

Show and Tell: A Neural Image Caption Generator

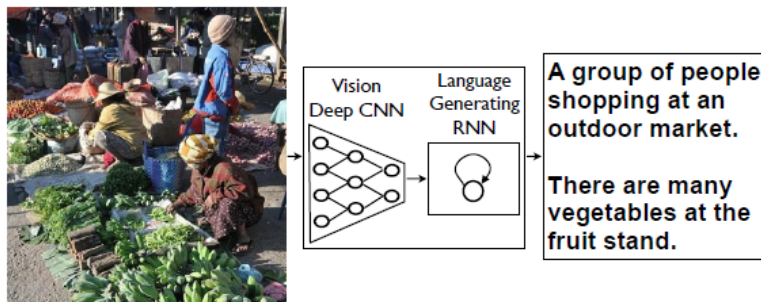
Oriol Vinyals(Google), Alexander Toshev(Google), Samy Bengio(Google), Dumitru Erhan(Google)

<https://arxiv.org/pdf/1411.4555.pdf>

1. Introduction

>> Image Caption Generation

- 이미지를 설명하는 문장을 생성하는 기술 분야를 의미.
- 대표적인 모델로는 본 논문에서 소개하는 Neural Image Caption(NIC)이 있음.
 - CNN을 이용해 이미지의 특징을 추출한 뒤 RNN을 거쳐 문장을 생성.



[그림 1]

- **"Computer Vision + Natural Language Processing"**
: Input을 이미지로 받으면 그 이미지를 설명해줄 수 있는 Caption을 Output으로 주기 때문에 Computer Vision과 NLP가 활용되었다고 할 수 있음.
- Image Captioning 기술은 앞을 못 보는 사람들에게 자신 앞의 화면을 설명해주어 도움을 줄수 있는 등 다양하게 응용 가능.
- Challenging task
 - 이미지를 설명하는 문장을 자동으로 만들어 내는 것(Image description)은 굉장히 어려운 문제.
 - Image classification, object detection보다 훨씬 어려운데, 단순히 이미지에 들어있는 object를 검출하는 것이 아니라 특성, 활동, 다른 object와의 관계까지 이해해야하기 때문.

>> Contribution

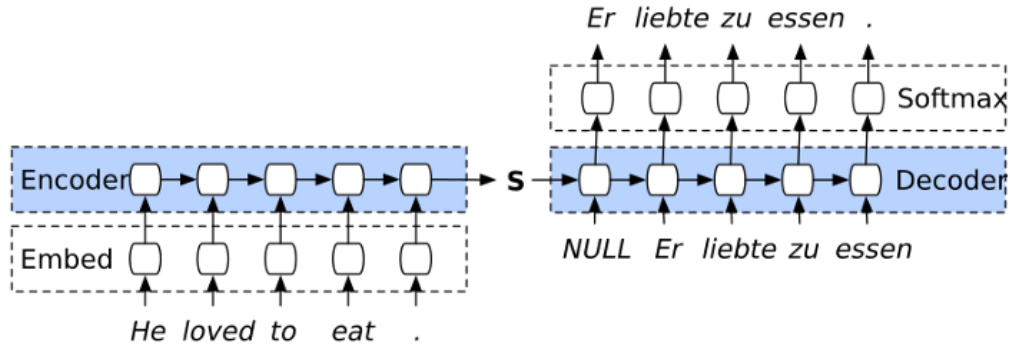
- **Image description 문제를 푸는 End-to-End 시스템을 제안.**
: Image Captioning 문제를 해결하려면 이미지에서 특징을 추출하는 단계, 그 특징을 단어로 바꾸는 단계 등 단계 별로 문제를 해결해야 한다. 그러나 본 논문에서는 한번에 문제를 해결하는 End-to-End 시스템을 제안한다. SGD를 사용하여 전체 신경망을 훈련 할 수 있다.

- 논문에서 제안하는 모델은 Vision과 Language 모델 중 SOTA인 모델들 구조의 일부를 조합하여 만들.

: Image Captioning은 기계 번역 문제에서 영감을 얻었다. 기계 번역 문제는 source 언어로 쓰인 문장(S)을 target언어로 번역된 문장(T)으로 변환하며 이때 $p(T|S)$ 를 최대화함으로써 해결한다.

예를 들어, 기계 번역 모델 중 하나인 seq2seq모델은 encoder RNN이 source 문장(S)를 입력으로 받아 고정 길이 벡터 표현으로 변환한다.

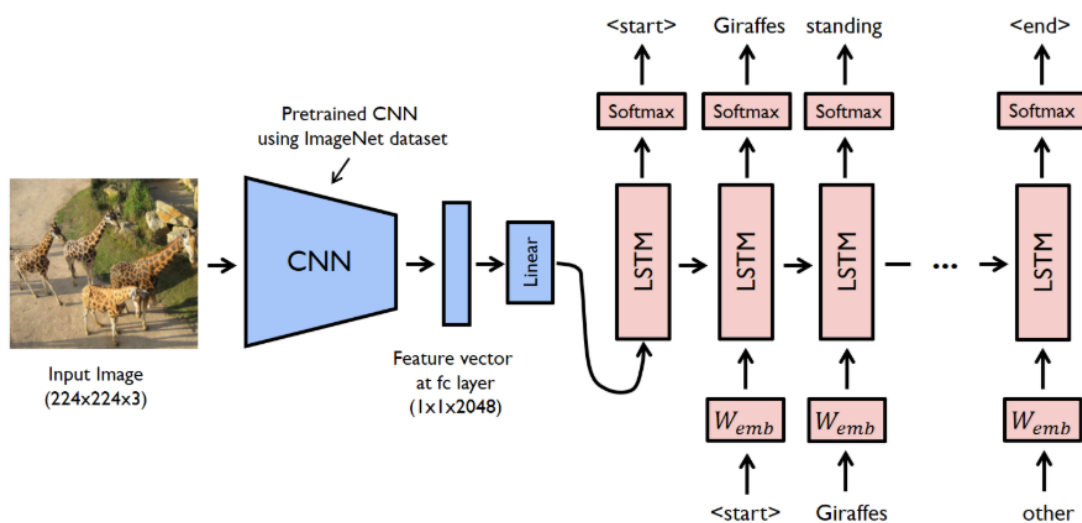
그리고 변환된 벡터 표현은 target 문장(T)을 생성하는 decoder RNN의 입력으로 사용된다.



[그림 2. seq2seq]

- 본 논문에서는 **encoder RNN을 CNN으로 대체**하는 방법을 제안한다.
- 이미지를 *encoder CNN에 입력*해서 나온 *output*을 *decoder RNN*으로 전달하여 문장을 생성하는 Neural Image Caption(NIC)모델을 개발.
- 이 모델은 이미지(I)를 입력으로 받아 단어 $S=\{s_1, s_2, \dots\}$ 의 타겟 시퀀스(문장, S)를 생성하는 Likelihood $p(S|I)$ 를 최대화하도록 훈련된다.
- 기존 SOTA보다 더 좋은 성능을 이끌어냄.

2. Model




[그림 3]

- Encoder의 구조를 CNN으로 대체, Decoder 구조는 Seq2Seq와 같고, 내부적으로 LSTM을 사용. (당시 SOTA 모델인 GoogleNet과 Seq2Seq를 사용함.)
- CNN의 Last hidden Layer(not output layer)를 Decoder의 input으로 넣어 문장을 생성하는 구조.



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


 A person riding a motorcycle on a dirt road.

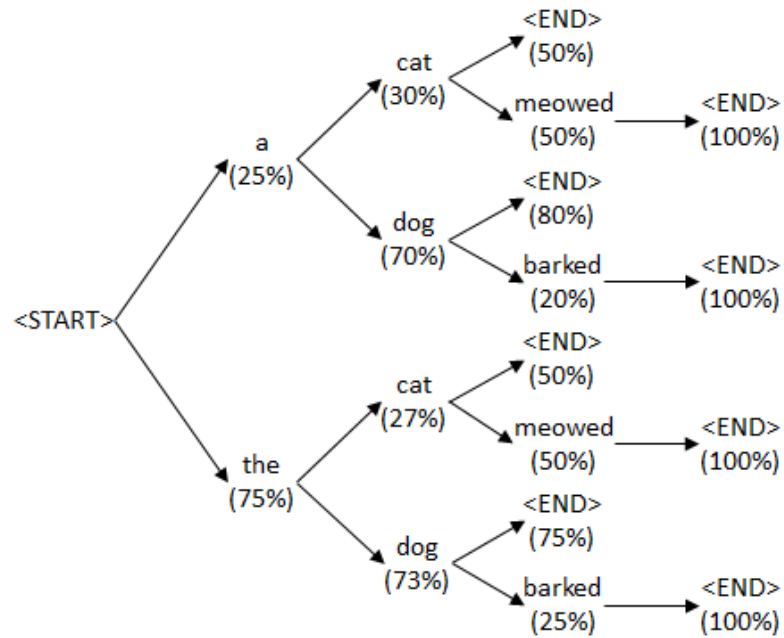
[그림 4]

- 학습 방향은 주어진 이미지(I)에 대응되는 정확한 묘사(S)의 확률을 최대화 하는 방향으로 network parameter(θ , not hyper-parameter)를 구하는 것.
 - 문장을 구성하는 단어의 길이는 그때그때 다르기 때문에, Description S 의 길이는 정해져 있지 않다. 즉, 실제 답의 길이가 N 개 일 때, 그에 대한 예측 답의 확률 $p(S|I)$ 은 고정된 수식이 아니다(N 이 사진마다 바뀌니까).
- 따라서 답의 확률 $p(s|I)$ 은 Chain Rule을 이용하여, $s_0 \dots s_N$ 까지 결합확률로 나타내야 한다.

(이미지 I 가 주어졌을 때 0번째 단어가 s_0 일 확률) \times (이미지 I 와 s_0 가 주어졌을 때, 1번째 단어가 s_1 일 확률) $\times \dots$ (이미지 I 와 $s_0 \dots s_{n-1}$ 주어졌을 때, n 번째 단어가 s^*_{n-1} 일 확률)

>> Inference

- **Sampling**
 - : 가장 확률이 높은 값(단어)을 고른다.
 - : 각 단계에서 최고의 단어 하나만 뽑는 것으로 Greedy 한 방법 -> 위험성이 있음.
 - 하나의 단어가 잘 못되면 그의 기반한 모든 결과가 망가지는 위험.
 - 각 단계에서의 최고의 단어만 뽑기 때문에 맥락적인 부분에서 어색할 수 있음.
- **Beam Search**
 - : k 개의 후보(단어)를 뽑아서 다음 $t+1$ 에서의 단어와의 조합의 확률을 보고 높은 값을 고른다.
 - : Beam Search는 단순히 가장 높은 확률을 가지는 단어를 선택하는 것이 아닌 상위 k 개를 뽑아 그 단어들로 다음 단어를 예측한 후 각 단어들의 확률을 더해 문장 단위로 높은 확률을 가지는 문장을 결과로 사용.



[그림 5]

3. Experiments

>> Evaluation Metrics

- **Amazon Mechanical Turk**

: 미국의 아마존이 운영하는 읽감을 가진 수요자 - 일을 할 수 있는 공급자를 연결해주는 웹 기반 서비스.

컴퓨터가 하지 못하고 사람이 가능한 일을 사람에게 시키는 것이 Mechanical Turk의 기본 철학.

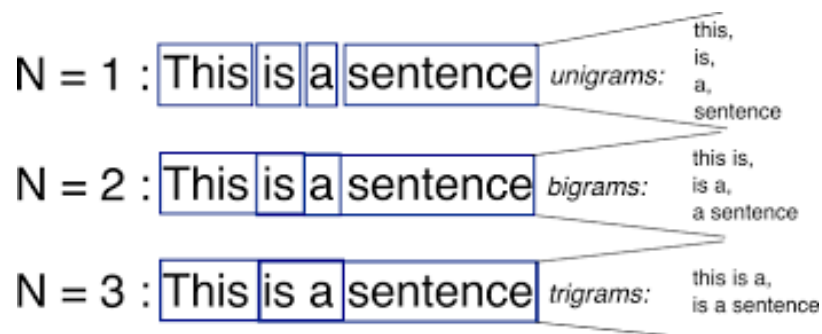
- **BLEU**

: Bilingual Evaluation Understudy, 자연어 처리에서 많이 쓰이는 성능 지표.

: 생성된 문장과 참조 문장간 단어 n-gram(n-grammar)의 정밀도 형태.

: 생성된 문장과 참조 문장 간의 n-gram을 통하여 얼마나 겹치는지 측정.

: 순서쌍들이 얼마나 겹치는지에 대한 측정을 통해 생성한 캡션의 실제 참조 캡션 유사도를 측정하는 방법.



[그림 6]

$$BLEU = \min(1, \frac{\text{output length}(\text{예측 문장})}{\text{reference length}(\text{실제 문장})}) (\prod_{i=1}^4 \text{precision}_i)^{\frac{1}{4}}$$

[그림 7]

: 문법구조, 유의어들에 대한 고려가 부족하기 때문에 한계가 있지만, 여전히 많이 사용.
 : 같은 의미의 다른 단어라도 틀렸다고 판단(단어 간 유사성 고려하지 않음)
 : BLEU는 단어별 가중치가 없기 때문에 중요한 단어가 대체되든, 안중요한 단어가 대체되는 비슷한 점수가 나옴.

- **METHOR**

: Metric for Evaluation of Translation with Explicit ORdering
 : 단어 일치, 형태소 분석 및 동의어 일치 등 파악 가능.
 : 문장 또는 세그먼트 수준에서 유사도를 측정하므로, BLEU가 말뭉치 수준에서 상관관계를 찾는다는 점에서 BLEU와 다름.
 : Precision, Recall(weighted towards recall)을 포함.

- **CIDer**

: Consensus-based Image Description Evaluation.
 : Image Captioning을 위해 제안된 지표.
 : 주로 정보 검색 평가에 사용되던 TF-IDF를 n-gram에 대한 가중치로 계산하고 참조 캡션과 생성 캡션에 대한 유사도를 코사인 유사도로 측정한 것.

<TF-IDF, 단어 빈도-역 문서 빈도, Term Frequency-Inverse Document Frequency>

-- TF-IDF는 모든 문서에서 자주 등장하는 단어는 중요도가 낮다고 판단하며, 특정 문서에서만 자주 등장하는 단어는 중요도가 높다고 판단.
 -- TF-IDF는 주로 문서의 유사도를 구하는 작업, 검색 시스템에서 검색 결과의 중요도를 정하는 작업, 문서 내에서 특정 단어의 중요도를 구하는 작업 등.

>> Data sets

- 이미지마다 5개의 문장 (SBU 제외).
- Pascal VOC 2008은 test로만 사용 (실험에서는 학습은 MSCOCO로 함).
- SBU는 Flickr에 올라온 사진과 글을 그대로 데이터로 사용.

-- Flickr는 사진과 사진에 대한 글을 올리는 사이트.
 -- SBU는 정확히 이미지를 설명하는 글이 아닌 사용자들이 올린 글이기 때문에 일종의 noise 역할을 하기를 기대함.

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	-	-

[그림 8]

4. Results

저자는 아래 3가지 질문에 답변을 하고자 함.

Q.1) 데이터 셋의 크기가 일반화에 어떻게 영향을 미치는가?

Q.2) 어떤 종류의 전이학습을 할 수 있는가?

Q.3) 약하게 라벨링 된 예제를 어떻게 처리할 것인가?

>> Generalization

- 좋은 데이터가 10만개 정도, 데이터가 많아지면 더 좋은 결과가 나올 것이라 예상.
- 데이터가 부족하기 때문에 generalization(overfitting 방지)을 하기 위해 노력함.
- **Generalization Result**
 - 여러 metric으로 평가해봄.
 - 사람보다 점수가 높은 경우가 있지만 실제 결과는 그렇지 않음.
 - metric에 대한 연구도 더 필요할 것으로 보임.

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]	25	55	58	11
TreeTalk [18]				19
BabyTalk [16]				
Tri5Sem [11]				48
m-RNN [21]				58
MNLM [14] ⁵				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.

[그림 9]

Generation Diversity Discussion

- generating model 모델이 새롭고 다양하고 높은 퀄리티의 문장을 만들어내는지 확인.
- 굵게 표시된 문장이 dataset에 없는 문장이다.

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

[그림 10]

>> Transfer Learning

- 다른 dataset 간의 transfer가 가능한지 실험.
- Flickr30k -> Flickr8k (유사 데이터, 데이터 차이 4배)
 - BLEU 4 증가
- MSCOCO -> Flickr8k (다른 데이터, 데이터 차이 20배)

- BLEU 10 감소, but 만든 문장은 괜찮음.
- MSCOCO -> SBU
 - BLEU 16 감소

>> Ranking Results

- 다른 연구에서 사용되는 지표인 Ranking Scores에서도 높은 점수를 받음.

Approach	Image Annotation			Image Search		
	R@1	R@10	Med <i>r</i>	R@1	R@10	Med <i>r</i>
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.

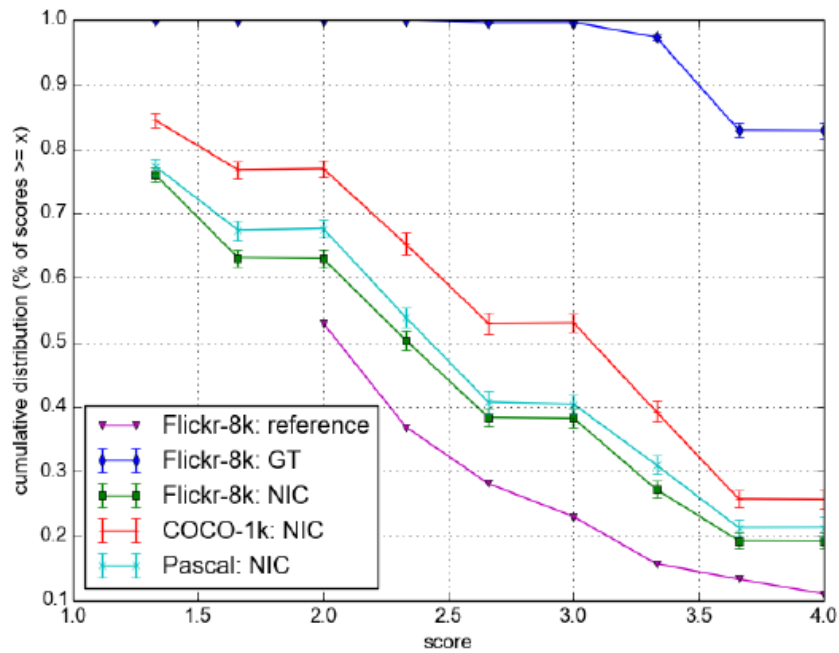
Approach	Image Annotation			Image Search		
	R@1	R@10	Med <i>r</i>	R@1	R@10	Med <i>r</i>
DeFrag [13]	16	55	8	10	45	13
m-RNN [21]	18	51	10	13	42	16
MNLM [14]	23	63	5	17	57	8
NIC	17	56	7	17	57	7

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

[그림 11]

>> Human Evaluation

- 사람이 직접 평가한 지표를 보여줌.
- BLEU Score는 human보다 높았는데, 여기서는 낮음.
- BLEU 지표가 완벽한 지표는 아님을 보여줌.



[그림 12]



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

[그림 13]

>> Analysis of Embedding

- Word embedding vector도 유사한 단어끼리 뭉쳐있도록 잘 학습됨을 확인.

Word	Neighbors
car	van, cab, suv, vehicule, jeep
boy	toddler, gentleman, daughter, son
street	road, streets, highway, freeway
horse	pony, donkey, pig, goat, mule
computer	computers, pc, crt, chip, compute

Table 6. Nearest neighbors of a few example words

[그림 14]

5. Conclusion

- End-to-End 뉴럴 네트워크 시스템인 NIC(Neural Image Caption)를 제안함.
 - NIC는 이미지를 보고 영어로 적절한 설명을 생성.
- NIC는 이미지를 compact한 표현으로 인코딩하는 convolution 신경망을 기반으로 하고, 해당 문장 생성은 순환 신경망을 활용.
- 모델은 이미지가 주어졌을 때 문장의 Likelihood를 최대화하도록 훈련됨.

[Reference]

- (논문) <https://www.youtube.com/watch?v=yfsFW-mfOEY&list=LL&index=3&t=6s>
- (논문) <https://velog.io/@a01152a/%EB%85%BC%EB%AC%B8-%EC%9D%BD%EA%B8%B0-%EB%B0%8F-%EA%B5%AC%ED%98%84-Image-Captioning>
- (논문) <https://mrsyee.github.io/nlp/2018/11/24/Show and tell/>
- (논문) <https://uding.tistory.com/20>
- (Amazon Mechanical Turk) <https://blog.daum.net/buzzweb/710>
- (성능지표) <http://javaspecialist.co.kr/board/1062;jsessionid=5193BB88FB52F9036F9180512756D3CE>
- (성능지표) <https://wikidocs.net/145607>
- (METEOR) <https://slideplayer.com/slide/7480005/>
- (CIDer) <https://s-space.snu.ac.kr/bitstream/10371/166538/1/000000159379.pdf>
- (그림1) <https://sh-tsang.medium.com/review-show-and-tell-a-neural-image-caption-generator-2d3928a90306>
- (그림2) <https://towardsdatascience.com/nlp-sequence-to-sequence-networks-part-2-seq2seq-model-encoderdecoder-model-6c22e29fd7e1>
- (그림3) <https://velog.io/@a01152a/%EB%85%BC%EB%AC%B8-%EC%9D%BD%EA%B8%B0-%EB%B>

[0%8F-%EA%B5%AC%ED%98%84-Image-Captioning](#)

(그림4) <https://velog.io/@a01152a/%EB%85%BC%EB%AC%B8-%EC%9D%BD%EA%B8%B0-%EB%B0%8F-%EA%B5%AC%ED%98%84-Image-Captioning>

(그림5) <https://velog.io/@a01152a/%EB%85%BC%EB%AC%B8-%EC%9D%BD%EA%B8%B0-%EB%B0%8F-%EA%B5%AC%ED%98%84-Image-Captioning>

(그림6) http://rstudio-pubs-static.s3.amazonaws.com/460514_4377b4f645a944d788ae7300782123f3.html#2

(그림7) <https://velog.io/@a01152a/%EB%85%BC%EB%AC%B8-%EC%9D%BD%EA%B8%B0-%EB%B0%8F-%EA%B5%AC%ED%98%84-Image-Captioning>

(그림8) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/

(그림9) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/

(그림10) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/

(그림11) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/

그림12) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/

(그림13) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/

(그림14) https://mrsyee.github.io/nlp/2018/11/24/Show_and_tell/