Optimized Neural Network Story Generator

Cong Ye, David Wang
SUNet IDs: yecong, dwangsf

Introduction

The goal of this project is to combine visual and language processing by building an intelligent storyteller based on input images. We create an intelligent storyteller, by giving any image, it can try to understand the scenario and semantics from it and generate a meaningful story. We focus on entertainment purpose first, and the same model with the appropriate dataset and feature adjustments could also be used for early Childhood Education, medical science, and many other areas.

To tell good stories, the model should be able to understand the semantics of the image captions. There are many traditional ways to solve the problem with supervised learning. After comprehensive researching and experiments, we learned that RNN could be a potential unsupervised solution without feature engineering.

To achieve our goal, we eventually employ unsupervised learning of a generic, distributed sentence encoder. Then, we leverage the continuity of text from novel and movie scripts as training dataset. To dedicate to the sentence semantics, we utilize Microsoft COCO images to captions dataset, which is the only source of supervision in our project.

We tried to explore different ways of using the text of the books to derive semantic relatedness of the sentences, to generate stronger correlated context. We consider the previous and next sentence is related to the current sentence we are trying to encode, also, a number of linguistics researchers have shown first sentence of each paragraph have significant semantic relatedness with the rest of the sentence in this same paragraph, we also consider the first sentence of the paragraph that's more than 100 words long, is semantically related to current sentence too. Our experiments showed that by adding a more related feature, our model's performance increased in most of the cases, especially with different training data other than the romance novels.

We evaluate different variants of GRU in recurrent neural network by reducing parameters in the update and reset gates, we achieved roughly 30% training efficiency increase, while keeping the performance of the network virtually unchanged.

Related work

To start the entire project, it's very important for us to do researching on all related technologies and papers.

For sentence semantics, many existing systems employ a large amount of feature engineering and additional resources. In recent past years, several different approaches have been developed for learning compositions including recurrent networks [1], convolutional networks [2][3], and recursive-convolutional methods [4, 5]. All of those methods produce sentence representations with supervised learning and depend on class labels to do back propagate in order to optimize the model. Moreover, those methods only learn high quality sentence representations to work on very respective tasks.

The paper inspired us most describe an approach for unsupervised learning of a distributed sentence encoder by using skip-thought vector algorithm [6]. Sentences that share semantic and syntactic properties are thus mapped to similar vector representations. Some other papers help our design include [7][8][9].

Dataset

For training visual-semantic embedding modal, we use Microsoft COCO dataset. COCO is a large-scale object detection, segmentation, and captioning dataset. COCO has several features: Object segmentation; Recognition in context; Superpixel stuff segmentation. It has 330K images (>200K labeled) and 1.5 million object instances.

To train the recurrent neural network (RNN) decoder, we requested and leverage BookCorpus dataset. It includes all contents in 11,038 books. There are duplicated data inside and our model is tolerant with it.

# of books	# of sentences	# of words	# of unique words
11,038	74,004,228	984,846,357	1,316,420

Table 1: Summary statistics of the BookCorpus dataset. We use this corpus to train our model.

Another dataset we have is SICK (Sentences Involving Compositional Knowledge) dataset. The SICK data set consists of about 10,000 English sentence pairs, generated starting from two existing sets: the 8K ImageFlickr data set and the SemEval 2012 STS MSR-Video Description data set. Each sentence pair was annotated for relatedness and entailment by means of crowdsourcing techniques. The sentence relatedness score (on a 5-point rating scale) provides a direct way to evaluate CDSMs.

Methods

The entire system includes three parts, encoder, decoder and objective function. Encoder maps English sentence to a vector, decoder the conditions on this vector to generate a translation for the source sentence. We use RNN encoder with GRU activation and an RNN decoder with GRU. We focused on optimizing the first two parts.

Encoder Optimization

The Gated Recurrent Neural Network have shown success in applications involving sequential or temporal data but increase parameterization and is expensive. We experiment with different variation of GRU and reduce the parameters in the network without compromising the performance.

Let $w_i^1,\ldots,w_i^N$ be the words in sentence s_i where N is the number of words in the sentence. Encoder produces hidden states h_i^N which can be interpreted as the representation of the input words vector. The hidden state represents the full sentence. The activation function we are experimenting here are tanh and ReLU.

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$
$$\tilde{h}_t = g(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h)$$

With the two gates presented as update gate and reset gate:

$$z_t = \sigma(W_z x_t + U_z h_{t-1} + b_z)$$
$$r_t = \sigma(W_r x_t + U_r h_{t-1} + b_r)$$

Total number of parameters in the GRU RNN:

$$3 \times (n^2 + nm + n)$$

We tried a variant of the gates:

$$z_t = \sigma(U_z h_{t-1} + b_z)$$
$$r_t = \sigma(U_r h_{t-1} + b_r)$$

Total number of parameters reduced by:

$$2 \times mn$$

We observed comparable performance with less gate parameters from the variant we are using. Future work:

$$z_t = \sigma(U_z h_{t-1})$$
$$r_t = \sigma(U_r h_{t-1})$$

Decoder Optimization

For decoder, we defined "Smart Vector" modal to better extract semantics of sentences with unsupervised learning. For Smart Vector, one decoder is for the first sentence of each paragraph, one for the next sentence and one for the previous sentence. Separate parameters are used for each decoder. We introduce bias terms here for the update gate, reset gate and hidden state computation by the

$$h_{t+1} = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t$$
$$\tilde{h}_t = g(W^d x_t + U^d (r_t \odot h_{t-1}) + b_h)$$
$$z_t = \sigma(W_z^d x_{t-1} + U_z^d h_{t-1} + b_z)$$
$$r_t = \sigma(W_r^d x_{t-1} + U_r^d h_{t-1} + b_r)$$

Objective: given a tuple, where S0 is the first sentence of the paragraph

$$(s_0, s_{i-1}, s_i, s_{i+1})$$

The objective optimized is the sum of the log-probabilities for the first sentence of each paragraph, the forward and backward sentence conditioned on the encoder representation:

$$\sum_t \log P(w_0^t \mid w_0^{<t}, h_i) + \sum_t \log P(w_{i+1}^t \mid w_{i+1}^{<t}, h_i) + \sum_t \log P(w_{i-1}^t \mid w_{i-1}^{<t}, h_i)$$

Experiments and Results

In our experiments, we firstly evaluate the capability of our encoder as a generic feature extractor by SICK dataset. The setup process is shown as follows:

- Using the learned encoder as feature extractor to extract smart vectors for all sentences.
- Compute component-wise features between sentence pairs and compare it with the score stored inside the dataset.

There are some widely used ways to improve the performance, but because our goal is to evaluate the small vector modal on general data, we keep text pre-processing to minimum. When encoding new sentences, no additional preprocessing is performed except some basic tokenization. We want to see the real performance and robustness of the modal.

Quantitative Evaluation

SENTENCE 1	SENTENCE 2	GT	PD
A man is playing the drums	A man is playing the instrument	4.3	4.5
A man is playing the drums	A woman is playing the drums	4.2	4.0
A man is playing the drums	A man is disassembling the drum	3.4	3.6
A man is drawing some figures	A person is drawing some figures	4.6	4.6
A man is drawing on a digital dry erase board	A person is drawing some figures	3.2	3.5
A small dog is lying on the bed	A small dog is lying on a bed	5	4.9
A small dog is lying on a bed	A small cat is lying on a bed	3.2	3.5
A small dog is lying on the bed	A small animal is lying on the bed	4.5	4.4
A woman is cutting broccoli	A man is slicing tomatoes	2.6	4.2
A woman is cutting broccoli	There is no man cutting tomatoes	1.9	2.1
A woman is cutting a lemon	A woman is cutting fruit	4.2	4.3
A man is slicing some bread	A man is cutting a slice of bread	4.8	4.4
Nobody is slicing a piece of bread	A man is slicing some bread	3.6	4.2
Someone is pouring ingredients into a pot	Someone is adding ingredients to a pot	4.4	4.1
Nobody is pouring ingredients into a pot	Someone is pouring ingredients into a pot	3.5	4.2
Someone is pouring ingredients into a pot	A man is removing vegetables from a pot	2.4	3.5

Table 2: Example predictions from SICK test set. GT is the ground truth relatedness marked in dataset, scored between 1 and 5 and PD is prediction of score from our modal.

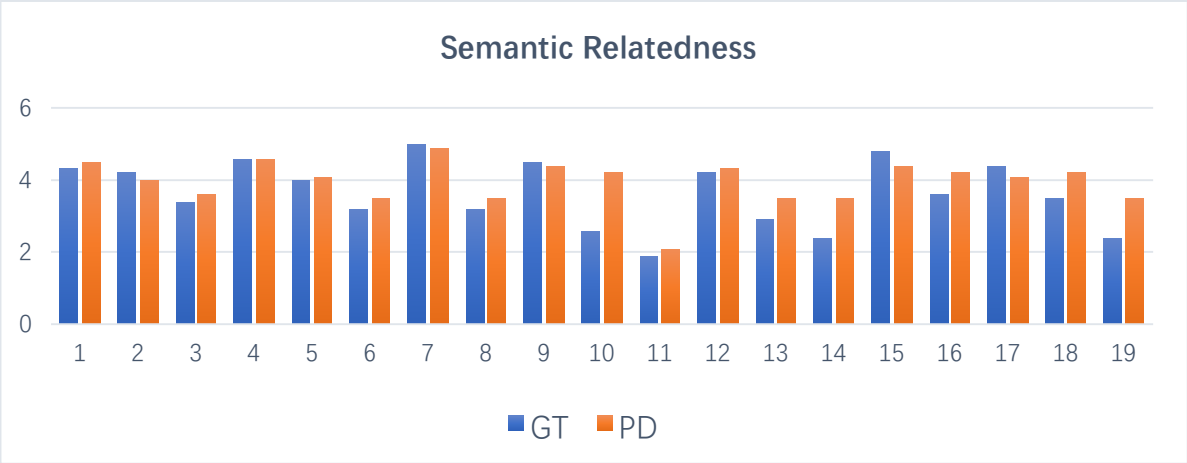

Figure 1: The GT and PD scores for more random examples from SICK dataset.

The above table and graphs show our modal has an overall good performance on analyzing the semantic relatedness between data. Even some difficult tasks have been handled well. However, some opposite pronouns bring some troubles to our modal. And slight changing sentence structure sometimes cause our model be unable to score correctly.

Qualitative Evaluation

After evaluating our smart vector modal, next step is to see how our storyteller performs on random pictures.

We first train the RNN decoder on romance novels in BookCorpus dataset. Each passage from a novel is mapped to a smart vector. The modal then conditions on the smart vector and aims to generate the passage that it has encoded.

At meantime, we train a visual-semantic embedding between COCO images and captions. In this model, captions and images are mapped into a common vector space. After training, we can embed new images and retrieve captions.

The ideal result is the story matches the content and semantics in the image. Main objects are recognized and reflected in the story. To better track the quality of generated stories, we defined the score code from 1 - 5 as follows:

Score	Standard
5	The story and image are matched very well. The story is meaningful and shows the emotion of image in good level.
4	The story and image are basically matched. The story contains some of important objects in the images but doesn't express the emotion very well.
3	The story and image are partially irrelevant. Some objects are reflected in the story, while others are not. The story doesn't express right emotion of the image.
2	The story and image are kindly irrelevant. Some objects are shown in the story but the semantics of story are not right.
1	The story and image are completely irrelevant or opposite

Table 3: Story quality score standards. Score 5 means the story generated is ideal and score 1 represent the worst case.

STORY (Score 5)

A woman was on the beach, holding her breath. She gave me a quick hug, and she had no idea what to do. In fact, it seemed as if I had never seen her come out of the surf. In fact, I was going to be the only woman in the world for the past twenty-four hours. She shook her head and bowed her head over my shoulder. In fact, it was so much easier for him to go on a tropical beach at The Shade. She felt as if I were the only woman in the world, standing naked on a sandy beach.

STORY (Score 4)

We were most people to the park, of course. It had been a long time since I walked out of the forest, and I couldn't help but notice that I had no idea what to do with them. In the midst of the crowd, it looked like they were walking away from my own personal space. By the time I reached the top of my building, I was pretty sure she was going to fall in love with him, soothing and soothing. Her father told her that he needed more than a few words.

STORY (Score 3)

We filled with the main room, and I expected to be quiet. The room was full of books and meetings, but I had no idea what to do with it. In fact, it seemed as if I had spent most of the rest of my life. As long as I was seated at the desk, I couldn't help but notice that there was no room for him to enter. I put my hands on Elena 's desk and bowed her head, making sure she understood that he wanted more than a few tables. They were very depressing, the only way to have room and wall records.

STORY (Score 1)

We men were in complete control of their meeting, and I looked up at my best friend. Of course, I did n't have to ask for more details. In fact, he was the most handsome man she had ever met. He wore a suit and tied ties with each other as if that would be the case, but there was no reason for him to leave her behind. I forced myself to go back and forth between the two men. A few minutes later, I felt relieved that the Master of Cadogan House fit us. I m not going to talk, until these men and I cared less.

Conclusion

The main goal of this project is to combine visual and language processing by building an intelligent storyteller based on input images. We create an intelligent storyteller, by giving any image, it can try to understand the scenario and semantics from it and generate a meaningful story. During the project, we design a smart vector modal to better abstract sentence semantics which produces an overall good performance on random images.

Future Work

We will find a way to improve training efficiency in further and reduce computational complexity in network parameterization. Also, instead of a simple description of the picture, we want to optimize our module to understand the meaning of a picture.

Reference

[1] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

[2] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. ACL, 2014.

[3] Yoon Kim. Convolutional neural networks for sentence classification. EMNLP, 2014.

[4] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. SSST-8, 2014.

[5] Han Zhao, Zhengdong Lu, and Pascal Poupart. Self-adaptive hierarchical sentence model. IJCAI, 2015.

[6] Ryan Kiros , Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Skip-Thought Vectors

[7] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. EMNLP, 2014.

[8] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neural networks. In NIPS, 2014.

[9] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. ICLR, 2015.