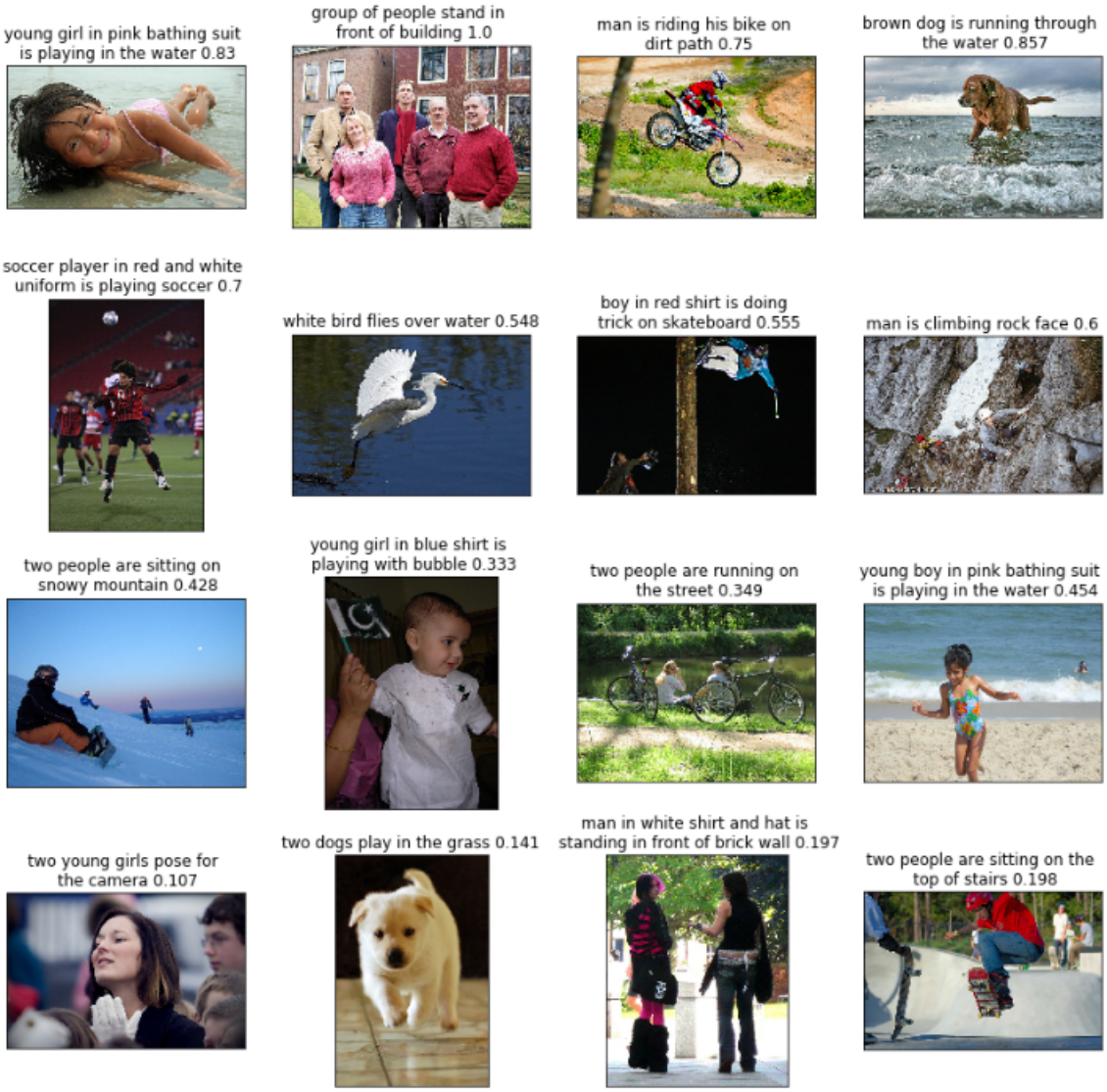


não conseguem validar de maneira precisa a semântica e a sintaxe das frases. Por isso, uma avaliação "humana" também tem sua importância. Ou seja, é relevante visualizar as legendas geradas para verificar se as descrições são plausíveis e podem ser consideradas como corretas. Embora as análises referentes às outras métricas não estejam demonstradas, resultados análogos a BLEU-1 foram encontrados.

Na Figura 5 são apresentados exemplos de legendas geradas pelo modelo proposto, utilizando o Flickr8k com *beam width* igual a 3, o qual obteve melhor desempenho. Em cada uma das imagens há a descrição gerada e seu respectivo valor de BLEU-1. É possível notar que algumas situações foram bem descritas pelo modelo, porém, outras obtiveram resultados insatisfatórios. Os melhores resultados, obtidos na primeira fileira horizontal de imagens na Figura 5, mostram que as legendas geradas foram bem coerentes com o contexto das cenas. Em casos como a primeira imagem, onde há uma garota deitada na água, percebe-se que o BLEU-1 foi alto, porém, não igual a um, mostrando que não foi coincidente a nenhuma legenda de referência. Logo, o modelo consegue gerar legendas novas, baseando-se no que foi aprendido em diferentes situações e ainda obter perfeita coerência com o que a imagem apresenta.

Na segunda fileira da Figura 5, os valores do BLEU-1 foram menores do que os apresentados na primeira, porém, há duas situações (segunda e quarta imagens) em que é nítido que a descrição foi perfeita. O defeito da métrica está justamente nesse aspecto, o fato das legendas diferirem do que há na referência faz com que sejam mal pontuadas. Entretanto, não é possível afirmar que a descrição foi incoerente. O modelo pode gerar uma descrição completamente nova e perfeitamente adequada à situação vista na imagem. Há situações totalmente incoerentes, como na última imagem da última da última fileira,

Figura 5. Algumas da legendas geradas pelo modelo com o conjunto Flickr8k e *beam width*=3 com seus respectivos valores da métrica BLEU-1



em que o homem praticando *skateboarding* foi referido como “duas pessoas sentadas no topo da escada”. Na terceira fileira há exemplos de imagens em que certos aspectos foram corretamente descritos, mas o modelo pecou em reconhecer o contexto corretamente.

Portanto, existem muitas situações que o modelo ainda não é capaz de assimilar. No geral, houve boas descrições. O que pode-se observar é que modelos de rede neurais tendem ao *overfitting* rapidamente, ou seja, tornam-se muito ajustadas aos dados de treinamento e, por isso, regularizações e normalizações são importantes. No caso da modelo em questão, é possível inferir que a rede aprendeu certas características muito bem, mas durante o processo de treinamento esqueceu algumas das características presentes nas imagens e decorou outras. Por isso, certas situações possuem 100% de acerto e outras nem chegam perto de uma descrição plausível. Com o conjunto Flickr30K foram obtidos resultados inferiores, porém, devido ao número maior de informações que o conjunto

contém, há potencial de se obter melhores resultados com alguns ajustes no modelo.

Com a análise das legendas é possível inferir que o uso das métricas, principalmente de BLEU-1, que não considera diversos fatores relevantes, não reconhece a capacidade do modelo de gerar legendas totalmente novas que são coerentes com a imagem, justamente porque utilizam exclusivamente as informações das legendas de referência. Por isso, o uso do METEOR, que considera sinônimos e palavras adjacentes, é considerado o mais próximo do julgamento humano, porém, ainda não substitui este último.

6. Conclusões

O presente trabalho buscou desenvolver um modelo de geração automática de descrição de imagens utilizando técnicas de DL, com o intuito de explorar o potencial da área. É notável que resultados bons foram obtidos, mas ainda há limitações no modelo que mostram falhas no aprendizado da rede neural profunda. O modelo resultou em legendas aceitáveis para a maioria das imagens e, apesar de haver erros quanto à predição das descrições, muitas não são totalmente incoerentes com o conteúdo da imagem, apresentando pequenos distorções que podem ser contornados com o ajuste do modelo.

Com os desempenho promissor obtido, existem algumas formas que podem refiná-lo, que devem ser exploradas em trabalhos futuros, como avaliar diversas formas de regularização, outros modelos pré-treinados de CNNs para extração das características das imagens, normalização de *batch* e uso de modelos de *embeddings* pré-treinados. Outros aspectos que podem ser investigado são a tentativa de melhorar a inicialização dos pesos, um melhor ajuste dos hiperparâmetros, ou até mesmo utilizar outras tamanhos de dimensão de *embeddings* e de LSTMs. Por último, pode-se investigar o uso de redes convolucionais que possuam camadas residuais (*ResNet*)

Agradecimentos

Os autores agradecem a Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), Processo n° 2018/09368-4.

Referências

Bai, S. and An, S. (2018). A survey on automatic image caption generation. *Neurocomputing*, 311:291–304.

Bengio, Y., Frasconi, P., and Simard, P. (1993). The problem of learning long-term dependencies in recurrent networks. In *IEEE international conference on neural networks*, pages 1183–1188. IEEE.

Denkowski, M. and Lavie, A. (2014). Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*, pages 376–380.

Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., and Darrell, T. (2015). Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2625–2634.

Goodfellow, I. J., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA, USA.

Hendricks, L. A., Venugopalan, S., Rohrbach, M., Mooney, R., Saenko, K., and Darrell, T. (2016). Deep compositional captioning: Describing novel object categories without paired training data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–10.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hodosh, M., Young, P., and Hockenmaier, J. (2013). Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899.

Hossain, M., Sohel, F., Shiratuddin, M. F., and Laga, H. (2019). A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CSUR)*, 51(6):118.

Karpathy, A. and Fei-Fei, L. (2015). Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., et al. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lin, C.-Y. (2004). Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Mao, J., Wei, X., Yang, Y., Wang, J., Huang, Z., and Yuille, A. L. (2015). Learning like a child: Fast novel visual concept learning from sentence descriptions of images. In *Proceedings of the IEEE international conference on computer vision*, pages 2533–2541.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. (2002). Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826.

Tanti, M., Gatt, A., and Camilleri, K. P. (2017). What is the role of recurrent neural networks (rnns) in an image caption generator? *arXiv preprint arXiv:1708.02043*.

Vedantam, R., Lawrence Zitnick, C., and Parikh, D. (2015). Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. (2015). Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhudinov, R., Zemel, R., and Bengio, Y. (2015). Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057.