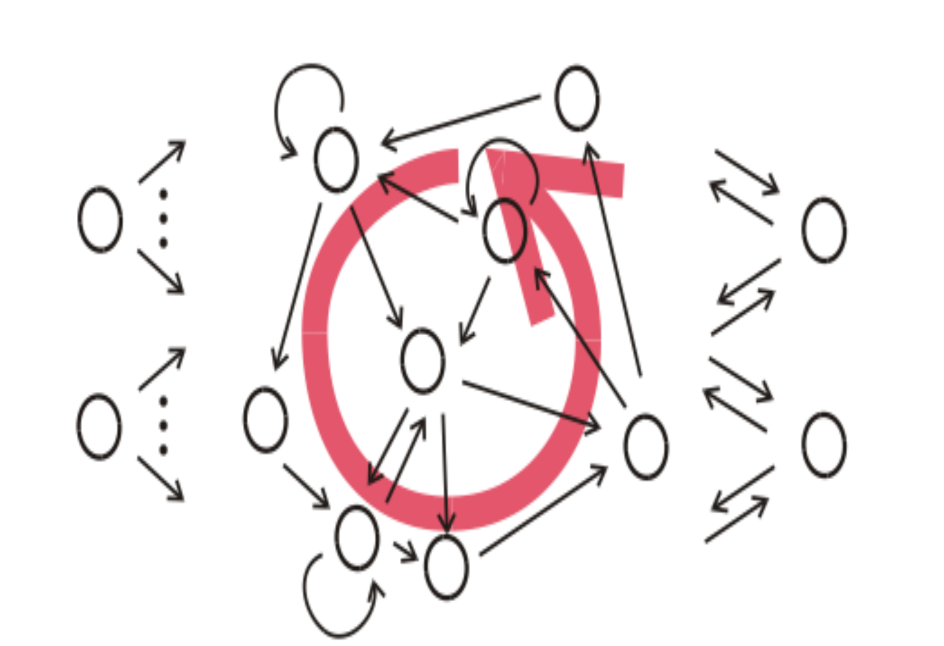
RNNs for Image Caption Generation

James Guevara

Recurrent Neural Networks

- Contain at least one directed cycle.
- Applications include: pattern classification, stochastic sequence modeling, speech recognition.
- Train using backpropagation through time.



Backpropagation Through Time

- “Unfold the neural network in time by stacking identical copies.
- Redirect connections within the network to obtain connections between subsequent copies.
- The gradient vanishes as errors propagate in time.

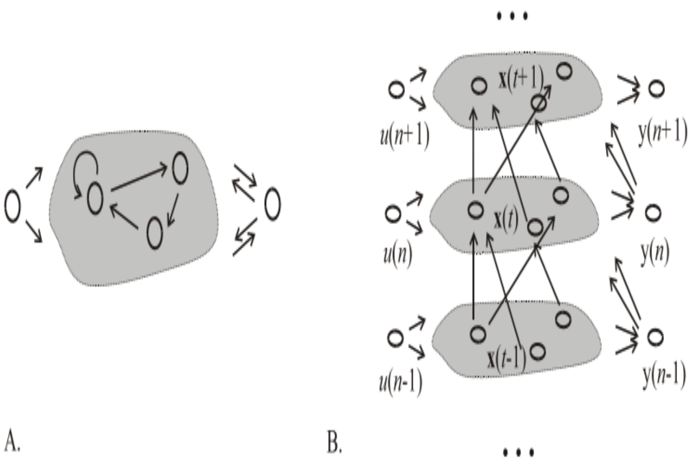
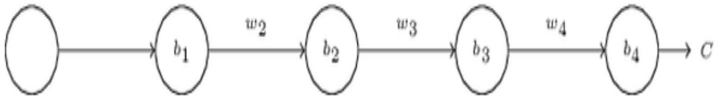


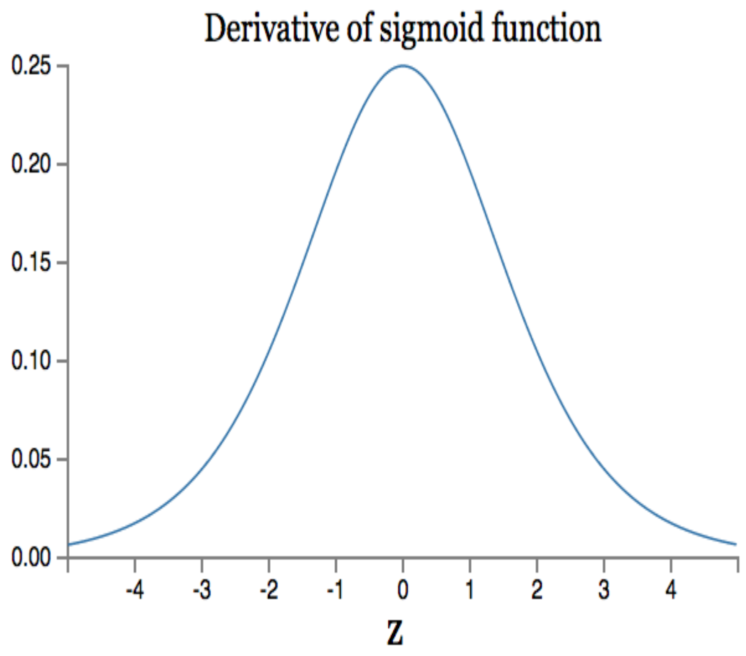
Figure 2.1: Schema of the basic idea of BPTT. A: the original RNN. B: The feedforward network obtained from it. The case of single-channel input and output is shown.

Vanishing Gradient Problem

- Derivative of sigmoid function peaks at .25.

$$\frac{\partial C}{\partial b_1} = \sigma'(z_1) \times w_2 \times \sigma'(z_2) \times w_3 \times \sigma'(z_3) \times w_4 \times \sigma'(z_4) \times \frac{\partial C}{\partial a_4}$$





Motivation

A good image description is often said to “paint a picture in your mind’s eye.”

- Bi-directional mapping between images and their descriptions (sentences).
  - Novel descriptions from images.
  - Visual representations from descriptions.
- As a word is generated or read, the visual representation is updated to reflect the new information contained in the word.
- The hidden layers, which are learned by “translating” between multiple modalities, can discover rich structures in data and learn long distance relations in an automatic, data-driven way.

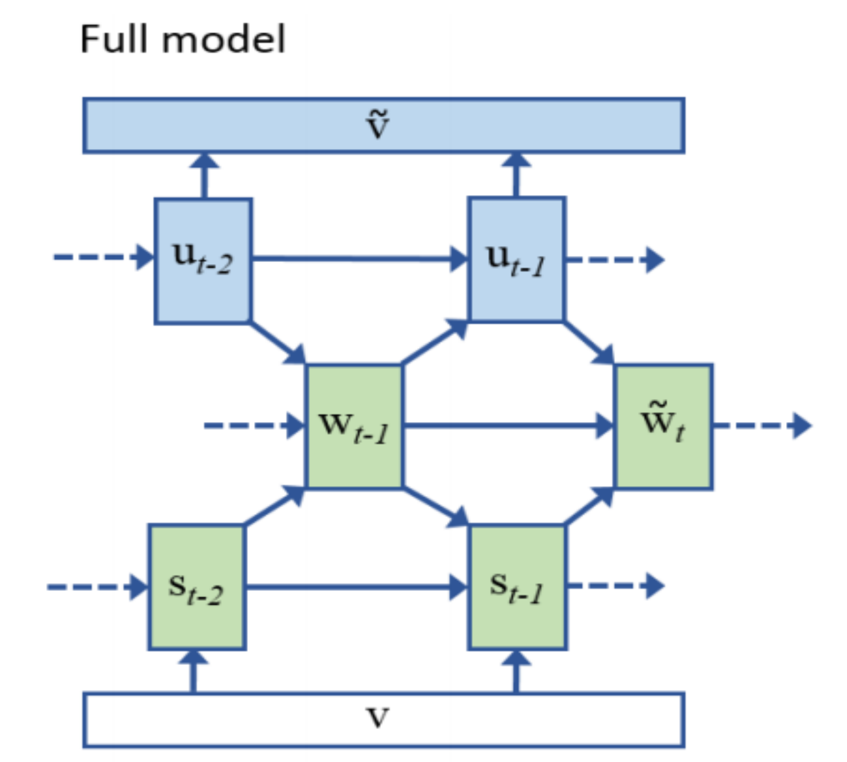
Goals

1. Compute probability of word w_t being generated at time t given a set of previously generated words $W_{t-1} = w_1, \dots, w_{t-1}$ and visual features V, i.e. $P(w_t \mid V, W_{t-1}, U_{t-1})$.
2. Compute likelihood of visual features V given a set of spoken or read words W_t in order to generate a visual representation of the scene or for performing image search, i.e. $P(V \mid W_{t-1}, U_{t-1})$.

Thus, we want to maximize $P(w_t, V \mid W_{t-1}, U_{t-1})$.

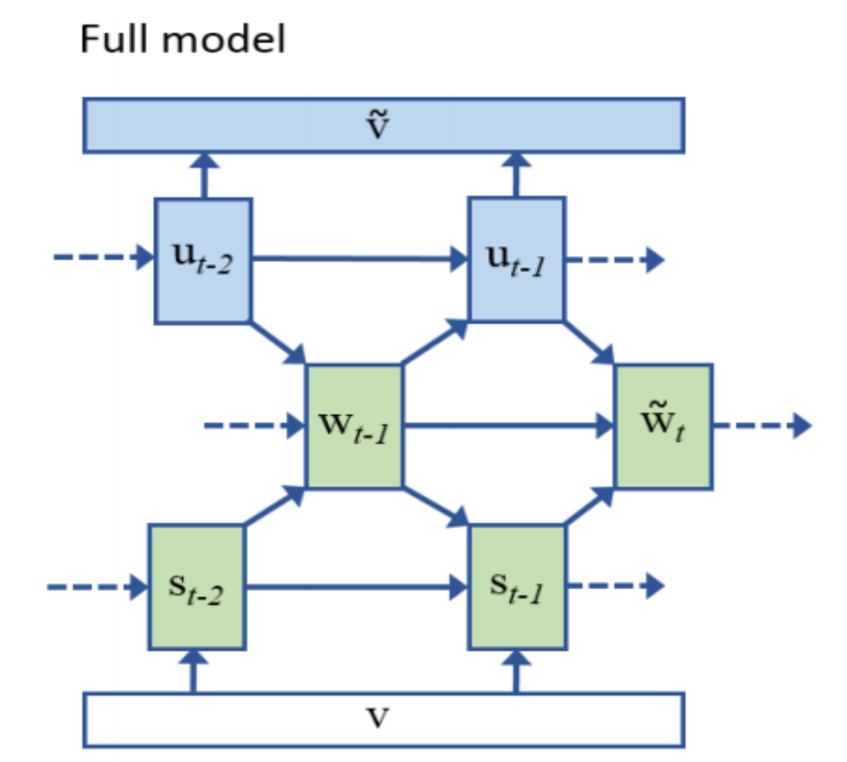
Approach

- Builds on previous model (shown by green boxes).
- The word at time t is represented by a vector w_t using a “one hot” representation (the size of the vector is the size of the vocabulary).
- The output contains likelihood of generating each word.



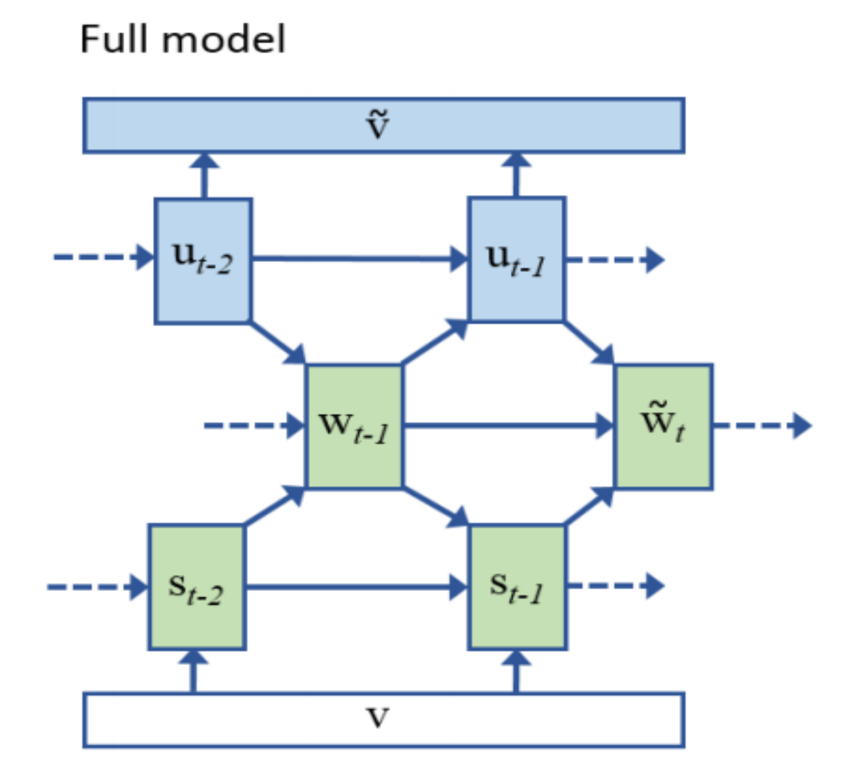
Approach

- Recurrent hidden state s provides context based on previous words, but can only model short-range interactions due to vanishing gradient).
- Another paper added an input layer V, which may represent a variety of static information.
- V helps with selection of words (e.g. if a cat is detected visually, then the likelihood of outputting the word “cat” increases).



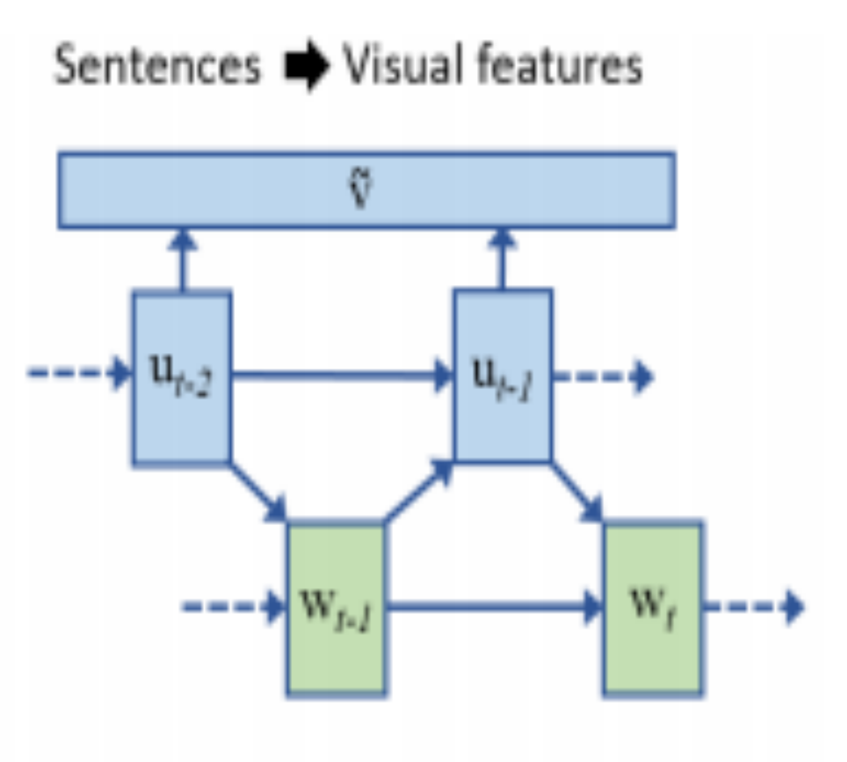
Approach

- Main contribution of this paper is visual hidden layer u, which attempts to reconstruct visual features v from previous words, i. e. v ~ v.
- Visual hidden layer is also used by w_t to predict next word.
- Force u to estimate v at every time step => long-term memory.



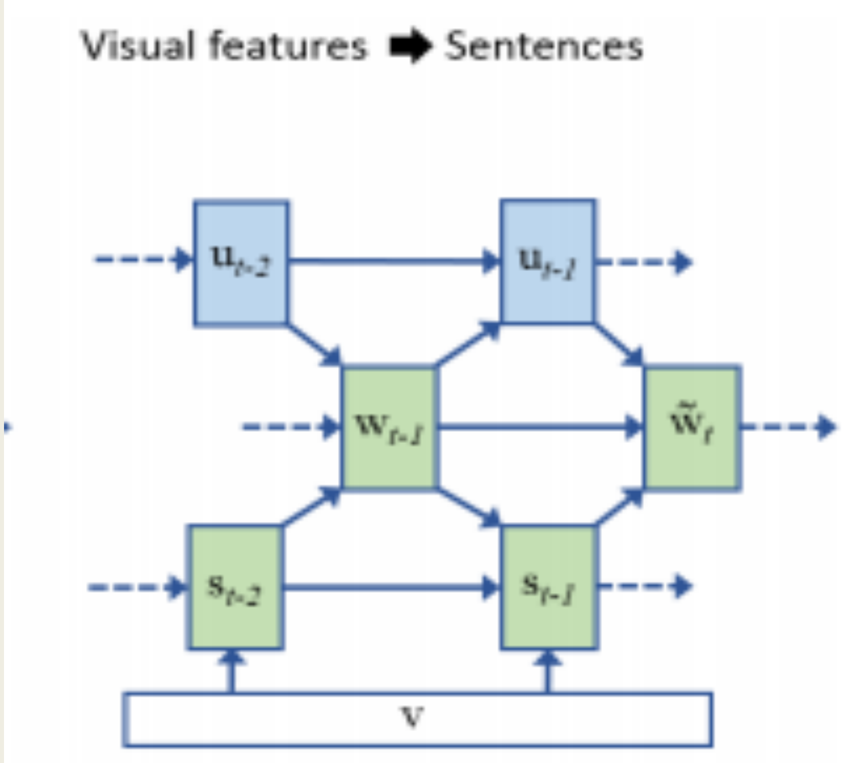
Approach

- Same network structure can predict visual features from sentences, or generate sentences from visual features.
- For predicting visual features from sentences, w is known, and s and v may be ignored.



Approach

- Same network structure can predict visual features from sentences, or generate sentences from visual features.
- For predicting visual features from sentences, w is known, and s and v may be ignored.
- For generating sentences, v is known and v (tilda) may be ignored.



Hidden Unit Activations

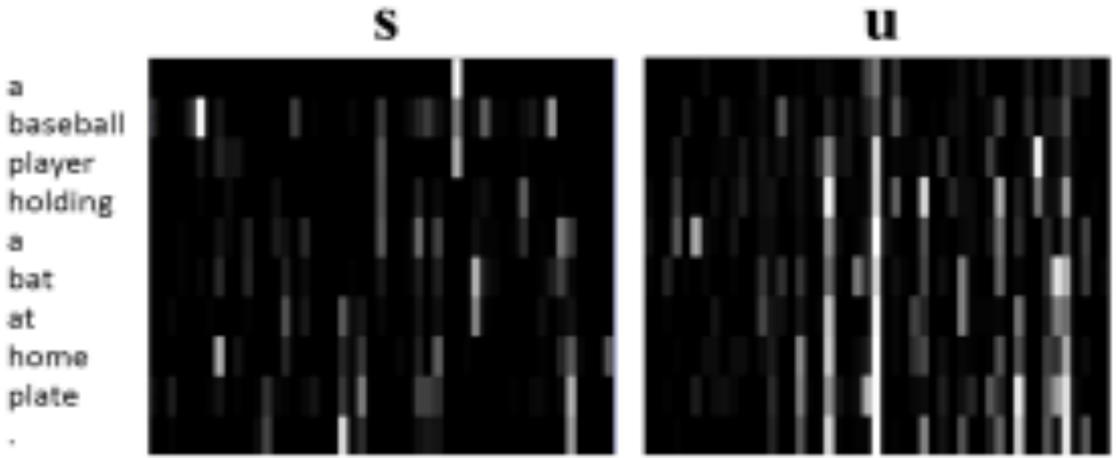


Figure 2. Illustration of the hidden units s and u activations through time (vertical axis). Notice that the visual hidden units u exhibit long-term memory through the temporal stability of some units, where the hidden units s change significantly each time step.

Language Model

- Language model typically has between 3,000 and 20,000 words.
- Use “word classing”:
  - $P\,(w_t \mid \bullet) = P(c_t \mid \bullet) * P(w_t \mid c_t, \bullet)$
  - $P\,(w_t \mid \bullet)$ is the probability of the word.
  - $P(c_t \mid \bullet)$ is the probability of the class.
  - Class label of the word is computed in unsupervised manner, grouping words of similar frequencies together.
  - Predicted word likelihoods are computed using soft-max function.
- To further reduce perplexity, combine RNN model’s output with the output from a Maximum Entropy model, simultaneously learned from the training corpus.
- For all experiments, fix how many words to look back when predicting the next word used by the ME model to three.
- Pre-processing: tokenize the sentences and lower case all the letter

Learning

- Backpropagation Through Time.
  - The network is unrolled for several words and BPTT is applied.
  - Reset the model after an EOS (End-of-Sentence) is encountered.
- Use online learning for the weights from the recurrent units to the output words.
- The weights for the rest of the network use a once per sentence batch update.
- Word predictions use soft-max function, the activations for the rest of the units use the sigmoid function.
- Combine open source RNN code with a Caffe framework.
  - Jointly learn word and image representations, i.e. the error from predicting the words can directly propage to the image-level features.
  - Fine-tune from pre-trained 1000-class ImageNet model to avoid potential over-fitting

Results

- Evaluate performance on both sentence retrieval and image retrieval.
- Datasets used in evaluation: PASCAL 1K, Flickr 8K and 30K, MS COCO.
- Hidden layers s and u sizes are fixed to 100.
- Compared final model with three RNN baselines
  - RNN based Language Model - basic RNN with no input visual features.
  - RNN with Image Features (RNN + IF).
  - RNN with Image Features Fine-Tuned - same as RNN + IF, but error is back-propagated to the CNN. CNN is initialized with the weights from the BVLC reference net. RNN is pre-trained.

Sentence Generation

- To generate a sentence:
  - Sample a target sentence length from the multinomial distribution of lengths learned from the training data.
  - For this fixed length, sample 100 random sentences.
  - Use the one with the lowest loss (negative likelihood and reconstruction error) as output.
- Three automatic metrics: PPL (perplexity), BLEU, METEOR.
  - PPL measures the likelihood of generating the testing sentence based on the number of bits it would take to encode it. (the lower the better)
  - BLEU and METEOR rate quality of translated sentences given several reference sentences. (the higher the better)

Sentence Generation (Results)

	PASCAL			Flickr 8K			Flickr 30K			MS COCO		
	PPL	BLEU	METR	PPL	BLEU	METR	PPL	BLEU	METR	PPL	BLEU	METR
Midge [28]	-	2.89	8.80					-				
Baby Talk [19]	-	0.49	9.69					-				
RNN	36.79	2.79	10.08	21.88	4.86	11.81	26.94	6.29	12.34	18.96	4.63	11.47
RNN+IF	30.04	10.16	16.43	20.43	12.04	17.10	23.74	10.59	15.56	15.39	16.60	19.24
RNN+IF+FT	29.43	10.18	16.45	-	-	-	-	-	-	14.90	16.77	19.41
Our Approach	27.97	10.48	16.69	19.24	14.10	17.97	22.51	12.60	16.42	14.23	18.35	20.04
Our Approach + FT	26.95	10.77	16.87	-	-	-	-	-	-	13.98	18.99	20.42
Human	-	22.07	25.80	-	22.51	26.31	-	19.62	23.76	-	20.19	24.94

Table 1. Results for novel sentence generation for PASCAL 1K, Flickr 8K, FLickr 30K and MS COCO. Results are measured using perplexity (PPL), BLEU (%) [30] and METEOR (METR, %) [1]. When available results for Midge [28] and BabyTalk [19] are provided. Human agreement scores are shown in the last row. See the text for more details.

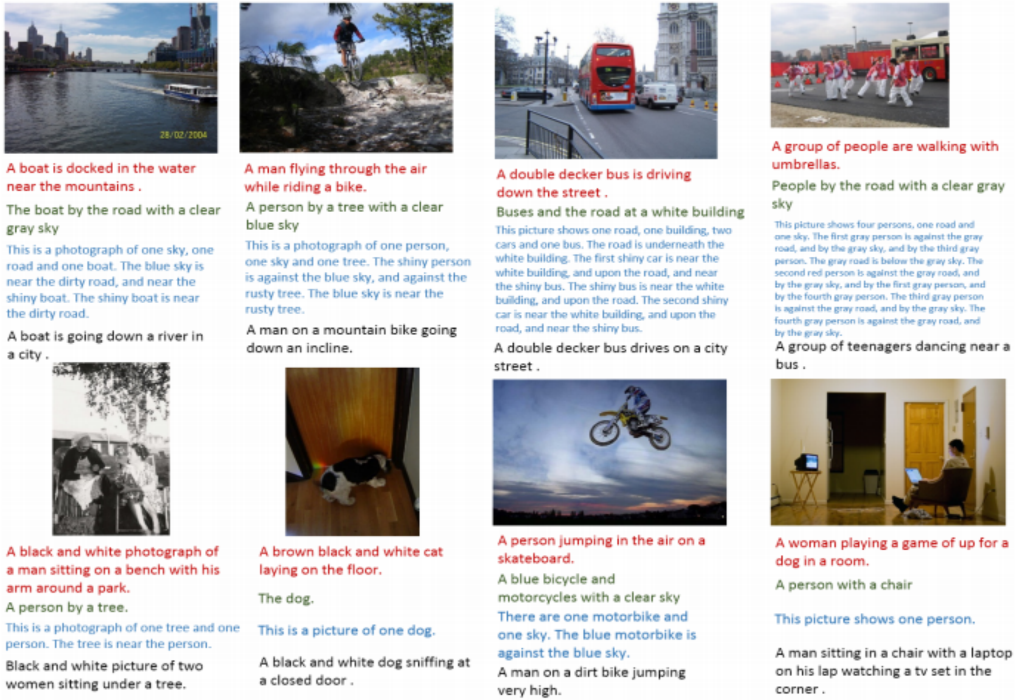


Figure 4. Qualitative results for sentence generation on the PASCAL 1K dataset. Generated sentences are shown for our approach (red), Midge [28] (green) and BabyTalk [19] (blue). For reference, a human generated caption is shown in black.

MS COCO Qualitative Results

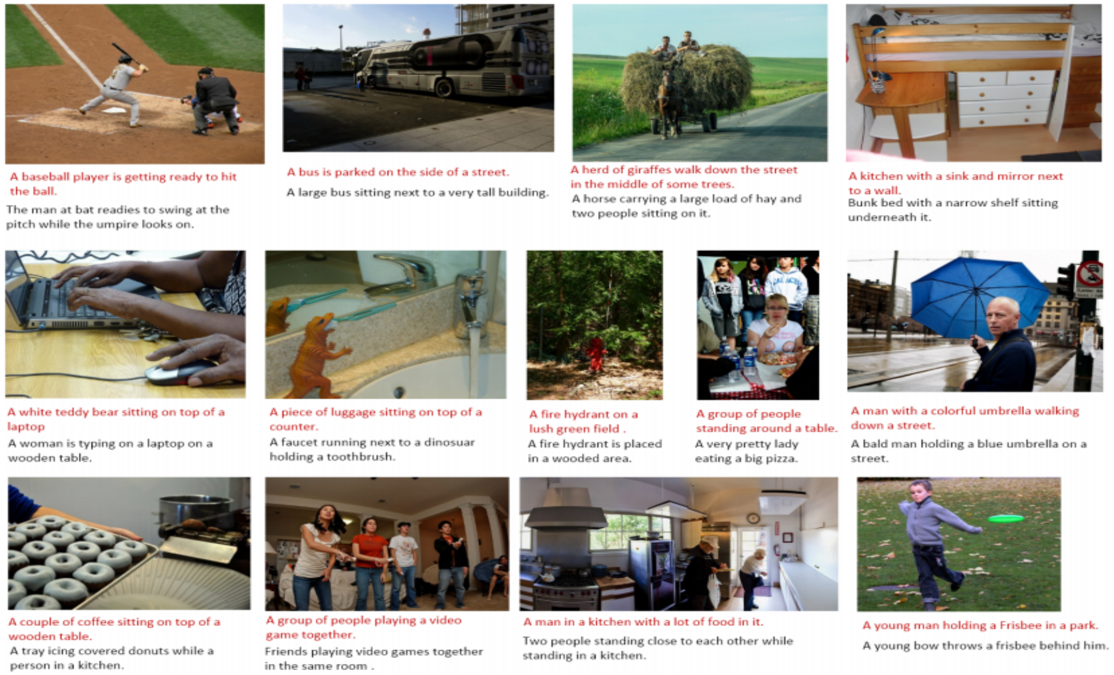


Figure 3. Qualitative results for sentence generation on the MS COCO dataset. Both a generated sentence (red) using (Our Approach + FT) and a human generated caption (black) are shown.

MS COCO Quantitative Results

- BLEU and METEOR scores (18.99 & 20.42) slightly lower than human scores (20.19 & 24.94).
- BLEU-1 to BLEU-4 scores: 60.4%, 26.4%, 12.6%, and 6.4%.
  - Human scores: 65.9%, 30.5%, 13.6%, and 6.0%.

“It is known that automatic measures are only roughly correlated with human judgment.”

- Asked 5 human subjects to judge whether generated sentence was better than human generated ground truth caption.
- 12.6% and 19.8% prefer automatically generated captions to the human captions without and with fine-tuning.
- Less than 1% of subjects rated captions the same.

Bi-directional Retrieval

- For each retrieval task, there are two methods for ranking:
  - Rank based on likelihood of the sentence given the image (T).
  - Rank based on reconstruction error between image’s visual features v and their reconstructed features v (I).
- Two protocols for using multiple image descriptions:
  - Treat each of the 5 sentences individually. The rank of the retrieved ground truth sentences are used for evaluation.
  - Treat all sentences as a single annotation, and concatenate them together for retrieval.
- Evaluation metric: R@K (K = 1,5,10)
  - Recall rates of the (first) ground truth sentences or images, depending on task at hand.
  - Higher R@K corresponds to better retrieval performance.
- Evaluation metric: Med/Mean r
  - median/mean rank of the (first) retrieved ground truth sentences or images.
  - Lower the better.

	Sentence Retrieval				Image Retrieval			
	R@1	R@5	R@10	Med r	R@1	R@5	R@10	Med r
Random Ranking	0.1	0.6	1.1	631	0.1	0.5	1.0	500
[32]	4.5	18.0	28.6	32	6.1	18.5	29.0	29
DeViSE [8]	4.8	16.5	27.3	28	5.9	20.1	29.6	29
DeepFE [16]	12.6	32.9	44.0	14	9.7	29.6	42.5	15
DeepFE+DECAF [16]	5.9	19.2	27.3	34	5.2	17.6	26.5	32
RNN+IF	7.2	18.7	28.7	30.5	4.5	15.34	24.0	39
Our Approach (T)	7.6	21.1	31.8	27	5.0	17.6	27.4	33
Our Approach (T+I)	7.7	21.0	31.7	26.6	5.2	17.5	27.9	31
[13]	8.3	21.6	30.3	34	7.6	20.7	30.1	38
RNN+IF	5.5	17.0	27.2	28	5.0	15.0	23.9	39.5
Our Approach (T)	6.0	19.4	31.1	26	5.3	17.5	28.5	33
Our Approach (T+I)	6.2	19.3	32.1	24	5.7	18.1	28.4	31
M-RNN [23]	14.5	37.2	48.5	11	11.5	31.0	42.4	15
RNN+IF	10.4	30.9	44.2	14	10.2	28.0	40.6	16
Our Approach (T)	11.6	33.8	47.3	11.5	11.4	31.8	45.8	12.5
Our Approach (T+I)	11.7	34.8	48.6	11.2	11.4	32.0	46.2	11

Table 3. Flickr 8K Retrieval Experiments. The protocols of [32], [13] and [23] are used respectively in each row. See text for details.

	Sentence Retrieval				Image Retrieval			
	R@1	R@5	R@10	Med r	R@1	R@5	R@10	Med r
Random Ranking	0.1	0.6	1.1	631	0.1	0.5	1.0	500
DeViSE [8]	4.5	18.1	29.2	26	6.7	21.9	32.7	25
DeepFE+FT [16]	16.4	40.2	54.7	8	10.3	31.4	44.5	13
RNN+IF	8.0	19.4	27.6	37	5.1	14.8	22.8	47
Our Approach (T)	9.3	23.8	24.0	28	6.0	17.7	27.0	35
Our Approach (T+I)	9.6	24.0	27.2	25	7.1	17.9	29.0	31
M-RNN [23]	18.4	40.2	50.9	10	12.6	31.2	41.5	16
RNN+IF	9.5	29.3	42.4	15	9.2	27.1	36.6	21
Our Approach (T)	11.9	25.0	47.7	12	12.8	32.9	44.5	13
Our Approach (T+I)	12.1	27.8	47.8	11	12.7	33.1	44.9	12.5

Table 4. Flickr 30K Retrieval Experiments. The protocols of [8] and [23] are used respectively in each row. See text for details.