Guiding the Long-Short Term Memory model for Image Caption Generation

Xu Jia
KU Leuven ESAT-PSI, iMinds
Xu.Jia@esat.kuleuven.be

Efstratios Gavves*
QUVA Lab, University of Amsterdam
E.Gavves@uva.nl

Basura Fernando*
ACRV, The Australian National University
basura.fernando@anu.edu.au

Tinne Tuytelaars
KU Leuven ESAT-PSI, iMinds
Tinne.Tuytelaars@esat.kuleuven.be

Abstract

In this work we focus on the problem of image caption generation. We propose an extension of the long short term memory (LSTM) model, which we coin gLSTM for short. In particular, we add semantic information extracted from the image as extra input to each unit of the LSTM block, with the aim of guiding the model towards solutions that are more tightly coupled to the image content. Additionally, we explore different length normalization strategies for beam search to avoid bias towards short sentences. On various benchmark datasets such as Flickr8K, Flickr30K and MS COCO, we obtain results that are on par with or better than the current state-of-the-art.

1. Introduction

Recent successes in visual classification have shifted the interest of the community towards higher-level, more complicated tasks, such as image caption generation [7, 9, 17, 19, 20, 21, 22, 23, 26, 27, 28, 37, 38, 39]. Although for a human describing a picture is natural, it is quite difficult for a computer to imitate this task. It requires the computer to have some level of semantic understanding of the content of an image, including which kinds of objects there are, what they look like, what they are doing, and so on. Last but not least, this semantic understanding has to be structured into a human-like sentence.

Inspired by recent advances in machine translation [1, 5, 32], neural machine translation models have lately been applied to the image caption generation task [7, 17, 26, 37, 38], with remarkable success. In particular, compared to template-based methods [9, 20, 28, 39], that use a rigid sentence structure, and transfer-based methods [21, 22, 23, 27], that re-use descriptions available in the training data, methods based on neural machine translation models stand out thanks to their capability to generate new sentences. They manage to effectively generalize beyond the sentences seen at training time, which is possible thanks to the language model learnt. Most neural machine translation models follow an encoder-decoder pipeline [1, 5, 32], where the sentence in the source language is first encoded into a fixed-length embedding vector, which is then decoded to generate a new sentence in the target language. For machine translation, parallel corpora are typically used for learning and evaluating the model [1, 5, 32]. The pairs of sentences in the source and target languages usually share similar sentence structures (often including regular phrases and the same order of words). This structural information is encoded in the fixed-length embedding vector and is helpful to the translation.

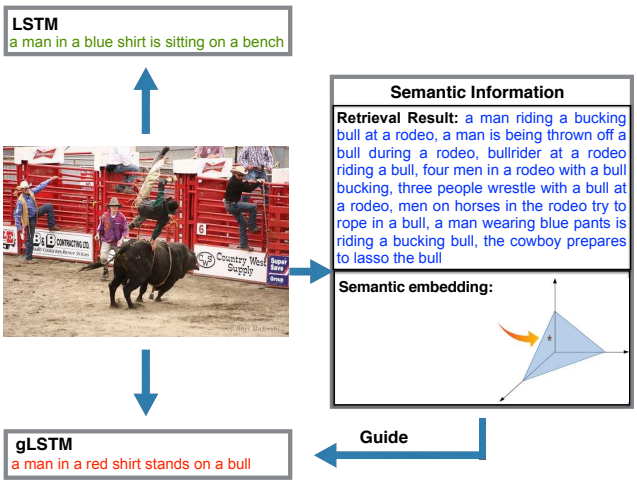


Figure 1: Image caption generation using LSTM and the proposed gLSTM. The generation by LSTM and gLSTM and the cross-modal result that is used as guidance, are marked respectively in green, red and blue.

Applied to caption generation, the aim is to "translate"

*This work was carried out while he was in KU Leuven ESAT-PSI.

an image into a sentence describing it. However, it is questionable whether these models can cope with the large differences between the two modalities. The structure of the visual information is very different from the structure of the description to be generated. During the encoding phase, the algorithm compresses all visual information into an embedding vector. Yet this vector is unlikely to capture the same level of structural information needed for correctly generating the textual description in the subsequent decoding phase.

One of the latest state-of-the-art methods [37] uses a convolutional neural network (CNN) for the encoding step and the long-short term memory (LSTM) network for the decoding step. While experimenting with this scheme, we notice that sometimes the generated sentence seems to "drift away" or "lose track" of the original image content, generating a description that is common in the dataset, yet only weakly coupled to the input image. We hypothesize this is because the decoding step needs to find a balance between two, sometimes contradicting, forces: on the one hand, the sentence to be generated needs to describe the image content; on the other hand, the generated sentence needs to fit the language model, with more likely word combinations to be preferred. The system then may "lose track" of the original image content if the latter force starts to dominate. From an image caption generation point of view, however, staying close to the image content may be considered the most important of the two.

To overcome the limitation of the basic encoding-decoding pipeline, extended pipelines have been proposed in the context of both machine translation [1] and image caption generation [38]. They introduce an attention mechanism to align the information in both the source and target domains, so that the model is able to attend to the most relevant part in the sentence from the source language or image.

Here, we propose an alternative extension of the LSTM model, that works at a more global scale. We start by extracting semantic information from the image and then use it to "guide" the decoder, keeping it "on track" by adding a positive bias to words that are semantically linked to the image content. More specifically, we add semantic information as an extra input to the gate of each LSTM memory cell. This extra input can take many different forms as long as they build a semantic connection between the image and its description, *e.g*. a semantic embedding, a classification or retrieval result. As an illustration we experiment with features either obtained from a multimodal semantic embedding using CCA or, the retrieved image descriptions.

Our contributions are two-folded. As our main contribution, we present an extension of LSTM which is guided by semantic information of image. We refer to the proposed method as gLSTM. We show experimentally on multiple datasets that such guiding is beneficial for learning to generate image captions. As an additional contribution, we make the observation that current inference methodologies for caption generation are heavily biased towards short sentences. We show experimentally that this hurts the quality of the generated sentences and therefore propose sentence normalization, which further improves the results. In the experiments, we show that the proposed method is on par or better than the latest state-of-the-art on the popular datasets.

2. Related Work

Caption generation. The literature on caption generation can be divided into three families. First, there are template-based methods [9, 20, 28, 39]. These approaches first detect objects, actions, scenes and attributes, then fill them in a fixed sentence template, *e.g*. using a subject-verb-object template. These methods are intuitive and can work with out-of-the-box visual classification components. However, they require explicit annotations for each class. Given the typically small number of categories available, these methods do not generate rich enough captions. Moreover, as they use rigid templates the generated sentence is less natural.

Second, there are also transfer-based caption generation strategies [21, 22, 23, 27]. They are related to image retrieval. These methods first retrieve visually similar images, then transfer captions of those images to the query image. The advantage of these methods is that the generated captions are more human-like than the generations by template-based methods. However, because they directly rely on retrieval results among training data, there is little flexibility for them to add or remove words based on the content of an image.

Inspired by the success of neural networks in machine translation [1, 5, 32], recently people have proposed to use neural language models for caption generation. Instead of translating a sentence from a source language into a target one, the goal is to translate an image into a sentence that describes it. In [19] a multimodal log-bilinear neural language model is proposed to model the probability distribution of a word conditioned on an image and previous words. Similarly, Mao *et al*. [26] and Karpathy *et al*. [17] have proposed to use a multimodal recurrent neural network [31] model for caption generation. Vinyals *et al*. [37] and Donahue *et al*. [7] have proposed to use LSTM [14], an advanced Recurrent Neural Network for the same task. Very recently, Xu *et al*. [38] have proposed to integrate visual attention into the LSTM model in order to fix its gaze on different objects during the generation of corresponding words. Neural language models have shown great prospects in generating human-like image captions. Most of these methods follow a similar encoding-decoding framework, except for the very recent method [38] which jointly learns visual attention and caption generation. However, [38] requires location sam-

pling both during training and testing, making the method more complicated. While they focus more on local information, our method rather exploits global cues.

Overview. Our work belongs to the third family of caption generation methods which uses a neural language model to generate captions. Different from the above methods, however, we propose to make use of the semantic information to guide the generation and propose an extension of LSTM model, coined gLSTM for the use of semantic information. The semantic information here denotes the correlation between an image and its description, which is obtained in a similar manner as in transfer-based methods. Experiments illustrate that semantic information brings significant improvement in the performance and our model outperforms recently proposed state-of-the-art methods [17, 37]. Interestingly, the proposed model is able to perform on par with the latest and unpublished state-of-the-art [38], despite their use of more complicated models that require location sampling during training and test stage.

3. Background

3.1. The LSTM Model

A Recurrent Neural Network (RNN) is a good choice to model temporal dynamics in sequences. However, it is difficult for traditional RNN to learn long-term dynamics because of the issue of vanishing and exploding gradients [14]. The Long Short-Term Memory (LSTM) network is proposed in [14] to address these issues. The core of the LSTM architecture is the memory cell, which stores the state over time, and the gates, which control when and how to update the cell's state. There are many variants with different connections between the memory cell and the gates.

The LSTM block that our model is built on follows the *LSTM with No Peepholes* architecture [13], which is illustrated in Figure 2 in black. The memory cell and gates in an LSTM block are defined as follows:

$$i_l = \sigma(W_{ix}x_l + W_{im}m_{l-1}) \quad (1)$$
$$f_l = \sigma(W_{fx}x_l + W_{fm}m_{l-1}) \quad (2)$$
$$o_l = \sigma(W_{ox}x_l + W_{om}m_{l-1}) \quad (3)$$
$$c_l = f_l \odot c_{l-1} + i_l \odot h(W_{cx}x_l + W_{cm}m_{l-1}) \quad (4)$$
$$m_l = o_l \odot c_l \quad (5)$$

where $\odot$ represents the element-wise multiplication, $\sigma(\cdot)$ represents the sigmoid function and $h(\cdot)$ represents the hyperbolic tangent function. The variable i_l stands for the input gate , f_l for the forget gate, o_l for the output gate of the LSTM cell, c_l is the state of the memory cell unit and m_l is the hidden state, that is the output of the block generated by the cell. The variable x_l is the element of the sequence at timestep l and $W_{[\cdot][\cdot]}$ denote the parameters of the model.

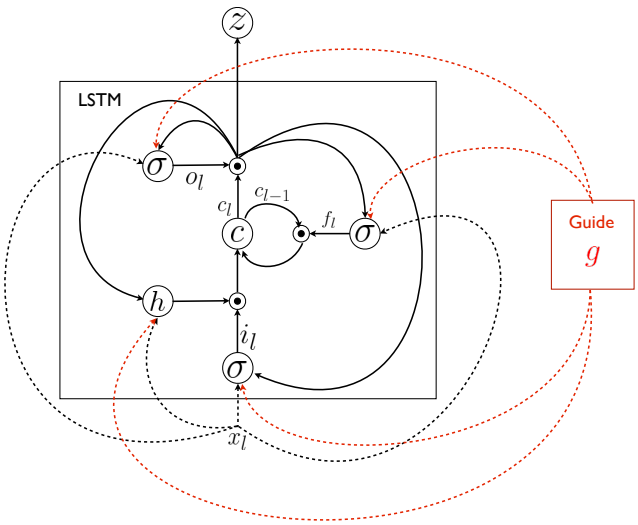


Figure 2: The LSTM block in black, the proposed gLSTM network in black and red. Striped lines stand for external connections. By considering semantic information as an extra input, we encourage the network to refresh its memory following a global guide.

3.2. Caption Generation with LSTM

The pipeline for caption generation with the RNN model [7, 17, 26, 37, 38] is inspired by the encoder-decoder principle in Neural Machine Translation [1, 5, 32]. An encoder is used to map a variable length sequence in the source language into a distributed vector and a decoder is used to generate a new sequence in the target language conditioned on this vector. During training, the goal is to maximize the log-likelihood of correct translation given the sentence in the source language. When applying this principle to caption generation, the goal becomes to maximize the log-likelihood of the image caption given an image, namely

$$\arg\max_{\theta} \sum_i \log p(s^i_{1:L^i}|x^i, \theta), \quad (6)$$

where x^i denotes an image, $s^i_{1:L^i}$ denotes a sequence of words in a sentence of length L^i and θ denotes the model parameters. For simplicity, in the following part we drop the superscript i whenever it is clear from the context. Since each sentence is composed of a sequence of words, it is natural to use Bayes chain rule to decompose the likelihood of a sentence,

$$\log p(s_{1:L}|x, \theta) = \log p(s_1|x, \theta) + \sum_{l=2}^{L} \log p(s_l|x, s_{1:l-1}, \theta), \quad (7)$$

where $s_{1:l}$ stands for the part of the sentence up to the l-th word. To maximize the objective in eq. (6) over the whole training corpus, we need to define the log-likelihood $\log p(s_l|x, s_{1:l-1}, \theta)$, which can be modeled with the hidden state of a timestep in RNN. The probability distribution

of the word at timestep $l+1$ over the whole vocabulary is computed using the softmax function $z(\cdot)$ based only on the output m_l of the memory cell, $p_{l+1} = z(m_l)$ similar to [37].

To feed images and sentences to LSTM, they need to be encoded as fixed-length vectors. For the image, CNN features are first computed and then mapped to an embedding space via a linear transformation. For the sentence, each word is first represented as a one hot vector and then mapped to the same embedding space via a word embedding matrix. Finally, an image and sequence of words in a sentence are concatenated to form a new sequence, that is, the image is treated as the beginning symbol of the sequence and the sequence of words forms the remaining part of the new sequence. This sequence is fed to the LSTM network for training by iterating the recurrence connection for l from 1 to L^i. The parameters of the model include the linear transformation matrix for image features, the word embedding matrix and the parameters of LSTM.

3.3. Normalized Canonical Correlation Analysis

To build our semantic representation, we rely on normalized Canonical Correlation Analysis (normalized CCA), proposed in [10] to address the cross-modal retrieval problem. Canonical Correlation Analysis (CCA) [16] is a popular method used to map visual and textual features into a common semantic space. CCA aims at learning projection matrices U_1 and U_2 for two views X_1 and X_2 such that their projections are maximally correlated, namely,

$$\arg\max_{U_1,U_2} \frac{U_1\Sigma_{12}U_2}{\sqrt{U_1\Sigma_{11}U_1}\sqrt{U_2\Sigma_{22}U_2}}, \tag{8}$$

where Σ_{12}, Σ_{11} and Σ_{22} are the covariance matrices. The CCA objective function can be solved via generalized eigenvalue decomposition. The normalized CCA [10] is computed by using a power of the eigenvalues to weight the corresponding columns of the CCA projection matrices, and followed by L2 normalization, that is,

$$g_1 = \frac{X_1U_1D^p}{\|X_1U_1D^p\|}, \quad g_2 = \frac{X_2U_2D^p}{\|X_2U_2D^p\|} \tag{9}$$

where D is a diagonal matrix whose elements are set to the eigenvalues of corresponding dimensions, while g_1 and g_2 denote the semantic representation of the two views. Cosine similarity is used to find the nearest neighbor in the learned common semantic space [10].

4. The Proposed Methods

In this section, we describe the proposed extension of the LSTM model for the caption generation task. In the new architecture, we add semantic information to the computation of the gates and cell state. The semantic information here is extracted from images and their descriptions, serving as a guide in the process of word sequence generation.

4.1. gLSTM

The generation of a word in the LSTM model mainly depends on the word embedding at the current timstep and the previous hidden state (which includes image information at the beginning). This process goes step by step until it encounters the end token of a sentence. However, as this process continues, the role of the image information, which is only fed at the beginning, becomes weaker and weaker. Words generated at the beginning of a sequence also suffer from the same problem. Therefore, for a long sentence, it may carry out the generation almost "blindly" towards the end of the sentence. Though LSTM is able to keep long-term memory to some extent, still it poses a challenge for sentence generation [1, 4]. In the proposed model, the generation of words is carried out under the guidance of global semantic information. Our extension of LSTM model is named gLSTM. The memory cell and gates in a gLSTM block are defined as follows:

$$\begin{aligned}
i'_l &= \sigma(W_{ix}x_l + W_{im}m'_{l-1} + W_{iq}g) && (10)\\
f'_l &= \sigma(W_{fx}x_l + W_{fm}m'_{l-1} + W_{fq}g) && (11)\\
o'_l &= \sigma(W_{ox}x_l + W_{om}m'_{l-1} + W_{oq}g) && (12)\\
c'_l &= f'_l \odot c'_{l-1} + i'_l \odot h(W_{cx}x_l + \\
&\quad +W_{cm}m'_{l-1} + W_{cq}g) && (13)\\
m'_l &= o'_l \odot c'_l && (14)
\end{aligned}$$

where g denotes the vector representation of semantic information. Compared to the standard LSTM architecture, in gLSTM we add a new term to the computation of each gate and cell state. This new term represents the semantic information which works as a bridge between visual and textual domains. The semantic information g does not depend on the timestep l, hence working as a global guide during the caption generation. The guidance term can also be made timestep dependent in expense of higher complexity models. We summarize with red the gLSTM network architecture additions in Figure 2.

4.2. Semantic Information.

In this section, we detail several kinds of semantic information that can be used as guidance in our model. Intuitively, there are three ways to extract the semantic information. First, we treat it as a cross-modal retrieval task and simply use the retrieved sentences as semantic information. Alternatively, semantic information can also be represented as the embedding in a semantic space where visual and textual representations are equivalent. The last one is to use the image itself as guidance.

Retrieval-based guidance (ret-gLSTM). The retrieval-based guidance is inspired by transfer-based caption generation methods. Though the generated sentences given by transfer-based methods may not be totally correct, they do

have something in common with the true captions annotated by humans. Given an image, we first do the cross-modal retrieval so as to find texts relevant to the query image. We collect descriptions with top rankings. Instead of generating a sentence by making direct modification on these sentences, we treat these captions as auxiliary information and feed them to the neural language model we proposed in the previous section. These sentences may not match perfectly to the image. However, they provide rich semantic information for the image. Since these sentences are annotated by humans, the words in these sentences are very natural and have a high probability to appear in the reference captions.

The cross-modal retrieval method used here is based on the normalized CCA mentioned in Section 3.3. In this paper, image and text features correspond to the two views for CCA. CNN features are computed for the images and TF-IDF weighted BoW features are computed for the sentences. We project both images and sentences from their own domain to the common semantic space. Given an image query, the closest sentences are then retrieved based on cosine similarity. We select the top T retrieved sentences from the training set ($T = 15$ in this paper). These sentences are represented by a bag-of-words (BoW) vector which is fed as extra input, i.e. the guide to the gLSTM model.

Semantic embedding guidance (emb-gLSTM). Instead of explicitly using the result of cross-modal retrieval as guidance as mentioned above, we can also implicitly use the intermediate result of cross-modal retrieval, that is, the semantic representation computed using normalized CCA as the extra input. An image is mapped into the common semantic space by the learned projection matrix and the computed semantic embedding is fed to gLSTM model as the guide. It is assumed that in the common semantic space of CCA both views share equivalent embedding representations. Therefore, we can treat the projected representation from image domain as equal to the one projected from text domain. Compared to the ret-gLSTM model, the semantic representation has much lower dimensionality than the BoW representation and saves the computation of finding nearest neighbors. In addition, we also find it even performs better than the previous method.

Image as guidance (img-gLSTM). Finally, we experiment with the image itself as the extra input. This is motivated by the fact that CCA is a linear transformation. A natural question then is whether we can learn this projection matrix directly during the training of the gLSTM model. Therefore, we add the image itself as a third kind of extra input. We experimentally verify this by simply feeding the image feature itself to the gLSTM model, namely $g = x$, and let the network learn the semantic information from scratch.

4.3. Beam Search with Length Normalization

In the generation stage, with a vocabulary of size K, there are K^l sentences of length l as potential candidates for an image caption, where l is unknown. Ideally, we want to find the sentence, which maximizes the log-likelihood of eq. (7). Considering the exponential search space, however, exhaustive search is intractable. Therefore, a heuristic search strategy is employed instead.

Here we use *beam search*, which is a fast and effective decoding method for RNN-based models [11, 32]. At each iteration only the T hypotheses generations with the highest log-likelihood are kept in the beam pool. The search along one beam stops once it encounters an end-of-sequence token which is generated given previous words along the beam. The searching process continues until the searching along all beams in the pool stops.

It is problematic to directly use the log-likelihood of words as the criterion to select a generation. Since the log-likelihood of each single word is negative (because the probability is smaller than 1), summation over the log-likelihood of more words leads to a smaller value. Therefore, when the beam width is larger than 1, there is a bias towards short sentences. That means this kind of beam search favors shorter sentence, which is also observed in [4, 12]

Interestingly, the bias towards short sentences tends to favor the low order of BLEU scores (BLEU@1,2), commonly used to evaluate machine translation algorithms. Hence, short sentences not only tend to dominate the inference, but also obscure the evaluations and methodology comparisons. To remedy the bias towards short sentences during inference, we propose to normalize the log-likelihood of words by length, namely

$$p = \frac{1}{\Omega(\ell)} \sum_{l=1}^{\ell} \log p(s_l | x, s_{1:l}, \theta) \tag{15}$$

We investigate various forms for Ω to do the normalization.

Polynomial normalization. A first possibility is to set $\Omega(\ell) = |\ell|^m$. Notice that when $m = 1$, eq. (15) becomes the definition of the perplexity. We use $m = 1$ in our paper. This kind of normalization punishes short sentences.

Min-hinge normalization. Intuitively we want to automatically generate a sentence whose length is close to the ground truth. Since in the test stage we do not know the length in advance, we use the average length of the sentences in the training data as a reference. We define the min-hinge length function as $\Omega(\ell) = \min\{\ell, \mu\}$. This means a generated sentence is only punished when it is shorter than the average length μ. For sentences that are long enough, we only pay attention to their log-likelihood.

Max-hinge normalization. Similarly, we define the max-hinge length function , $\Omega(\ell) = \max\{\ell, \mu\}$. Instead of pe-

nalizing short sentences, the max-hinge function favors long sentences.

Gaussian normalization. We can also employ a Gaussian function, $\Omega(\ell) \sim \mathcal{N}(\mu, \sigma)$ to normalize the loglikelihood, where the μ and σ are the mean and the standard deviation of the sentence lengths in the training corpus. The Gaussian regularization encourages the inference to select sentences that have similar lengths as the sentences in the training set.

We experimentally verify the effectiveness of these strategies in Section 5.1.

5. Experiments

Datasets and experimental setup. We perform experiments on the following datasets. *Flickr8k [15], Flickr30k [40] and MS COCO [25]*. The Flickr8k dataset is a popular dataset composed of $8,000$ images in total collected from Flickr, divided into a training, validation and test set of $6,000$, $1,000$ and $1,000$ images respectively. Each image in the dataset is accompanied with 5 reference captions annotated by humans. Similar to Flickr8k, the Flickr30k dataset contains $31,000$ images collected from Flickr, together with 5 reference sentences provided by human annotators. However, it does not provide a split setting file. So we use the publicly available split setting used in [17, 18], that is, $29,000$ images for training, $1,000$ for validation and $1,000$ for testing. The large scale dataset MSCOCO contains 82,783 images for training and 40504 for validation, with each image associated with 5 captions. Note that we do not evaluate it on the test set used for MS COCO Image Captioning challenge but use the publicly available splits used in previous work [17], that is, all 82,783 images from training set for training, 5,000 images from validation set for validation and another 5000 from validation set for testing.

Evaluation measures. Here we use the two most popular measures in the machine translation and image caption generation literature, namely the *BLEU* [29] and the *METEOR* [6] measure.

BLEU is a precision-based metric. The main component of BLEU is n-gram precision of the generated caption with respect to the references. Precision is computed separately for each n-gram and then B@n is computed as a geometric mean of these precisions. *BLEU* of high order n-grams indirectly measures the grammatical coherence.

However, *BLEU* is criticized to favor short sentences. It only considers precision but does not take recall into consideration. For this reason *METEOR* is also reported in recent works [3, 8, 38]. *METEOR* evaluates a generated sentence by computing a score based on word level matches between the generation and a reference and returning the maximum score over a set of references. In the computation of the matching score, it considers unigram-precision, unigram-recall and a measure of alignment. Hence, *METEOR* accounts for precision, recall and the importance of grammaticality. In user evaluation studies *METEOR* [24] has been shown to have a higher correlation with human judgments than any order of *BLEU*.

Besides, we also compute the *CIDEr* score [36] for the experiment on MS COCO. All scores are computed using the coco-caption code [1].

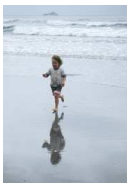
Implementation details. In the following experiments we use the MatConvNet toolbox [35] and the 16-layer OxfordNet [30] pretrained model to compute CNN features and extract the last fully-connected layer's output as image representation. As for preprocessing of texts, for the neural language model, we use the publicly available data where texts are converted to lowercase, non-alphanumeric characters are ignored and only words appearing at least 5 times in the training set are kept to create a vocabulary. For CCA, we use the NLTK toolbox [2] to further lemmatize words and build a vocabulary based on the most frequent words (3000 words for flickr8k and 5000 for flickr30k and MS COCO). Then tf-idf-weighted BoW vectors are computed as sentence representation for CCA. For Flickr8k and Flickr30K we set the number of dimensions for the image and word embeddings and the hidden layer of the gLSTM to 256. For MSCOCO we set the number to 512 (note that this is much smaller than the one used in other work). The gLSTM Models are trained with RMSProp [34], which is a stochastic gradient descent method using an adaptive learning rate algorithm. The learning rate is initialized with 1e-4 for Flickr8k and Flickr30k and 4e-4 for MS COCO. We use dropout and early stopping to avoid overfitting and use validation set log-likelihood for model selection. For CCA, we set $p = 4$ as suggested in [10] and the dimension of the common space to 200 for Flickr8k and Flickr30k, and 500 for MS COCO which we find works well in practice. At the test stage, we set the beam size to 10 for all experiments. We built our code for the proposed gLSTM model on Karpathy's NeuralTalk code [2], which implements the single model in Google's paper [37]. Note that we take that model as the **baseline**.

5.1. Length Normalization

In this experiment we evaluate the importance of the sentence length normalization to caption generation. We carry out the experiment on the Flickr8k dataset and report the results in Table 1. For clarity we perform this experiment based on the LSTM baseline, not gLSTM.

We observe that compared to the baseline whose selection is based on unnormalized log-likelihood, length normalization has a positive effect on either the *BLEU* metric or *METEOR* metric. Polynomial, min-hinge and Gaussian normalization respectively bring the largest improvement

[1] https://github.com/tylin/coco-caption
[2] https://github.com/karpathy/neuraltalk



a young boy is running on the beach, a man in a blue shirt is riding a dirt bike, a little boy runs away from the approaching waves of the ocean, a little girl runs across the wet beach, a little girl runs on the wet sand near the ocean, a young girl runs across a wet beach with the ocean in the background, child running on the beach, two children are running towards the ocean on a beach, a dog is running in the ocean beside the beach, a dog playing in the ocean on the beach , a boy running through surf on a beach, boy running through the water at the beach, a girl runs down a beach, a boy standing on a beach, a man riding his bike on the beach by the ocean, a young girl running on the beach, a dog is running on the beach, a young child running along the shore at a beach, boy and girl running along the beach, a dog running on the beach, a dog running on the beach, a dog running on the beach

a group of dogs are running on a track, a group of people racing on a track, a a dog with a muzzle is leading several other dogs in a race, a greyhound leaps in a race, a muzzled dog in a race with four dogs following, five dogs are racing, five dogs are racing on a dirt track, two greyhounds with muzzles race along the inside curb of a railed dirt track, the greyhound racing dogs are running around a bend in the track, three muzzled greyhounds race around a turn in a track, several muzzled greyhound dogs racing around a track, two muzzled greyhounds dogs racing around a track, two greyhounds race around a track, greyhounds racing chasing a mechanical rabbit around the track, three greyhounds are racing on a track at night, three greyhound dogs race around a dark track, muzzled greyhounds are racing along a dog track at night, three greyhounds racing around the corner of a track, greyhounds racing on a track, greyhounds race on a track, greyhounds race on a track, three greyhounds are in a dog race at the track

a woman in a black shirt and sunglasses smiles, a man and a woman pose for a picture, a brunette girl wearing sunglasses and a yellow shirt, a girl in sunglasses smiles, a girl wearing a yellow shirt and sunglasses smiles, a girl wearing sunglasses smiles for the camera, a woman with a yellow shirt wears sunglasses and smiles, a woman wearing sunglasses smiles, young man with upturned hair posing with young man with sunglasses and woman with glasses, a blonde woman wearing sunglasses and dice earrings smiles, a woman wearing black sunglasses looks to the right and smiles, a smiling woman is wearing sunglasses on a day with sparse clouds, a smiling woman with long dark hair wearing sunglasses on top of her head, a man and woman wearing sunglasses and white t-shirts smile for the camera, a man in sunglasses smiles, a blonde lady with sunglasses smiles, women in hat and sunglasses smiles, a woman wearing sunglasses, man and woman wearing sunglasses posing for picture, woman with green sweater and sunglasses smiling, a woman in a sunhat is wearing sunglasses and laughing, a woman wearing sunglasses on her head looking down

Figure 3: Results of the gLSTM and LSTM model. We mark the generated sentence by gLSTM and LSTM respectively in red and green, the ground truth references in black and the most relevant retrieval results in blue. We observe that the retrieval results are helpful to caption generation. Notice that for the third example, the result of our model is not that accurate but still much better than the one of the LSTM model.

Normalization	B@1	B@2	B@3	B@4	METEOR
Baseline	59.6	40.4	26.1	17.0	17.45
Polynomial	57.8	39.2	26.0	17.6	**18.86**
Min-hinge	60.4	41.4	27.6	**18.6**	18.53
Max-hinge	57.6	38.8	25.2	16.7	17.65
Gaussian	**60.7**	**41.7**	**27.8**	**18.6**	18.35

Table 1: The performance of different length normalization strategies on Flickr8k.

GT Refs	*Baseline*	*Polynom.*	*Min-hinge*	*Max-hinge*	*Gaussian*
10.87(3.74)	8.75(2.44)	11.07(2.62)	9.64(1.92)	9.55(1.69)	9.57(3.30)

Table 2: The average and the standard deviation of the sentence length for the ground truth references, and different normalization strategies on Flickr8k.

to *METEOR* and *BLEU*. Therefore, in the following experiments, we only report the performance of the proposed gLSTM with these three kinds of length normalization. Besides, we also compute the average length of generated sentences and references.

5.2. gLSTM with Different Types of Guidance

In this experiment we evaluate the gLSTM model with different types of semantic information, as described in Section 4.2. For fair comparison, we also apply beam search with length normalization to the baseline. We run this experiment on Flickr8k and report the results in Table 3.

	B@1	B@2	B@3	B@4	METEOR
Baseline, Original	59.6	40.4	26.1	17.0	17.45
Baseline, Polynomial	57.8	39.2	26.0	17.6	18.86
Baseline, Min-hinge	60.4	41.4	27.6	18.6	18.53
Baseline, Gaussian	60.7	41.7	27.8	18.6	18.35
Baseline 512, Original	61.0	42.4	28.6	18.9	18.21
Baseline 512, Polynomial	58.2	40.2	27.1	18.1	19.83
Baseline 512, Min-hinge	61.3	42.9	29.2	19.6	19.13
Baseline 512, Gaussian	61.3	42.8	29.1	19.5	19.07
ret-gLSTM, Original	63.4	43.7	29.2	19.3	18.54
ret-gLSTM, Polynomial	58.8	40.4	27.5	18.6	19.86
ret-gLSTM, Min-hinge	63.0	43.8	29.9	20.2	19.46
ret-gLSTM, Gaussian	63.5	44.2	30.2	20.6	19.38
emb-gLSTM, Original	63.7	44.7	30.2	20.2	19.10
emb-gLSTM, Polynomial	61.0	43.0	29.6	20.1	**20.60**
emb-gLSTM, Min-hinge	64.3	45.7	31.6	21.5	20.28
emb-gLSTM, Gaussian	**64.7**	**45.9**	**31.8**	**21.6**	20.19
img-gLSTM, Original	61.5	42.5	27.2	16.7	17.10
img-gLSTM, Polynomial	55.7	38.1	24.9	15.8	17.69
img-gLSTM, Min-hinge	60.4	41.9	27.6	17.7	17.76
img-gLSTM, Gaussian	60.1	41.4	27.2	17.3	17.69

Table 3: Comparison between gLSTM with different semantic information on Flickr8k. We denote the gLSTM model with retrieval-based guidance as ret-gLSTM, the one with semantic embedding guidance as emb-gLSTM, and the one with the image as guidance as img-gLSTM.

The result illustrates that semantic information brings

	Flickr8k					*Flickr30k*				
	B@1	B@2	B@3	B@4	METEOR	B@1	B@2	B@3	B@4	METEOR
LogBilinear [19]	65.6	42.4	27.7	17.7	17.31	60.0	38.-	25.4	17.1	16.88
multimodal RNN [17]	57.9	38.3	24.5	16.0	16.7	57.3	36.9	24.0	15.7	15.3
Google NIC [37]	63.-	41.-	27.-	—	—	66.3	42.3	27.7	18.3	—
LRCN-CaffeNet [7]	—	—	—	—	—	58.7	39.1	25.1	16.5	—
m-RNN-AlexNet [26]	—	—	—	—	—	54.-	36.-	23.-	15.-	—
m-RNN [26]	—	—	—	—	—	60.-	41.-	28.-	19.-	—
Soft-Attention [38]	**67.-**	44.8	29.9	19.5	18.93	66.7	43.4	28.8	19.1	18.49
Hard-Attention [38]	**67.-**	45.7	31.4	21.3	20.3	**66.9**	43.9	29.6	19.9	18.46
emb-gLSTM, Polynomial	61.0	43.0	29.6	20.1	**20.60**	59.8	41.3	29.3	19.2	**18.58**
emb-gLSTM, Min-hinge	64.3	45.7	31.6	21.5	20.28	63.8	44.1	30.2	20.5	18.13
emb-gLSTM, Gaussian	64.7	**45.9**	**31.8**	**21.6**	20.19	64.6	**44.6**	**30.5**	**20.6**	17.91

Table 4: Comparison with state-of-the-art methods on Flickr8k and Flickr30k.

much improvement in the performance, especially emb-gLSTM, the gLSTM with semantic embedding guidance. We also observe that img-gLSTM, the gLSTM with the image itself as guidance, does not bring any improvement but even deteriorates the performance. Besides, we also conduct an experiment for a baseline but with more parameters (512 dimension instead of 256 dimension) for each gate to emphasize the improvement mainly comes from the global guide. The total number of network parameters is 5.2M in total compared to 5.9M and 3.1M for the proposed ret-gLSTM and emb-gLSTM. As is shown in Table 3, we can see that increasing parameters indeed improves the performance, but still a little worse than the proposed emb-gLSTM even though it has much fewer parameters.

5.3. Comparison with State-of-the-art methods

We compare the proposed gLSTM with state-of-the-art methods for caption generation in the literature. We perform the experiment on Flickr8k and Flickr30k and report the results in Table 4. We only evaluate emb-gLSTM in this experiment, since it is computationally efficient and obtains the best performance among the different models in the previous experiment. For most evaluated methods, they use CNN with deeper network architecture such as OxfordNet [30] and GoogLeNet [33]. Methods which do not use a deeper CNN include LRCN-CaffeNet [7] and m-RNN-AlexNet [26]. Note that Google's method [37] uses an ensemble of multiple LSTM models, while ours only uses a single emb-gLSTM model. We can see from the table, the proposed emb-gLSTM model performs favorably against state-of-the-art approaches. Interestingly, it performs even on par with the latest state-of-the-art [38], which is based on more complicated and expensive attention mechanisms.

6. Conclusion

In this work we have proposed an extension of the LSTM model for image caption generation. By adding semantic information as extra input to each unit of the LSTM block, we have shown that the model can better stay "on track", describing the image content without drifting away to unrelated yet common phrases. In addition, we explore different types of length normalization for beam search in order to prevent a bias towards very short sentences, which further improves the results. The proposed method achieves state-of-the-art performance on various benchmark datasets. Moreover, our key contributions are, to a large extent, complementary to key aspects of other methods, such as attention mechanisms [38] or model ensembles [37], indicating that further improvements on performance may be obtained by integrating these schemes.

	B@1	B@2	B@3	B@4	METEOR	CIDEr
multimodal RNN [17]	62.5	45.0	32.1	23.0	19.5	66
Google NIC [37]	66.6	46.1	32.9	24.6	—	—
LRCN-CaffeNet [7]	62.8	44.2	30.4	—	—	—
m-RNN [26]	67	49	35	25	—	—
Soft-Attention [38]	70.7	49.2	34.4	24.3	**23.9**	—
Hard-Attention [38]	**71.8**	**50.4**	35.7	25.0	23.04	—
emb-gLSTM, Polynomial	63.8	46.3	33.6	24.8	23.33	79.03
emb-gLSTM, Min-hinge	66.3	48.5	35.4	26.2	22.95	**81.26**
emb-gLSTM, Gaussian	67.0	49.1	**35.8**	**26.4**	22.74	81.25

Table 5: Comparison with state-of-the-art methods on MS COCO.

7. Acknowledgment

The authors acknowledge the support of the IWT-SBO project PARIS and the iMinds project HiViz.

References

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. In *ICLR*, 2015. 1, 2, 3, 4

[2] S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python*. O'Reilly, 2009. 6

[3] X. Chen and C. L. Zitnick. Mind's eye:a recurrent visual representation for image caption generation. In *CVPR*, 2015. 6

[4] K. Cho, B. van Merrienboer, D. Bahdanau, and Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. In *Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (SSST-8)*, 2014. 4, 5

[5] K. Cho, B. van Merrienboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *EMNLP*, 2014. 1, 2, 3

[6] M. Denkowski and A. Lavie. Meteor universal: Language specific translation evaluation for any target language. In *EACL 2014 Workshop on Statistical Machine Translation*, 2014. 6

[7] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 1, 2, 3, 8

[8] H. Fang, S. Gupta, F. N. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. In *CVPR*, 2015. 6

[9] A. Farhadi, S. M. M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. A. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV (4)*, 2010. 1, 2

[10] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *IJCV*, 106(2):210–233, 2014. 4, 6

[11] A. Graves. Sequence transduction with recurrent neural networks. *CoRR*, abs/1211.3711, 2012. 5

[12] A. Graves. Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850, 2013. 5

[13] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber. LSTM: A search space odyssey. *CoRR*, abs/1503.04069, 2015. 3

[14] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997. 2, 3

[15] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, 47:853–899, 2013. 6

[16] H. Hotelling. Relations between two sets of variates. *Biometrika*, pages 321–377, 1936. 4

[17] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 1, 2, 3, 6, 8

[18] A. Karpathy, A. Joulin, and F. Li. Deep fragment embeddings for bidirectional image sentence mapping. In *NIPS*, 2014. 6

[19] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Multimodal neural language models. In *ICML*, 2014. 1, 2, 8

[20] G. Kulkarni, V. Premraj, V. Ordonez, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Babytalk: Understanding and generating simple image descriptions. *TPAMI*, 35(12):2891–2903, 2013. 1, 2

[21] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012. 1, 2

[22] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Generalizing image captions for image-text parallel corpus. In *ACL*, 2013. 1, 2

[23] P. Kuznetsova, V. Ordonez, T. Berg, and Y. Choi. Treetalk: Composition and compression of trees for image descriptions. *TACL*, 2:351–362, 2014. 1, 2

[24] A. Lavie and A. Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Second Workshop on Statistical Machine Translation*, 2007. 6

[25] T. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. In *ECCV*, 2014. 6

[26] J. Mao, W. Xu, Y. Yang, J. Wang, and A. L. Yuille. Deep captioning with multimodal recurrent neural networks (m-rnn). In *ICLR*, 2015. 1, 2, 3, 8

[27] R. Mason and E. Charniak. Nonparametric method for data-driven image captioning. In *ACL*, 2014. 1, 2

[28] M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Stratos, X. Han, A. Mensch, A. C. Berg, T. L. Berg, and H. D. III. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012. 1, 2

[29] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 6

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 6, 8

[31] I. Sutskever, J. Martens, and G. Hinton. Generating text with recurrent neural networks. In *ICML*, 2011. 2

[32] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014. 1, 2, 3, 5

[33] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2014. 8

[34] T. Tieleman and G. Hinton. Leccture 6.5 - rmsprop. Technical Report MSU-CSE-00-2, 2000. 6

[35] A. Vedaldi and K. Lenc. Matconvnet – convolutional neural networks for matlab. *CoRR*, abs/1412.4564, 2014. 6

[36] R. Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, 2015. 6

[37] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015. 1, 2, 3, 4, 6, 8

[38] K. Xu, J. Ba, R. Kiros, K. Cho, A. C. Courville, R. Salakhutdinov, R. S. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 1, 2, 3, 6, 8

[39] Y. Yang, C. L. Teo, H. D. III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011. 1, 2

[40] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2:67–78, 2014. 6