Image Captioning 문제소개

- **Image Captioning** : Image가 주어지면 그에 대한 적절한 설명을 만들어주는(Captioning) 문제

"man in black shirt is playing guitar."

"construction worker in orange safety vest is working on road."

"two young girls are playing with lego toy."

"boy is doing backflip on wakeboard."

"girl in pink dress is jumping in air."

"black and white dog jumps over bar."

"young girl in pink shirt is swinging on swing."

"man in blue wetsuit is surfing on wave."

Image Captioning Dataset 1 – MS-COCO

- **M**icro**S**oft-**C**ommon **O**bjects in **CO**ntext(MS-COCO) Dataset
- COCO 2017 train/val browser (**123,287 images**, 886,284 instances), **5 captions per image**
- http://cocodataset.org/index.htm

a cat lays on its back while sitting in someone's lap.
a person sitting on a couch with a cat laying on its back.
a cat is lying on its back in a man's lap.
a cat laying on a persons lap who is sitting on a couch
a cat sits with his belly up in a person's lap.

Image Captioning Dataset 2 – Flickr 8K, 30K

- Flickr 8K, 30K Dataset
- 8000 images, 30000 images, 5 captions per image
- http://cocodataset.org/index.htm

- A blue car drives down a mostly dirt road.
- A blue race car on an off road track
- Blue car on a small road with men in the background and a res car on a slightly higher road.
- Blue race car driving down narrow road
- Bright blue car going down a dirt road onlookers watch.

논문 리뷰 - Show and Tell: A Neural Image Caption Generator

- Vinyals, Oriol, et al. "Show and tell: A neural image caption generator." CVPR 2015.
- Image Captioning를 문제를 위한 Architecture 제안
- http://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Vinyals_Show_and_Tell_2015_CVPR_paper.pdf

논문 리뷰 - Show and Tell: A Neural Image Caption Generator

- Neural Image Caption(NIC) Generator 모델을 제안

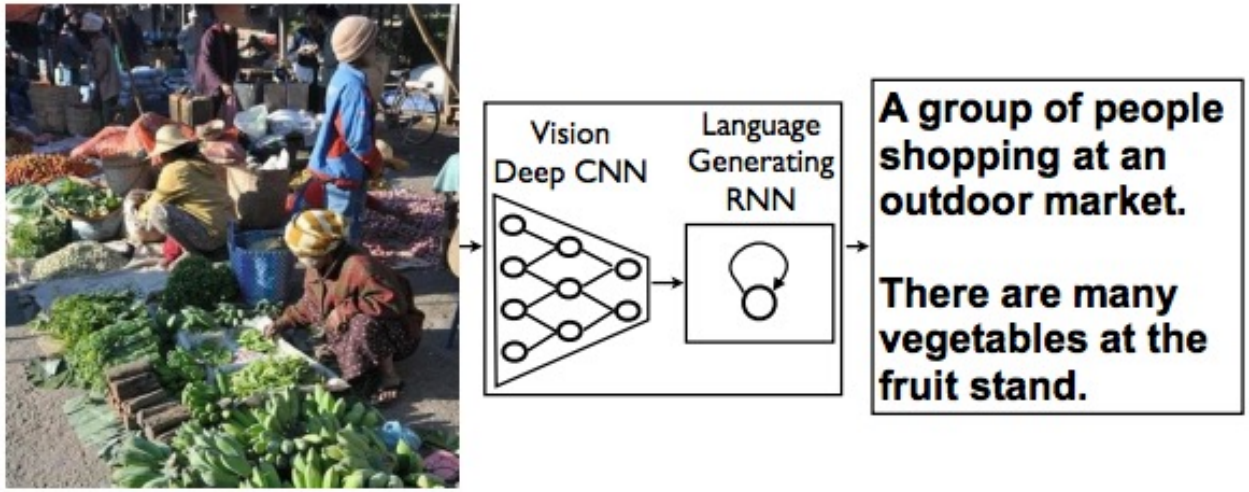


Figure 1. NIC, our model, is based end-to-end on a neural network consisting of a vision CNN followed by a language generating RNN. It generates complete sentences in natural language from an input image, as shown on the example above.

Problem Formulation

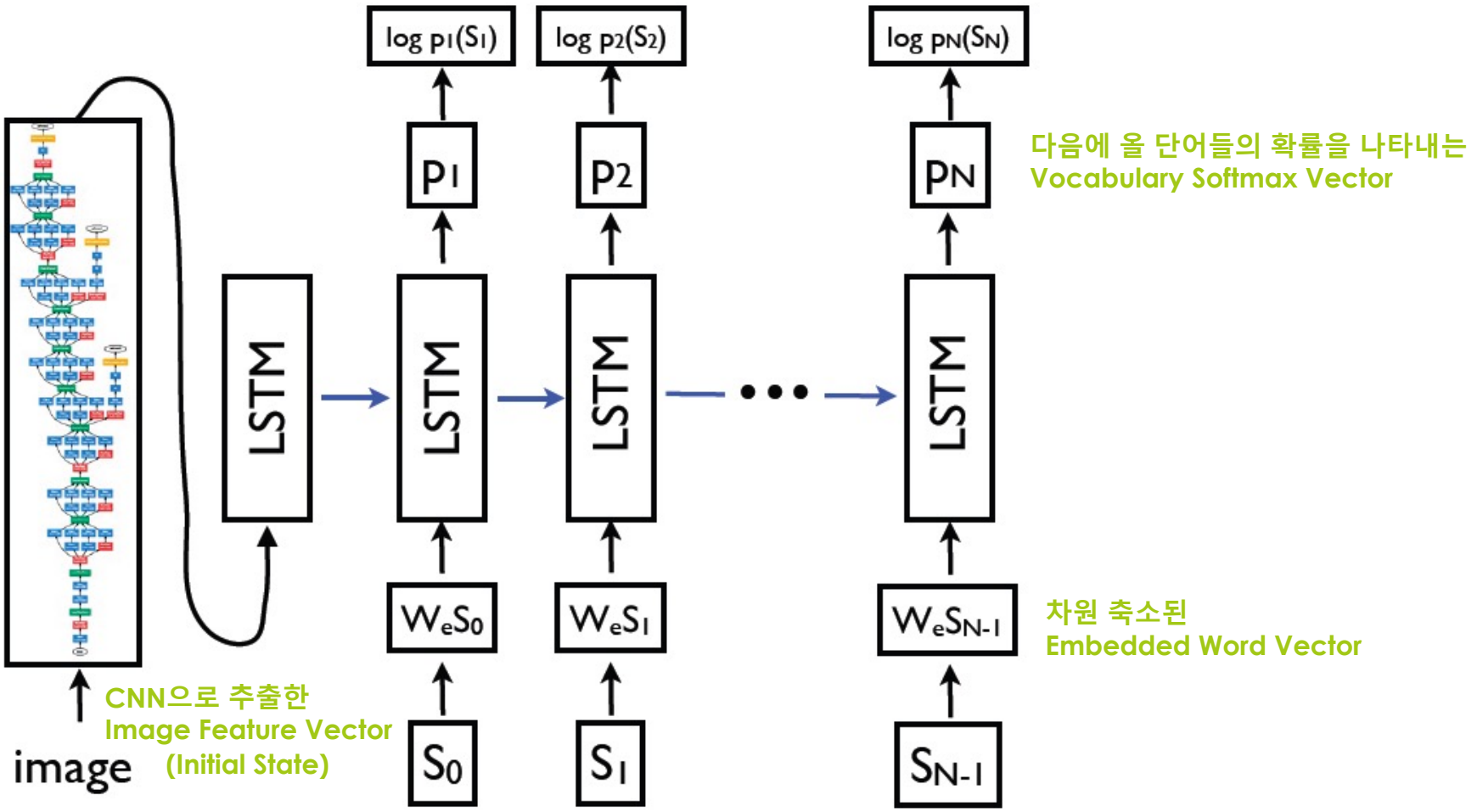
- Input Image I가 주어졌을때 이를 가장 잘 설명하는 문장 S를 생성해내는 파라미터 \theta(W, b)를 찾는 문제

$$\theta^{\star} = \arg\max_{\theta} \sum_{(I,S)} \log p(S|I;\theta) \qquad (1)$$

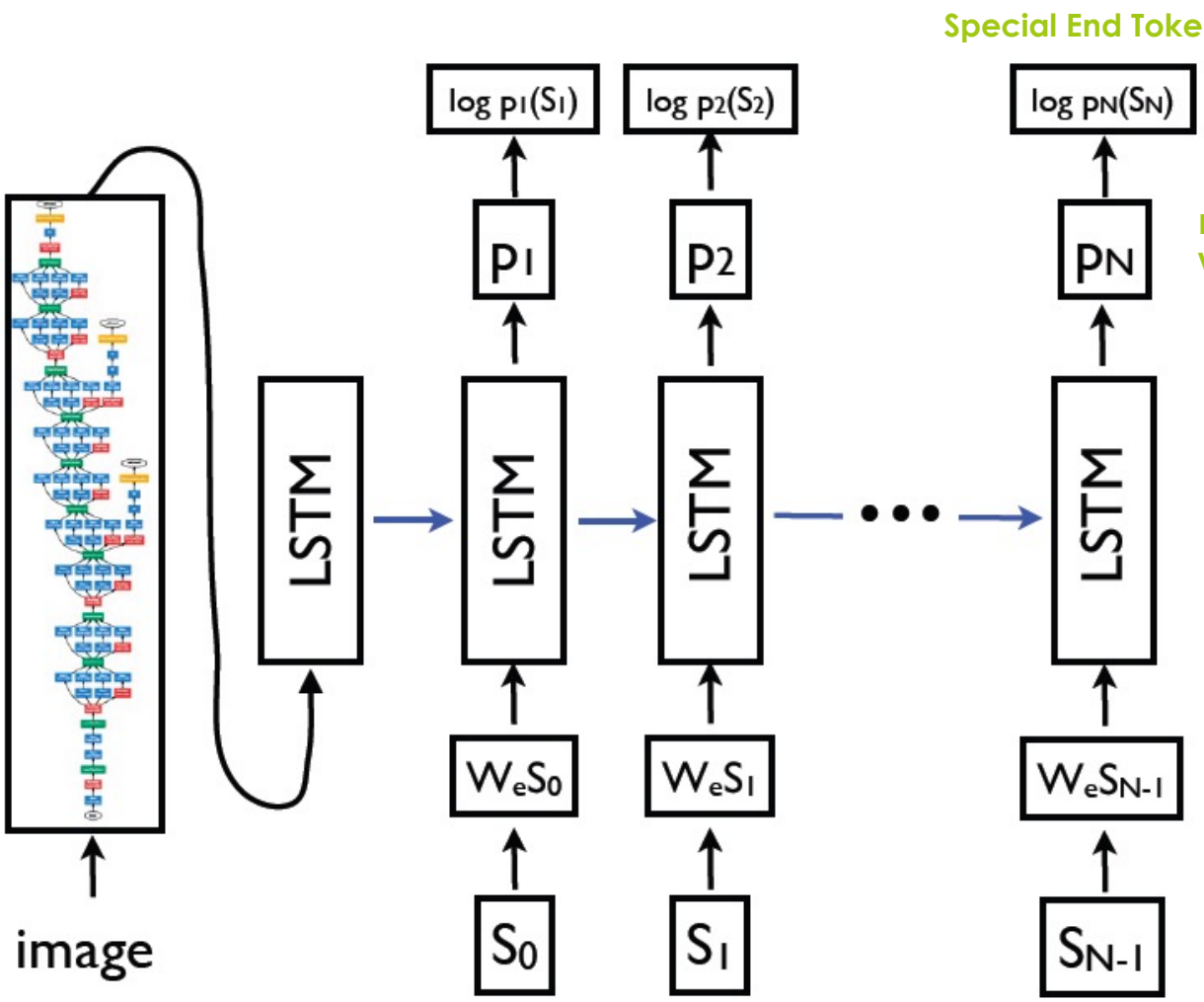
- 문장 S는 여러개의 단어로 이루어져 있고 현재 시간의 단어 S_t는 이전에 등장한 단어들(S_0,...,S_t-1)과 연관관계를 가지고 있다.

$$\log p(S|I) = \sum_{t=0}^{N} \log p(S_t|I, S_0, \ldots, S_{t-1}) \qquad (2)$$

NIC Architecture



Sampling & Beam Search



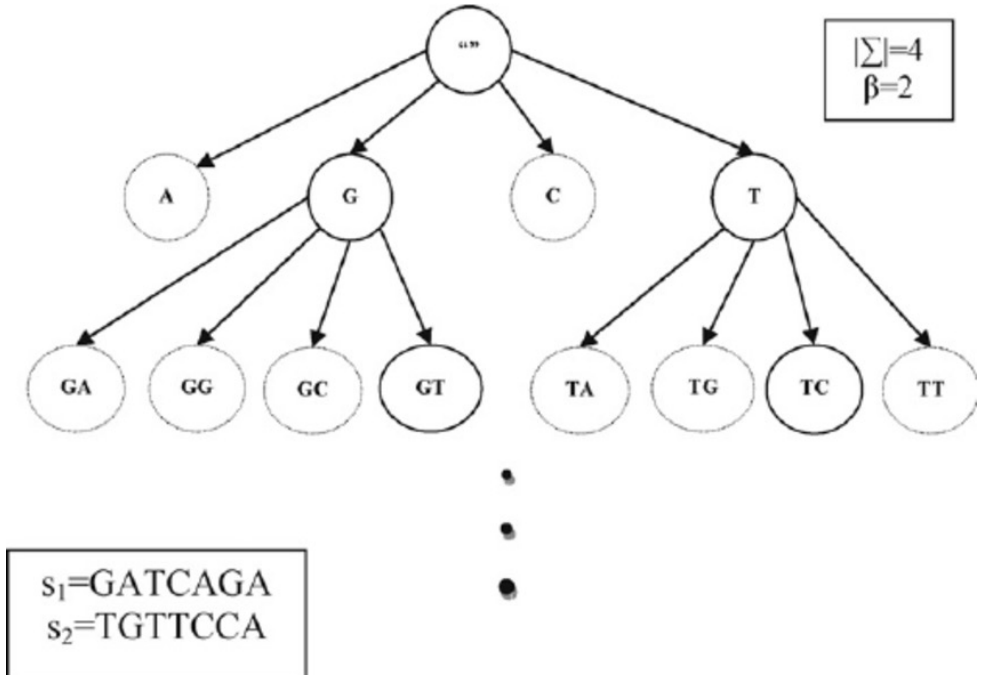
다음에 올 단어들의 확률을 나타내는 Vocabulary Softmax Vector

1. Sampling :
argmax를 취해서 다음에 올 단어중 가장 확률이 높은 1개의 단어를 취함

2. Beam Search :
다음에 올 단어 중 가장 확률이 높은 K개를 단어(Beam)를 유지하면서 탐색(Search)을 진행

Beam Search

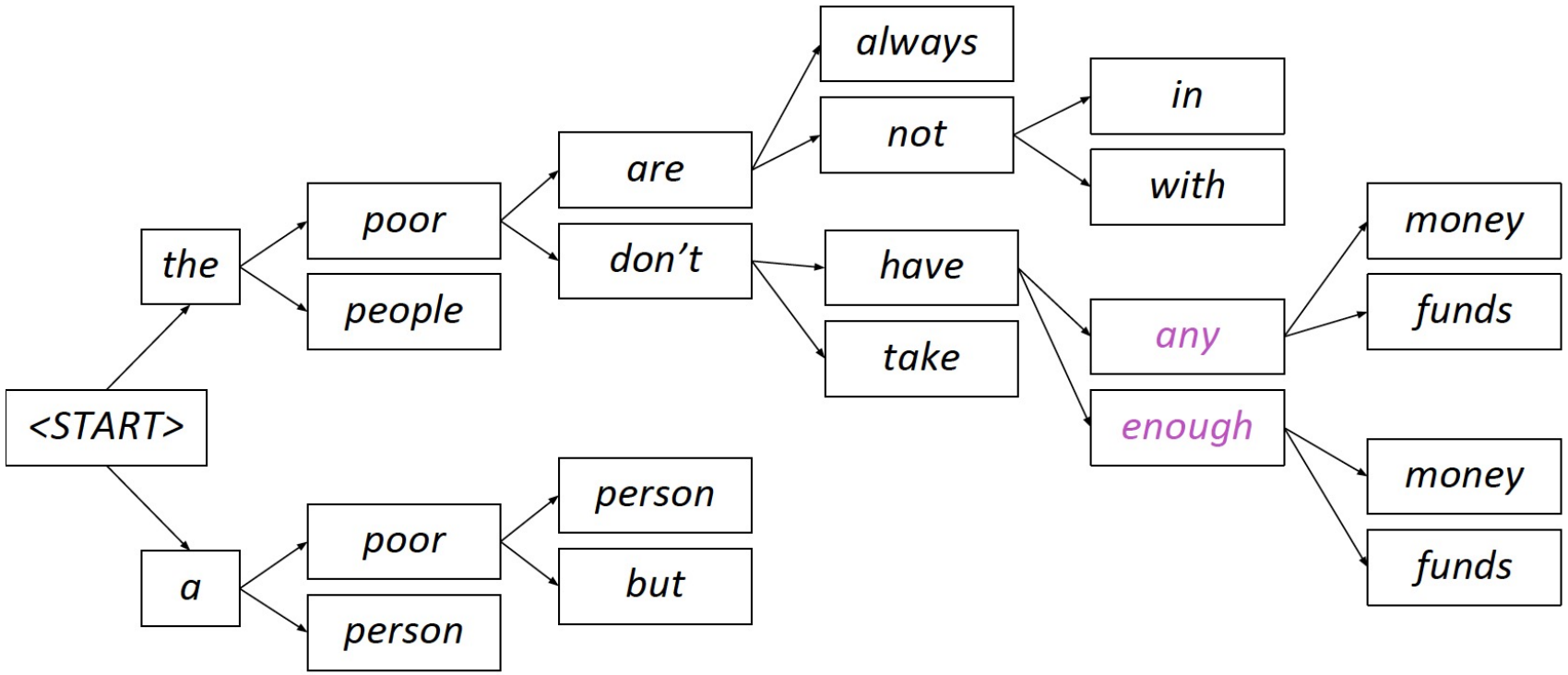
- 단순 Sampling보다 한단계 업그레이드 된 방법
- 가장 높은 확률을 가진 1개의 결과값이 아니라, **가장 높은 확률을 가진 k개의 결과값을 유지**하면서 이로 부터 다음 결과값을 탐색한다.



Beam Search

Beam search decoding: example

Beam size = 2



실험 환경 구성 & Experiment Result (정량적 평가)

Dataset name	size		
	train	valid.	test
Pascal VOC 2008 [6]	-	-	1000
Flickr8k [26]	6000	1000	1000
Flickr30k [33]	28000	1000	1000
MSCOCO [20]	82783	40504	40775
SBU [24]	1M	-	-

Metric	BLEU-4	METEOR	CIDER
NIC	**27.7**	**23.7**	**85.5**
Random	4.6	9.0	5.1
Nearest Neighbor	9.9	15.7	36.5
Human	21.7	25.2	85.4

Table 1. Scores on the MSCOCO development set.

Approach	PASCAL (xfer)	Flickr 30k	Flickr 8k	SBU
Im2Text [24]				11
TreeTalk [18]				19
BabyTalk [16]	25			
Tri5Sem [11]			48	
m-RNN [21]		55	58	
MNLM [14][5]		56	51	
SOTA	25	56	58	19
NIC	**59**	**66**	**63**	**28**
Human	69	68	70	

Table 2. BLEU-1 scores. We only report previous work results when available. SOTA stands for the current state-of-the-art.

Experiment Result (정량적 평가)

Approach	Image Annotation			Image Search		
	R@1	R@10	Med r	R@1	R@10	Med r
DeFrag [13]	13	44	14	10	43	15
m-RNN [21]	15	49	11	12	42	15
MNLM [14]	18	55	8	13	52	10
NIC	**20**	**61**	**6**	**19**	**64**	**5**

Table 4. Recall@k and median rank on Flickr8k.

Approach	Image Annotation			Image Search		
	R@1	R@10	Med r	R@1	R@10	Med r
DeFrag [13]	16	55	8	10	45	13
m-RNN [21]	18	51	10	13	42	16
MNLM [14]	**23**	**63**	**5**	**17**	**57**	**8**
NIC	17	56	7	**17**	**57**	**7**

Table 5. Recall@k and median rank on Flickr30k.

Experiment Result (정성적 평가)

- 원본 데이터에 없던 새로운 문장도 생성해냄

A man throwing a frisbee in a park. **A man holding a frisbee in his hand.** **A man standing in the grass with a frisbee.**
A close up of a sandwich on a plate. A close up of a plate of food with french fries. A white plate topped with a cut in half sandwich.
A display case filled with lots of donuts. **A display case filled with lots of cakes.** **A bakery display case filled with lots of donuts.**

Table 3. N-best examples from the MSCOCO test set. Bold lines indicate a novel sentence not present in the training set.

Experiment Result (정성적 평가)

Figure 5. A selection of evaluation results, grouped by human rating.

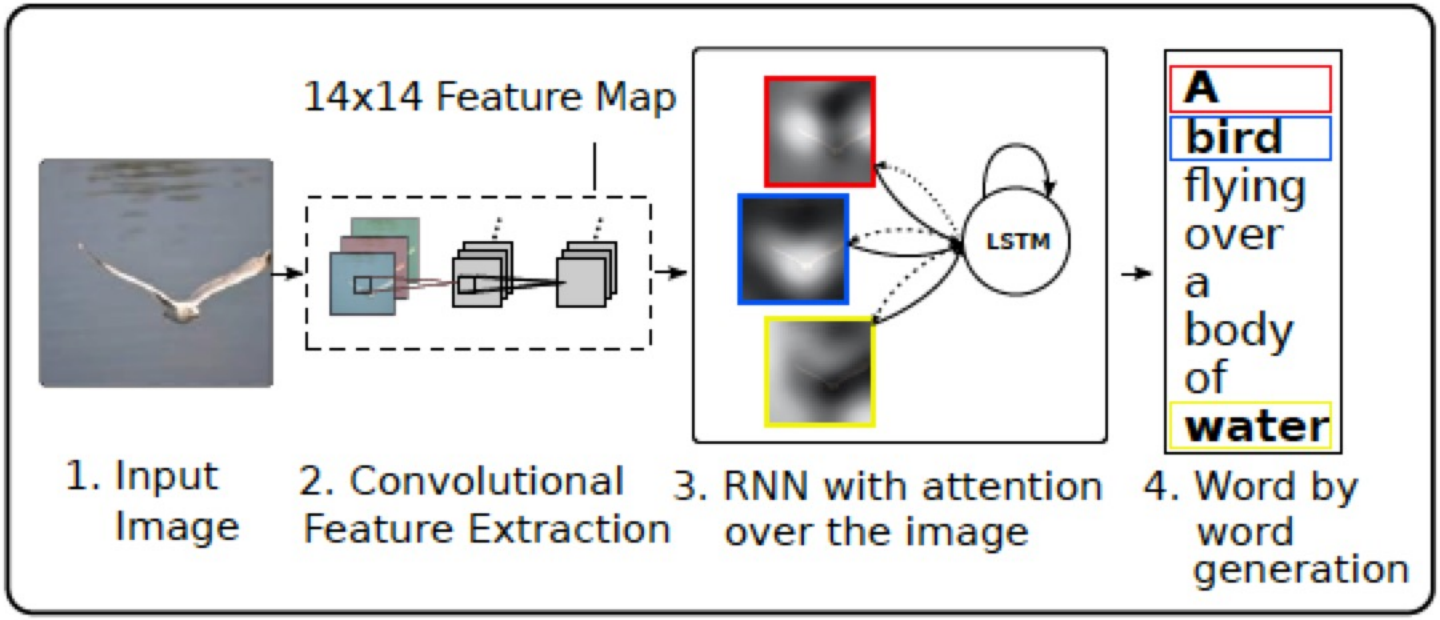
논문 리뷰 - Show, attend and tell: Neural image caption generation with visual attention

- Xu, Kelvin, et al. "Show, attend and tell: Neural image caption generation with visual attention." *International conference on machine learning*. 2015.
- Image Captioning를 문제를 위한 Attention Architecture 제안
- http://proceedings.mlr.press/v37/xuc15.pdf

논문 리뷰 - Show, attend and tell: Neural image caption generation with visual attention

- Attention을 결합한 이용한 Image Captioning 모델을 제안

Figure 1. Our model learns a words/image alignment. The visualized attentional maps (3) are explained in section 3.1 & 5.4



ENCODER: CONVOLUTIONAL FEATURES

- The extractor produces L vectors, each of which is a D-dimensional representation corresponding to a part of the image.

$$a = \{\mathbf{a}_1, \ldots, \mathbf{a}_L\}, \ \mathbf{a}_i \in \mathbb{R}^D$$

- In order to obtain a correspondence between the feature vectors and portions of the 2-D image, we **extract features from a lower convolutional layer** unlike previous work which instead used a fully connected layer.

- This allows the **decoder to selectively focus on certain parts of an image** by selecting a subset of all the feature vectors.

DECODER: LONG SHORT-TERM MEMORY NETWORK

- In simple terms, the context vector $\hat{z}_t$ (equations (1)–(3)) is a dynamic representation of the relevant part of the image input at time t.

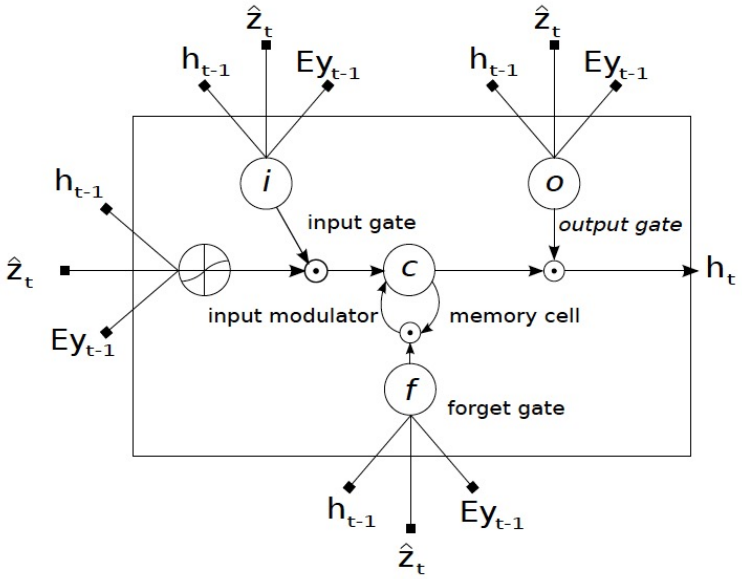


Figure 4. A LSTM cell, lines with bolded squares imply projections with a learnt weight vector. Each cell learns how to weigh its input components (input gate), while learning how to modulate that contribution to the memory (input modulator). It also learns weights which erase the memory cell (forget gate), and weights which control how this memory should be emitted (output gate).

DECODER: LONG SHORT-TERM MEMORY NETWORK

- The weight α_i of each annotation vector a_i is computed by an attention model f_{att} for which we use a multilayer perceptron conditioned on the previous hidden state h_{t-1}.

- The initial memory state and hidden state of the LSTM are predicted by an average of the annotation vectors fed through two separate MLPs (init,c and init,h):

$$\mathbf{c}_0 = f_{\text{init,c}}\left(\frac{1}{L}\sum_i^L \mathbf{a}_i\right)$$

$$\mathbf{h}_0 = f_{\text{init,h}}\left(\frac{1}{L}\sum_i^L \mathbf{a}_i\right)$$

- In this work, we use a deep output layer (Pascanu et al., 2014) to compute the **output word probability** given the LSTM state, the context vector and the previous word:

$$p(\mathbf{y}_t|\mathbf{a}, \mathbf{y}_1^{t-1}) \propto \exp(\mathbf{L}_o(\mathbf{E}\mathbf{y}_{t-1} + \mathbf{L}_h\mathbf{h}_t + \mathbf{L}_z\hat{\mathbf{z}}_t)) \quad (7)$$

DECODER: LONG SHORT-TERM MEMORY NETWORK

- The soft version of this attention mechanism was introduced by Bahdanau et al. (2014)

$$e_{ti} = f_{\text{att}}(\mathbf{a}_i, \mathbf{h}_{t-1}) \quad (4)$$

$$\alpha_{ti} = \frac{\exp(e_{ti})}{\sum_{k=1}^{L} \exp(e_{tk})}. \quad (5)$$

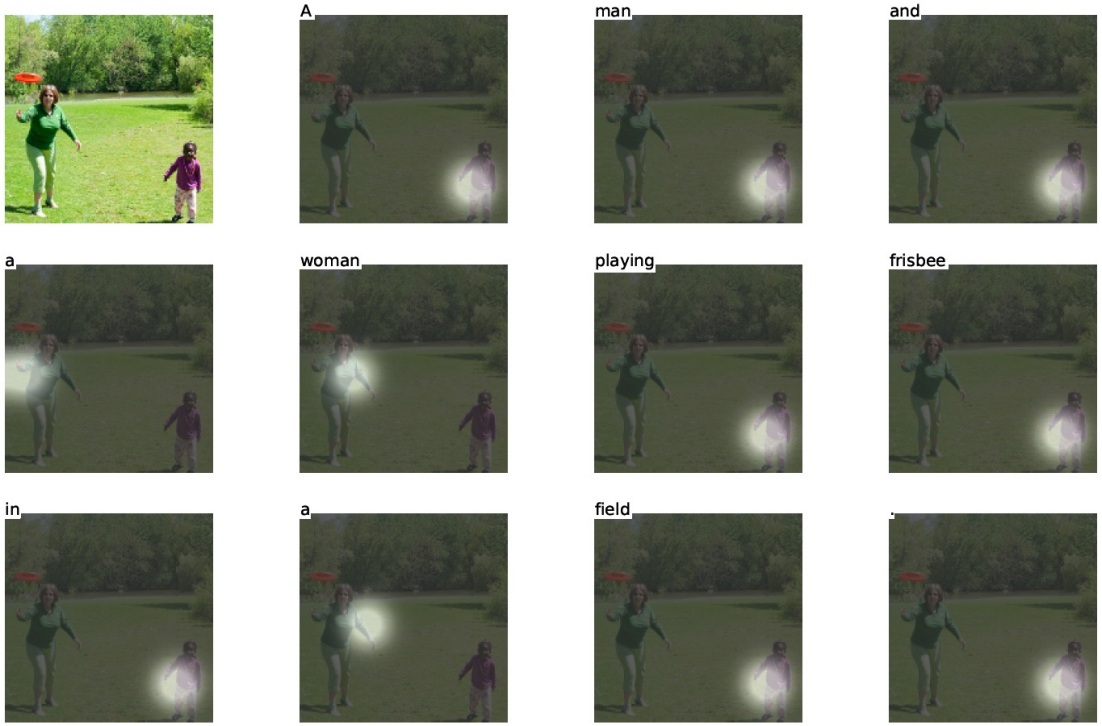
- Once the weights (which sum to one) are computed, the **context vector $\hat{z}_t$** is computed by

$$\hat{\mathbf{z}}_t = \phi\left(\{\mathbf{a}_i\}, \{\alpha_i\}\right), \quad (6)$$

- where ϕ is a function that returns a single vector given the set of annotation vectors and their corresponding weights. The details of ϕ function are discussed in Sec. 4.

Stochastic “Hard” Attention

- Policy Gradient(REINFORCE) Reinforcement Learning 방법론을 이용해서 이미지에서 집중해야하는 부분을 찾아냄



(a) A man and a woman playing frisbee in a field.

Deterministic "Soft" Attention

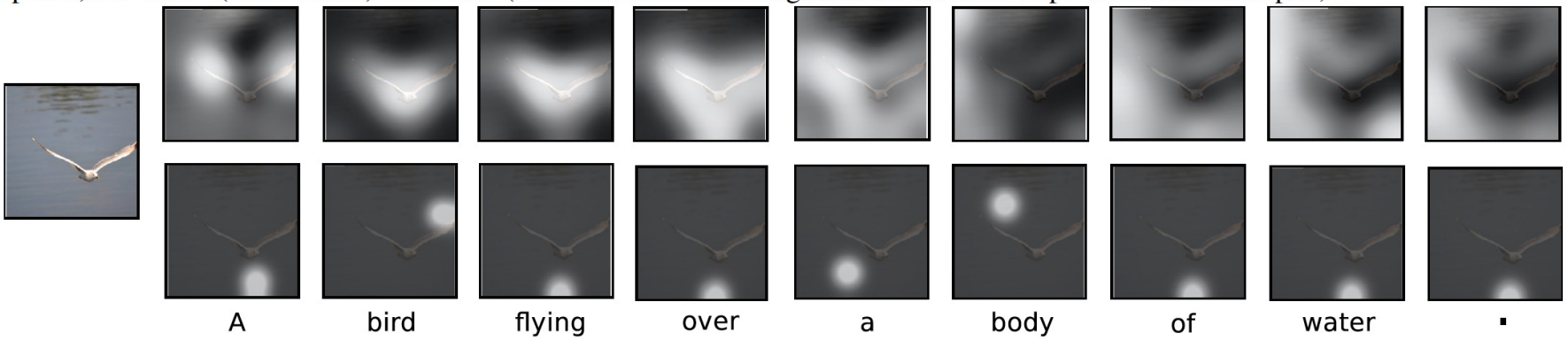
- Learning stochastic attention requires sampling the attention location s_t each time, instead we can take the expectation of the **context vector $\hat{z}_t$** directly,

$$\mathbb{E}_{p(s_t|a)}[\hat{\mathbf{z}}_t] = \sum_{i=1}^{L} \alpha_{t,i}\mathbf{a}_i \qquad (13)$$

- and formulate a deterministic attention model by computing a soft attention weighted annotation vector $\boldsymbol{\phi}(\{\boldsymbol{a_i}\},\{\boldsymbol{\alpha_i}\}) = \sum_i^L \boldsymbol{\alpha_i a_i}$ as introduced by Bahdanau et al. (2014).

Experiment Result

Figure 2. Attention over time. As the model generates each word, its attention changes to reflect the relevant parts of the image. "soft" (top row) vs "hard" (bottom row) attention. (Note that both models generated the same captions in this example.)



Experiment Result

Figure 3. Examples of attending to the correct object (*white* indicates the attended regions, *underlines* indicated the corresponding word)

A woman is throwing a frisbee in a park.

A dog is standing on a hardwood floor.

A stop sign is on a road with a mountain in the background.

A little girl sitting on a bed with a teddy bear.

A group of people sitting on a boat in the water.

A giraffe standing in a forest with trees in the background.

Experiment Result - mistakes

Figure 5. Examples of mistakes where we can use attention to gain intuition into what the model saw.

A large white bird standing in a forest.

A woman holding a clock in her hand.

A man wearing a hat and
a hat on a skateboard.

A person is standing on a beach
with a surfboard.

A woman is sitting at a table
with a large pizza.

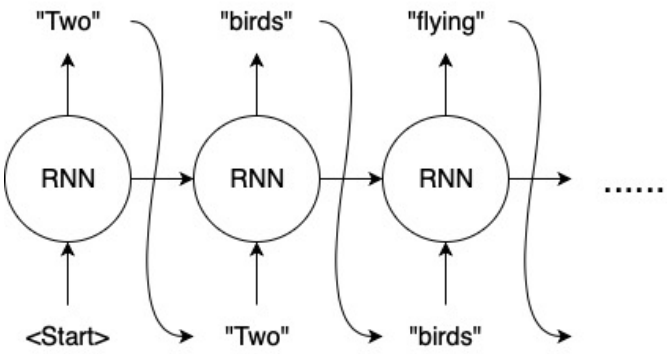
A man is talking on his cell phone
while another man watches.

Experiment Result (정량적 분석)

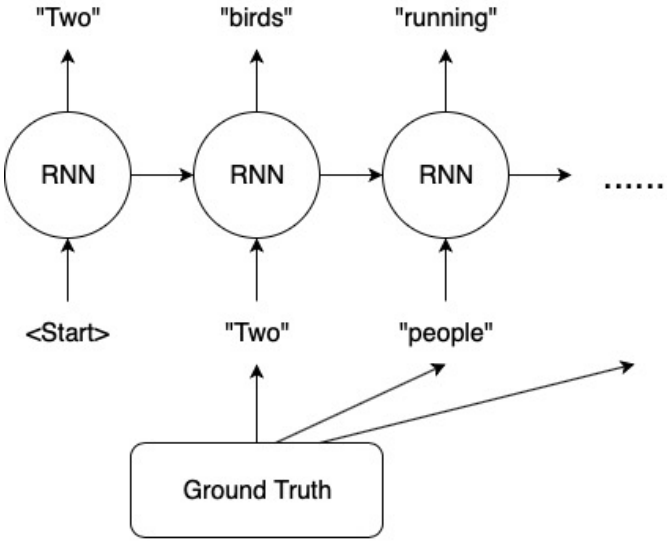
Table 1. BLEU-1,2,3,4/METEOR metrics compared to other methods, † indicates a different split, (—) indicates an unknown metric, ∘ indicates the authors kindly provided missing metrics by personal communication, Σ indicates an ensemble, a indicates using AlexNet

		BLEU				
Dataset	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR
Flickr8k	Google NIC(Vinyals et al., 2014)$^{\dagger\Sigma}$	63	41	27	—	—
	Log Bilinear (Kiros et al., 2014a)°	65.6	42.4	27.7	17.7	17.31
	Soft-Attention	**67**	44.8	29.9	19.5	18.93
	Hard-Attention	**67**	**45.7**	**31.4**	**21.3**	**20.30**
Flickr30k	Google NIC$^{\dagger\circ\Sigma}$	66.3	42.3	27.7	18.3	—
	Log Bilinear	60.0	38	25.4	17.1	16.88
	Soft-Attention	66.7	43.4	28.8	19.1	**18.49**
	Hard-Attention	**66.9**	**43.9**	**29.6**	**19.9**	18.46
COCO	CMU/MS Research (Chen & Zitnick, 2014)^{a}	—	—	—	—	20.41
	MS Research (Fang et al., 2014)$^{\dagger a}$	—	—	—	—	20.71
	BRNN (Karpathy & Li, 2014)°	64.2	45.1	30.4	20.3	—
	Google NIC$^{\dagger\circ\Sigma}$	66.6	46.1	32.9	24.6	—
	Log Bilinear°	70.8	48.9	34.4	24.3	20.03
	Soft-Attention	70.7	49.2	34.4	24.3	**23.90**
	Hard-Attention	**71.8**	**50.4**	**35.7**	**25.0**	23.04

Teacher Forcing



Without Teacher Forcing



With Teacher Forcing

TensorFlow를 이용한 Show, Attend and Tell 모델 구현

- https://github.com/solaris33/deep-learning-tensorflow-book-code/blob/master/Ch09-Image_Captioning/show_attend_and_tell/train_and_evaluate.py

Chapter 9 - Image Captioning

- im2txt - Show and Tell 모델 구현 (Code)
- show_attend_and_tell - Show, Attend and Tell 모델 구현 (TF v2 Keras Code)

Chapter 10 - Semantic Image Segmentation

- FCN.tensorflow - FCN(Fully Convolutional Networks) 모델 구현 (Code)

Chapter 11 - 생성모델(Generative Model) - GAN(Generative Adversarial Networks)

- GAN을 이용한 MNIST 데이터 생성 (Code) (TF v2 Code) (TF v2 Keras Code)

Chapter 12 - 강화학습(Reinforcement Learning)

- DQN을 이용한 게임 에이전트 구현 - CatchGame (Code) (TF v2 Code) (TF v2 Keras Code)

Questions & Answers

Thank You!