ISSN 2383-630X(Print) / ISSN 2383-6296(Online)
Journal of KIISE, Vol. 48, No. 3, pp. 340-351, 2021. 3
https://doi.org/10.5626/JOK.2021.48.3.340

이미지 캡션 생성을 위한 다중 관점을 가진 자가 교열 트랜스포머

(Self-revising Transformer with Multi-view for Image Captioning)

이 지 은 † 박 진 욱 † 박 상 현 ††
(Jieun Lee) (Jinuk Park) (Sanghyun Park)

요 약 이미지 캡션 생성이란 주어진 이미지로부터 객체 요소를 파악하여 장면을 설명하는 자연어를 자동으로 서술하는 연구이다. 선행 연구에서는 주로 단일 특징 추출기를 통해 이미지에서 정보를 포착한 후, 순환 신경망 기반의 디코더를 통해 캡션을 생성한다. 하지만 단일 특징 추출기를 사용하기 때문에 다중 관점의 이미지 정보를 사용할 수 없고, 순환 신경망 기반의 장기 의존성 문제를 가지는 디코더를 사용한다. 이를 해결하기 위해서 본 연구는 복수의 특징 추출기를 사용하는 다중 관점 인코더를 통해 다양한 각도의 이미지 정보를 가공하여 전달한다. 또한, 순환 신경망의 한계를 보완하기 위해서, 트랜스포머 모델 기반의 디코더 레이어에 추가적인 멀티-헤드 주의 기제 기법을 통해 생성된 문장을 재구축하여 문장의 완성도를 높이는 자가 교열 트랜스포머를 제안한다. 제안하는 모델의 검증을 위해 MSCOCO 데이터셋을 이용하여 다양한 비교실험으로 정량적, 정성적 평가를 통해 제안한 방법론의 우수성을 검증하였다.

키워드: 자연어 처리, 이미지 캡션 생성, 멀티-헤드 주의 기제 기법, 다중 관점 인코더

Abstract Image captioning is a task of automatically describing a scene by identifying an object element from a given image. In prior research, information has mainly been captured from the image using a single feature extractor, and captions have then been generated by a recurrent neural network-based decoder. However, multi-view image information is not available with this method because of the use of a single feature extractor, and the use of a recurrent neural network-based decoder causes a long-term dependency problem. To address these issues, the proposed model employs a multi-view encoder using a couple of feature extractors that provide processed image information from various view. In addition, to supplement the limits of the recurrent neural network, we propose a self-revising transformer that increases the completeness of sentences by revising the generated sentences by focusing additional multi-head attention in the transformer-based decoder layer. To present the proposed model, we verify its superiority through quantitative and qualitative evaluations with various comparative experiments using MSCOCO datasets.

Keywords: natural language processing, image captioning, multi-head attention, multi-view encoder

· 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(IITP-2017-0-00477, (SW스타랩) IoT 환경을 위한 고성능 플래시 메모리 스토리지 기반 인메모리 분산 DBMS 연구개발)과 국토교통부의 스마트시티 혁신인재육성사업의 지원을 받아 수행된 연구임

† 비 회 원 : 연세대학교 컴퓨터과학과 학생
jieun199624@yonsei.ac.kr
parkju536@yonsei.ac.kr
†† 종신회원 : 연세대학교 컴퓨터과학과 교수(Yonsei Univ.)
sanghyun@yonsei.ac.kr
(Corresponding author임)

논문접수 : 2020년 10월 12일
(Received 12 October 2020)
논문수정 : 2020년 12월 12일
(Revised 12 December 2020)
심사완료 : 2020년 12월 15일
(Accepted 15 December 2020)

정보과학회논문지 제48권 제3호(2021. 3)

1. 서 론

이미지 캡션 생성(Image Captioning)은 주어진 이미지의 구성요소를 파악하여 장면을 묘사해주는 자연어를 자동으로 생성하는 작업을 말한다. 이미지 캡션 생성은 여러 어플리케이션에 적용될 수 있다. 예를 들면, 시각장애인들을 위해 이미지를 실시간으로 설명해주거나[1], 웨어러블 카메라를 통한 라이프로그 사진의 캡션을 자동으로 생성해준다[2]. 또한, 드론을 통한 범죄 및 안전사고를 감지[3]하는 등을 연구하는 스마트시티 분야에서도 활용될 수 있다.

대표적인 이미지 캡션 생성 연구는 두 가지의 딥러닝(Deep Learning) 기술의 결합이다. 컴퓨터 비전 분야의 이미지 정보를 얻는 합성곱 신경망(Convolution Neural Network) 모델과 기계 번역(Machine Translation)에서 일반적으로 사용되는 순환 신경망(Recurrent Neural Network) 모델을 사용한다. 일반적인 이미지 캡션 생성 모델은 인코더-디코더 구조를 가진다. 인코더는 사전 학습된 합성곱 신경망 모델을 전이 학습(Transfer Learning)하여 이미지의 오브젝트 분류 및 특징을 얻는 특징 추출기(Feature Extractor)로 사용한다. 합성곱 신경망 모델은 전통적으로 사전 학습된 이미지 분류(Image Classification) 모델[4-7]을 사용하지만, 추후 연구에서 객체 탐지(Object Detection)를 위한 Faster R-CNN[8]을 이미지 캡션 생성 연구에 맞춰 수정[9]하여 사용한다. 그리고 디코더는 순환 신경망 모델에 속하는 LSTM(Long Short-Term Memory)이나 GRU(Gated Recurrent Unit)를 통해 인코더에서 얻은 이미지 특징 정보와 결합하여 자연어로 이루어진 캡션을 생성한다[10]. 이 후, 기계 번역 연구에서 주의 기제 기법(Attention)[11]이 등장하며 단어와 단어 사이의 관계의 정보를 이용하여 성능 향상에 큰 기여를 하며 이미지 캡션 생성에도 적용되었다. 이미지 캡션 생성 모델 안의 주의 기제 기법은 단어와 이미지 사이의 관계 정보를 사용하여 자연어를 생성할 때 이미지의 특정 부분을 집중해줌으로써 이미지 안의 객체를 더 잘 뽑을 수 있도록 하였다[12].

이미지 캡션 생성 모델의 성능은 특징 추출기로 뽑아내는 이미지 정보의 우수성과 관련이 깊다. 추출된 이미지 정보의 우수성이 낮을수록 언어 모델이 생성하는 캡션은 일반적인 문장이 된다는 한계가 있다. 기존의 이미지 캡션 생성 모델들은 인코더에서 하나의 특징 추출기를 통해 단일 관점으로 이미지의 정보를 추출한다. 본 연구는 특징 추출기의 성능 개선에 초점을 두어 다중 관점(Multi-View)을 가지는 모델을 제안한다. 다중 관점을 갖기 위해서 복합적인 특징 추출기를 사용하여 이미지 정보를 뽑아낸다. Faster R-CNN과 Mask R-CNN[13], 이 두 가지의 특징 추출기를 적용하여 같은 이미지에서 서로 다른 정보를 얻음으로써 부족한 정보를 상호 보완할 수 있도록 한다. 더불어 트랜스포머 기반의 다중 관점 인코더를 통해서 얻은 이미지 정보의 중요도를 산정하여 업데이트한다. 인코더를 통해 구한 두 이미지 정보는 모델의 디코더에서 스택처럼 쌓는 결합 구조를 통해 합한다.

그러나 특징 추출기를 통해서 우수한 이미지 정보를 뽑아내더라도, 캡션을 생성하는 언어 모델의 성능이 떨어지면 캡션의 질을 기대할 수 없다. 본 연구는 언어 생성 모델을 통해 얻은 캡션의 중복성을 제거하고 정확성을 향상시키기 위해 자가 교열 트랜스포머(Self-Revising Transformer)를 제안한다. 자가 교열 트랜스포머는 기존의 모델과 달리 디코더 레이어에 Mask 멀티-헤드 주의 기제 네트워크를 하나 더 쌓아서 언어 모델의 역할을 강화하여 단어의 중복된 표현을 억제하고, 생성된 문장을 재구축하여 연결성을 높인 문장을 생성할 수 있게 도움을 준다.

본 논문의 기여는 다음과 같다. 1) 다중 관점을 갖기 위해 이미지 특징 추출기를 복합적으로 사용함으로써, 이미지 정보를 다각도로 제공하고, 다중 관점 인코더를 통해 이미지 정보사이의 중요도를 계산한다. 2) 자가 교열 트랜스포머를 제안하여 생성되는 캡션의 완성도를 높인다. 3) 앞의 제안한 방법론을 정량적, 정성적 평가를 통해 성능의 향상을 증명한다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구를 소개하고, 3장에서는 사용하는 캡션 모델과 제안하는 모델의 방법론을 제시한다. 4장에서는 본 모델의 성능 비교 실험 및 생성한 예제를 분석하고, 5장에서 결론 및 향후 계획을 기술한다.

2. 관련 연구

이미지 캡션 생성 연구는 기계 번역 연구가 발전에 힘입어 성능의 향상이 이루어졌다. 성공적인 기계 번역 연구의 하나인 시퀀스(Sequence)에서 시퀀스를 생성하는 모델[14]을 적용하여 인코더-디코더 구조를 적용한 모델[10]은 입력 값이 이미지인 것에 맞춰 순환 신경망 모델을 사용하는 인코더를 합성곱 신경망 모델로 수정하여 성능의 향상을 보였다. 그러나 디코더에서 순환 신경망 모델을 사용하여 모델의 단점인 장기 의존성(Long-Term Dependency)로 인한 정보 손실의 특징을 안고 있다. 이를 보완하기 위해 [12]는 [11]을 통해 이미지 캡션 생성 모델에 하드 주의 기제 기법(Hard Attention)과 소프트 주의 기제 기법(Soft Attention)을 통해 성능의 발전을 보였다. 이후의 연구는 [12]의 소프트 주의 기제 기법을 응용하여 다양한 모델을 제안한다. 예를 들

면, [15]은 적응형 주의 기제 기법(Adaptive Attention)을 통해 캡션을 생성할 때 이미지 정보와 언어 모델 중 어떤 정보를 편중할지 게이트를 통해 결정한다.

이후 자연어 처리 분야에서 연구된 모델인 트랜스포머[16]의 등장은 기계 번역의 성능을 높은 수준으로 개선시켰다. 트랜스포머 모델은 기존의 자연어처리에서 일반적으로 사용된 순환 신경망 모델 대신 주의 기제 기법으로 대체함으로써, 입력 문장 사이의 관계 정보를 추가로 갖을 수 있게 되었고 훈련 시간을 크게 단축했다. 트랜스포머 모델을 사용한 [17]는 이미지 캡션 생성 연구에 맞게 모델 구조를 수정하였고, 기하학 주의 기제 기법(Geometric Attention)을 통해 이미지 안의 객체 간의 관계에 대한 정보를 결합한다. [18] 또한, 트랜스포머 모델의 디코더 구조를 차용하고 멀티-레벨 지도학습을 적용한다. 앞의 선행연구들과 달리, [19]는 트랜스포머의 모델 구조를 차용하지 않았지만 순환 신경망 모델을 사용하면서 자가-주의 기제 기법을 융합하여 사용한다.

이 외에도, 강화학습을 통해 모델을 최적화하는 방법론이 제안되었다. SCST(Self-Critical Sequence Training)[20]는 샘플링된 캡션과 추론 알고리즘으로 생성된 캡션의 사이의 CIDEr-D 스코어[21]의 오차를 리워드로 주어 강화 학습하여 모델을 최적화한다. 해당 방법론은 많은 선행 연구에서 모델을 최적화하기 위해 사용되었다[9,22-25]. [26]는 SCST를 변형(Variant)하여 리워드를 계산할 때, 추론 알고리즘을 사용하지 않고 샘플링된 캡션의 평균을 이용하여 성능을 향상시켰다. 다른 선행 연구로 [27]는 트랜스포머 모델 기반의 연구로, 트랜스포머 모델과 강화학습의 결합의 어려움을 지적하며 SCST를 변형한 off-policy SCST를 제안한다.

본 연구는 입력 이미지의 특징 간의 관계 정보를 얻기 위하여 다중 관점 인코더와 순환 신경망의 장기 의존성 한계를 극복한 트랜스포머 모델 기반의 디코더를 사용한다. 특히, 일반적인 이미지 캡션 생성 모델[10, 12,15,17-19,22-27]과는 차별적으로 단일 특징 추출기가 아닌 다중 특징 추출기를 사용함으로써, 다양한 관점의 이미지의 정보를 제공한다. 그리고 [17,18]과는 다르게 디코더에 기존의 트랜스포머 구조를 차용하지 않고 언어 모델의 역할을 강화한 자가 교열 트랜스포머를 제안하여 적용한다. 마지막으로, SCST를 통해 훈련된 모델을 최적화하여 더욱 성능을 높였다.

3. 모델 구조

3.1 Captioning 모델

캡션을 생성하는 모델은 순환 신경망을 사용하는 전통적인 방법론과 달리 주의 기제 기법만으로 훈련하는 기계 번역 모델, 트랜스포머 모델의 구조를 차용한다. 트랜스포머 모델은 다음과 같은 4가지의 핵심 네트워크로 구성되어 있다.

3.1.1 위치 인코딩

자연어를 모델이 이해할 수 있도록 실수 벡터로 매핑하는 과정을 단어 임베딩(Word Embedding)이라고 한다. 전통적인 순환 신경망 모델을 사용하는 자연어처리 연구는 단어 임베딩을 통해 자연어를 벡터로 매핑을 한다. 트랜스포머 모델 또한 단어 임베딩이 필요하지만, 순차적이거나 합성곱 방법론이 아닌 주의 기제를 사용하는 병렬적인 방법론이기 때문에 입력 문장의 단어들의 순서들을 유지하기가 어렵다. 따라서, 단어의 위치 정보를 단어 임베딩에 추가로 제공할 필요가 있다. 트랜스포머 모델은 이 문제점을 보완하기 위해서 위치 인코딩(Positional Encoding)을 통해 단어의 상대적, 절대적 위치에 대한 정보를 단어 임베딩에 포함한다. 본 연구에서는 위치 정보가 중요하지 않은 이미지를 입력 값으로 사용하는 인코더에는 위치 인코딩을 적용하지 않고, 문장을 입력 값으로 사용하는 디코더에만 적용시킨다.

3.1.2 스케일 내적 주의 기제

주의 기제 기법이 핵심인 트랜스포머 모델은 내적 주의 기제 기법(Dot-Product Attention)을 수정하여 사용한다. 수식은 다음과 같다.

$$Attention(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (1)$$

주어지는 입력 벡터는 Q(Query), K(Key), V(Value) 그리고 d_k (Key의 크기)이다. Query와 Key의 내적을 구한 뒤 Key의 크기의 제곱근으로 나눠 줌으로써 벡터의 규모를 축소한다. 소프트맥스(Softmax) 함수를 사용하여 벡터 안의 인자들을 합이 1이 되는 분포로 변환하고 Value를 곱함으로써 Value에 대한 가중치를 얻는다. 트랜스포머 모델은 내적 주의 기법에서 추가적으로 Key의 크기로 벡터의 규모를 축소하는 규모 축소 내적 주의 기제 기법(Scaled Dot-Product Attention)[16]을 사용한다. Key의 크기가 커질수록 내적 결과 값의 규모가 커지기 때문에, 소프트맥스 함수를 취하게 되면 그라디언트 값이 급격히 줄어든다. 때문에 소프트맥스 함수를 취하기 전에 벡터의 규모를 축소해줄 필요가 있다.

3.1.3 멀티-헤드 주의 기제

일반적인 주의 기제 기법은 식 (1)을 한 번 연산하는 것으로 Value의 가중치를 얻는다. 트랜스포머 모델은 한 번의 연산으로 끝내지 않고 하나의 데이터로부터 복수의 다른 표현형을 얻은 후 합쳐서 다각도의 정보를 얻는 방법론을 사용한다. Query, Key 그리고 Value를 h번 선형 투영을 하여 h번의 스케일 내적 주의 기제를 계산한다.

$$head_i = Attention\left(QW_i^Q,\ KW_i^K, VW_i^V\right) \quad (2)$$

$$MHAttention\,(Q,K,V) = [head_1,\ldots,head_h]W^O \quad (3)$$

수식 (2)를 통해 서로 다른 값을 가지는 개의 head를 계산하여 수식 (3)에서 모든 head들을 취합하여 최종 값을 얻는다. 이를 통해 단어들에 대한 다양한 주의 기제 정보를 얻을 수 있다.

3.1.4 포지션-와이즈 순방향 네트워크

트랜스포머 모델의 레이어들은 서로 독립된 포지션-와이즈 순방향 네트워크(Position-wise Feed-Forward Network)를 포함한다. 이 네트워크는 수식 (4)와 같이 두 개의 선형 변환과 렐루(ReLU) 활성화 함수로 계산된다. 수식 (4)에서 $W_1 \in \mathbb{R}^{d_{ff}}, W_2 \in \mathbb{R}^{d_{model}}$ 이다.

$$FFN(x) = \max(0, xW_1 + b_1)\,W_2 + b_2 \quad (4)$$

본 연구에서 사용하는 이미지 캡션 생성 모델의 구조는 Fig. 1과 같다. 인코더와 디코더는 앞서 설명한 네트워크들로 구성되어 있고, 본 연구는 기존의 트랜스포머 모델과는 달리 입력 값이 자연어가 아니기 때문에 이미지의 특징 벡터를 생성하는 과정이 필요하다. 이미지는 특징 추출기를 통해 특징 벡터를 생성하여 각 인코더의 입력 값으로 주어진다. 제안하는 모델은 특징 벡터를 복수로 사용하기 때문에, 인코더는 특징 벡터마다 독립적으로 사용되어야 한다. 이미지로부터 인코딩된 결과값은 디코더에 입력되어 최종적으로 이미지를 설명하는 캡션이 생성된다.

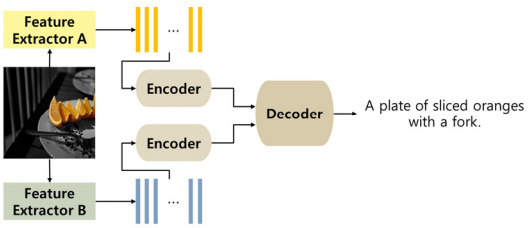

그림 1 제안하는 인코더-디코더 구조 개요

Fig. 1 Overview of the encoder-decoder framework of the proposed model

3.2 이미지 특징 추출기

기계 번역 연구에 속하는 트랜스포머 모델은 텍스트로부터 텍스트를 생성하지만, 이미지 캡션 생성은 이미지로부터 텍스트를 생성하는 연구이다. 때문에 인코더의 입력 값이 이미지이므로 트랜스포머 모델처럼 단어 임베딩을 구하는 것이 아니라 이미지를 벡터로 변환하는 과정이 필요하다. 본 연구는 다중 관점을 갖는 이미지 정보를 얻기 위해 이미지 특징 벡터를 추출하는 두 종류의 특징 추출기를 복합적으로 사용한다. R-CNN[28] 계열의 사전 학습된 합성곱 신경망 모델 객체 Faster R-CNN[8]과 Mask R-CNN[13]을 사용한다. Faster R-CNN은 컴퓨터 비전 분야에서 객체 탐지 모델로 이미지에서 탐지되는 객체의 위치를 박스(bounding box)로 표시를 하고 해당 객체의 클래스를 분류한다. Mask R-CNN은 인스턴스 분할(Instance Segmentation) 모델로 Faster R-CNN과는 달리 객체를 박스로 구분하는 것뿐만 아니라 박스 안의 해당 객체의 영역을 더 구체적으로 찾고, 객체의 클래스가 다른 클래스와 중복이 되어도 서로 다른 클래스로 분류한다.

Faster R-CNN으로 추출한 특징 벡터는 [29]로 사전 학습된 후에 [30]로 훈련하여 얻은 데이터를 사용한다 [9]. Mask R-CNN은 [9]의 방법과 유사하게 벡터를 추출하여 적응형 평균 풀링(Adaptive Average Pooling)을 통해 차원을 축소한다. 각 합성곱 신경망 모델에서 추출된 특징 벡터는 선형 변환을 통해 입력 값 임베딩 차원 d_{model} 로 축소시킨다.

$$V_a = \{a_1, a_2, \ldots, a_n\}, \qquad V_b = \{b_1, b_2, \ldots, b_m\}$$
$$a_i, b_j \in \mathbb{R}^{d_{model}}$$

이미지 특징 벡터 V는 차원이 d_{model} 인 a_i, b_j 벡터로 구성되어 있다. n과 m은 이미지에서 선택된 객체 영역의 개수이기 때문에 10에서 100사이의 동적인 값이다.

3.3 다중 관점 인코더

이미지에서 추출한 특징 벡터 V_a와 V_b는 다른 합성곱 신경망 모델에서 얻어졌기 때문에 동일한 인코더를 사용하여 파라미터를 공유할 수 없다. 따라서 Fig. 2에서 보이는 독립적인 두 개의 인코더를 사용하여 개별적인 복수의 결과값을 얻는 다중 관점 인코더(Multi-View Encoder)를 사용한다. 다중 관점 인코더 안의 멀티-헤드 주의 기제 기법을 통해 이미지 정보 사이의 중요도를 기반으로 가중치를 업데이트한다. 그리고 포지션-와이즈 순방향 네트워크를 통과하여 결과값을 구한다. 차원은 d_{model} 로 이미지 특징 벡터 V와 차원이 동일하다.

3.4 멀티모달 트랜스포머 디코더

선행연구들과 달리 본 연구는 다중 관점의 인코더를 사용하여 복수의 인코딩된 특징 값들을 사용하므로, 디코더에서 복수의 특징 값들을 모두 사용할 수 있도록 수정된 주의 기제 기법을 제안한다. 트랜스포머 모델은 디코더 레이어 안의 멀티-헤드 주의 기제 기법은 두 종류가 있다. 첫 번째로 디코더의 입력 값으로 자가-주의 기제(Self-Attention)를 통해 중간 값을 계산한다. 자가-주의 기제는 수식 (1)에서의 Query, Key 그리고 Value가 모두 같은 값인 상황에서 주의 기제를 할 때를 말한다. 디코더는 다음에 생성되는 단어를 예측하는 동작하기 때문에 현재 단계 이전의 데이터만 계산하는 곳에 포함되어야 한다. 데이터 사이의 인과 관계를 유지하기 위해서 마스크를 사용하는 Masked 멀티-헤드 주의 기제 기법을 사용한다. 두 번째는 인코더의 결과 값과 이

전 네트워크에서 나온 중간 값으로 멀티-헤드 주의 기제를 한다. 트랜스포머 모델의 경우 단어와 단어 사이의 주의 기제 기법인데 반해, 본 연구는 이미지와 단어 사이의 멀티모달(Multi Modal) 주의 기제 기법을 계산한다.

3.4.1 스택 멀티모달 주의 기제 기법

본 연구는 인코더의 결과 값이 단일이 아닌 복수이므로 인코더의 결과 값과 디코더의 중간 값으로 멀티-헤드 주의 기제 기법의 네트워크가 복수 개 필요하다. 또한, 네트워크를 통해 얻은 값들을 적절히 결합하는 방법, 즉 복수 개의 멀티-헤드 주의 기제 네트워크의 구성을 고안할 필요가 있다. 이미지 특징 벡터 V_a와 V_b에서 인코더를 통해 나온 값 M_a와 M_b (Fig. 2의 Embedded Image Features)을 결합하는 방법론을 제안한다.

인코더와 디코더를 결합하는 방법으로 Fig. 3과 같이 세 가지를 제안한다. Fig. 3에 나타나는 "MHA with M_a, M_b"는 각 M과 멀티모달 멀티-헤드 주의 기제를 하는 네트워크를 나타낸다. 첫 번째 방법론 (a)는 한 레이어 안에서 M_a와 M_b의 멀티모달 멀티-헤드 주의 기제를 연속적으로 쌓은 구조이다. 따라서 (a)는 한 레이어 안에서 서로 다른 M의 정보를 주의 기제 기법을 통해 순차적으로 결합하며 훈련을 진행한다. 두 번째 방법론 (b)는 (a)와 달리, 별개의 레이어에서 하나씩 주의 기제를 하여 레이어를 쌓는다. 먼저 하나의 M에 대해서 훈련을 진행하고, 그 이후에 다른 이미지 정보를 이용하여 모델을 학습시킨다. 그리고 각 레이어의 개수는 원래 수치의 절반에 해당한다. 마지막으로 (c)는 M_a와 M_b에 대한 주의 기제를 동시에 한 후에 결과값을 연결(Concatenate)하여 선형변환(Linear Transformation)을 통해 차원을 감소시켜서 레이어를 쌓는 구조이다. (a)는 M의 정보를 주의 기제 기법을 통해 결합하는 방법이지만, (c)는 선형 변환을 통해 결합하는 차이가 있다.

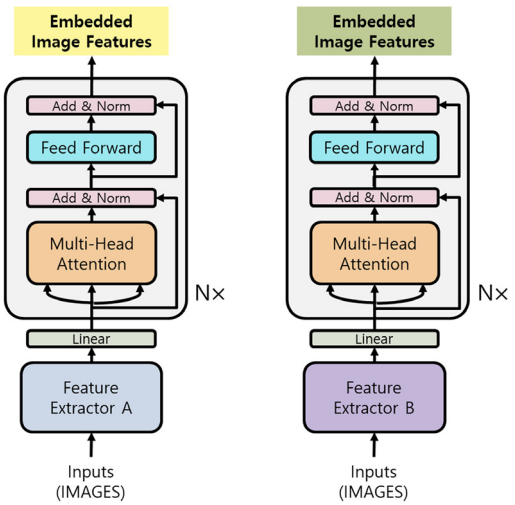

그림 2 다중 관점 인코더

Fig. 2 Multi-view Encoder

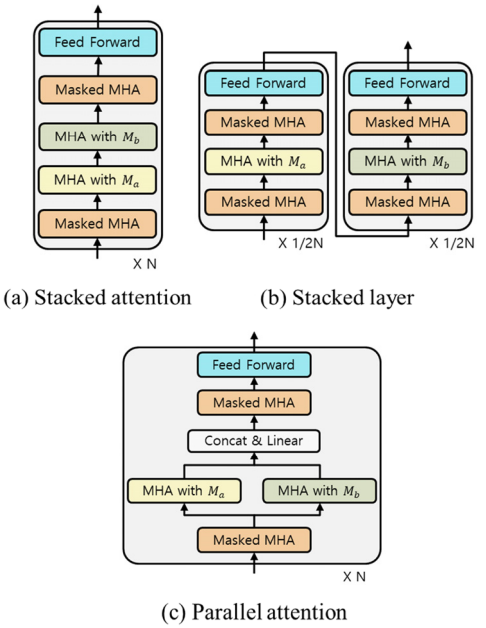

그림 3 인코더와 디코더의 결합 방법론

Fig. 3 Combined Methods between Encoder and Decoder

본 연구는 실험 결과를 통해 (a)의 방법론을 채택하여 비교 실험을 진행하였다. 즉, 모델에서 사용한 방법론은 멀티모달 멀티-헤드 주의 기제를 스택처럼 쌓아서 결합하는 구조이다.

$$h_1^l = MHAttention(x^l,\ x^l,\ x^l) \tag{5}$$

$$h_2^l = MHAttention(h_1^l, M_a, M_a) \tag{6}$$

$$h_3^l = MHAttention(h_2^l, M_b, M_b) \tag{7}$$

$$x^l, h_{1,2,3}^l, , M_{a,b} \in \mathbb{R}^{d_{model}} \tag{8}$$

x^l은 디코더의 입력 값이고, 각 인자의 차원은 수식 (8)과 같다. 입력 값에 대해서 수식 (5)를 계산하여 h_1^l을 구한다. 결과 값을 통해 수식 (6)에서 h_1^l과 M_a를 멀티-헤드 주의 기제 기법을 통해 h_2^l를 얻는다. 이와 비슷하게, 수식 (7)은 h_2^l과 M_b를 통해 h_3^l를 얻는다. 수식에서 보이는 것처럼, 본 연구는 다중 특징 추출기를 통해 얻은 서로 다른 이미지 특징들을 순차적으로 쌓아서 정보를 합하는 스택 멀티모달 주의 기제 기법(Stacked Multimodal Attention)을 사용하여 훈련을 진행한다. 구조는 Fig. 4와 같고, 수식 (5)의 경우 Fig. 5에서 첫 번째 Masked 멀티-헤드 주의 기제 기법을 나타낸다.

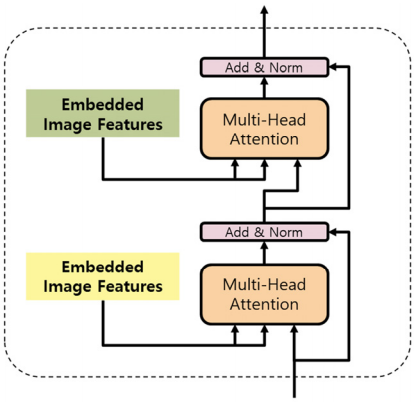

그림 4 스택 멀티모달 주의 기제 기법
Fig. 4 Stacked Multimodal Attention

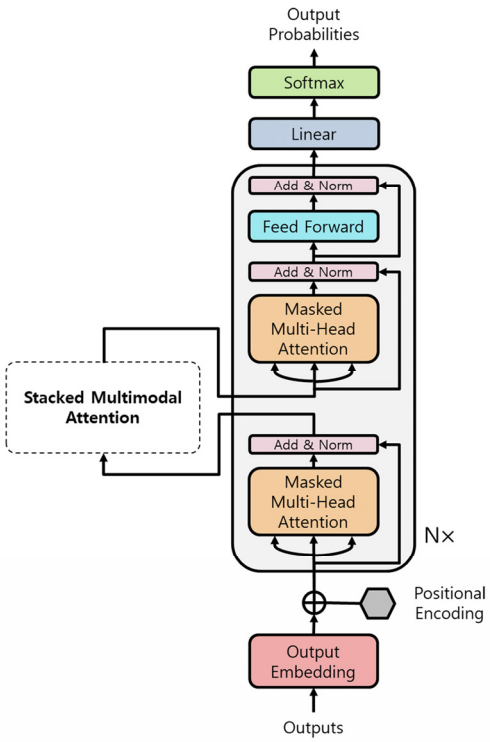

그림 5 자가 교열 트랜스포머
Fig. 5 Self-revising Transformer

3.4.2 자가 교열 트랜스포머

본 연구에서는 생성되는 문장의 완성도를 높이기 위해 언어 모델을 강화한 자가 교열 트랜스포머(Self-Revising Transformer)를 제안한다. 기존의 트랜스포머 모델은 다음 단어를 예측할 때 입력 문장의 인코더 결과 값과 현재 단계의 단어들과 멀티-헤드 주의 기제 기법을 통해 구한다. 이와는 달리, 이미지 캡션 생성 연구는 인코더의 입력 값이 이미지 특징 벡터이기 때문에 다음 단어를 예측할 때 이미지의 정보를 통해서 생성하게 된다. 결과적으로 생성한 문장의 완성도가 떨어지거나 일반적이게 되는 점을 보완해주기 위해서 Fig. 5에서 나타나는 것처럼, 디코더 레이어의 마지막 포지션-와이즈 순방향 네트워크를 계산하기 전에 추가적인 Masked 멀티-헤드 주의 기제 기법을 더한다. Masked 주의 기제를 추가함으로써, 이전 단계에서 얻은 h_3^l을 자가-주의 기제하여 단어들 사이의 관계에 대한 정보를 한 번 더 가중치에 업데이트한다. 이를 통해 단어의 중복된 표현을 제재하고, 앞선 네트워크에서 생성된 문장을 재구축하여 문장의 연결성을 높인다.

3.5 훈련 방법

본 연구는 두 종류의 손실 함수(Loss Function)를 사용하여 모델을 학습시킨다. Fig. 6과 같이 라벨 스무딩(Label Smoothing)[31]을 사용한 KLDiv(Kullback-Leibler divergence) 손실 함수로 모델을 훈련한 후에, 강화학습으로 모델을 최적화 시키는 SCST의 방법론을 적용하여 성능을 더욱 향상시킨다.

첫 번째로 라벨 스무딩이 적용된 KLDiv 손실함수는 아래와 같이 계산된다.

$$L_{KLDiv} = -\sum_x p'(x) \log \frac{p'(x)}{q(x)} \tag{9}$$

$$p'(x) = (1-\epsilon)p(x) + \frac{\epsilon}{X} \tag{10}$$

이때, 수식 (10)는 라벨 스무딩을 적용하는 식으로, 모델이 정답을 정확하게 예측하지 않게 억압하는 방식으로 모델을 정규화(Regularization)한다. $p(x)$는 정답 분포(Distribution)이고, $q(x)$는 모델이 예측한 분포, 그리고 $\epsilon = 0.2$는 라벨 스무딩의 인자[31]이다.

두 번째로는 앞의 손실 함수로 훈련한 모델을 강화학습 SCST를 이용하여 최적화한다.

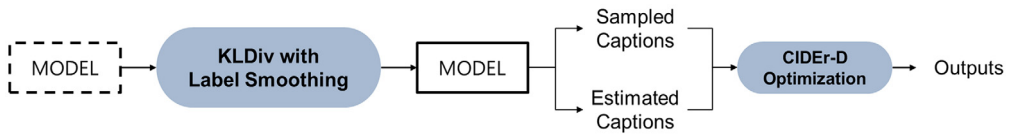

그림 6 모델 손실 함수 흐름도
Fig. 6 A Flow Chart of the Model Loss Function

$$L_{RL}(\theta) = -\mathbb{E}_{y^s \sim p(\theta)}[r(y^s)] \quad (11)$$

$$\nabla_\theta L_{RL}(\theta) \approx -(r(y^s) - r(\hat{y}))\nabla_\theta \log p_\theta(y^s) \quad (12)$$

수식 (11)은 SCST의 계산 식으로, Fig. 6과 같이 추론 알고리즘을 이용한 예측 캡션(Estimated Captions)과 샘플링된 캡션(Sampled Captions)의 CIDEr-D 스코어[21]를 각각 구하여, 두 스코어의 오차를 구하는 리워드 r을 통해 모델을 훈련시킨다. 그라디언트 값을 계산하는 수식 (12)에서 y^s는 샘플링된 예측 분포이고, $\hat{y}$는 추론 알고리즘 탐욕 디코딩(Greedy Decoding)을 통해 얻은 예측 분포이다.

4. 실험 및 결과

4.1 데이터셋

본 연구는 실험을 위해서 이미지 캡션 생성 연구에 쓰이는 MSCOCO 데이터셋[32]을 사용한다. MSCOCO 데이터셋은 평균적으로 한 이미지당 5개의 이미지를 설명하는 캡션이 존재하고, 훈련 데이터셋으로 82,783장의 이미지, 평가 데이터셋으로 40,504장의 이미지를 제공한다. 총 123,287장의 이미지를 훈련, 평가, 그리고 성능 평가를 위한 테스트 데이터셋으로 세 종류로 분할하여 사용하였고, [33]에서 성능 평가 비교를 위해 배포하는 분할된 데이터셋 정보를 이용하였다. [33]는 113287장의 훈련 데이터셋, 각 5000장의 평가, 테스트 데이터셋의 분할 정보를 제공한다. 단어 사전(Vocabulary)은 단어가 등장하는 최소 횟수가 5이상인 단어들로 구성하여 총 9487개로 구성한다.

4.2 훈련 세부사항

특징 추출기로는 Faster R-CNN과 Mask R-CNN 두 종류를 사용한다. Faster R-CNN으로 얻은 특징 벡터[9]는 2048차원이고 Mask R-CNN으로 얻은 특징 벡터는 4096차원이다. 이 두 가지의 특징 벡터들은 인코더에서 $d_{model} = 1024$의 차원으로 축소된다. Mask R-CNN 특징 벡터의 경우 4096에서 d_{model}로 직접적으로 축소하지 않고 한 단계를 더 거친다. 모델은 2개의 레이어를 사용한다. 훈련 중 사용된 최적화 알고리즘은 아담(Adam Optimizer)[34]이고, 손실함수는 라벨 스무딩을 정규화 방법으로 적용한 KLDiv 손실함수를 사용하여 25 에폭을 훈련한다. 훈련이 끝난 후, CIDEr-D 스코어를 리워드로 사용하는 SCST 방법론을 이용하여 15 에폭을 추가로 훈련하여 모델을 최적화한다. 모델 성능 평가를 위한 추론 알고리즘은 빔 탐색 디코딩(Beam Search Decoding)을 사용하였고, 빔의 크기는 2이다.

4.3 정량적 평가

본 연구는 BLEU[35], METEOR[36], ROUGE-L[37], CIDEr-D[21], SPICE[38] 스코어를 사용하여 모델을 평가한다. 성능 비교에 사용된 모델들은 SCST로 최적화가 완료된 모델로 스코어를 계산한다.

Table 1은 선행 연구와의 성능 비교를 보여준다. 본 연구가 제안하는 자가 교열 트랜스포머 모델은 SRT로 표기하고, FRC와 MRC는 사용된 특징 추출기로 각 Faster R-CNN과 Mask R-CNN의 약어를 나타낸다. 모델 명은 스택 멀티모달 주의 기제 기법 안의 레이어 하단에 위치한 특징 추출기의 이름을 앞에 표기한다. 예를 들면, Mask R-CNN이 하단에 위치한 경우 MRC-FRC로 표기한다. 또한, 자가 교열 트랜스포머를 SRT로 축약하여 사용한다. 선행 연구와 비교했을 때, MRC-FRC+SRT 모델의 경우 BLEU 1 스코어와 CIDEr-D를 제외한 모든 스코어에서 높은 성능을 보였고, FRC-MRC+SRT 모델의 경우에도 대체적으로 우수한 성능을 보였다. 결과적으로 Table 1을 통해, 제안하는 모델이 선행연구보다 성능이 우수함을 정량적으로 보여준다.

Table 2는 제안하는 모델의 이미지 특징 추출기에 따른 비교 실험이다. 모든 모형은 KLDiv 손실 함수를 사용하여 훈련한 후, CIDEr-D 스코어를 이용하여 최적화한다. 비교 모델들의 디코더는 동일하게 자가 교열 트랜스포머를 사용하였다. 결과를 통해 SCST로 최적화를 거친 모델들의 성능 향상을 보여준다. 특징 추출기가 MRC 하나 만을 사용한 모델의 결과의 경우 성능이 좋지 않다. 이는 Mask R-CNN 모델은 인스턴스 분할 연구 분야의 특화되어 있기 때문에 이미지 캡션 생성 연구의 특징 추출기로 사용하기 적합하지 않다는 것을 보여준다. 그러나 Faster R-CNN과 Mask R-CNN의 다중 특징 추출기를 사용한 모델들은 더 많은 이미지 정보를 디코더에 전달해주기 때문에 성능이 향상되었다. FRC-MRC 모델은 FRC 모델과 비교했을 때 대체적으로 성능이 올랐고, MRC-FRC 모델은 모든 스코어에서 성능의 향상을 보였다. 이는 다중 관점을 갖는 모델이 단일 관점을 갖는 모델에 비해 성능이 개선됨을 나타낸다. 추가적으로 서로 다른 특징 추출기를 사용한 것이 같은 특징 추출기를 사용하는 것보다 효과적임을 알기 위해서 FRC-FRC 모델로 비교 실험을 진행하였다. 결과적으로 Faster R-CNN을 하나만 사용한 모델보다 성능이 낮았고, 본 연구가 제안하는 다른 추출기의 사용이 더 효과적임을 알 수 있다.

Table 3은 FRC-MRC와 MRC-FRC 모델에 대해서 자가 교열 트랜스포머 유무에 대한 비교 실험이다. Base는 디코더로 SRT가 아닌 트랜스포머를 사용한 모델을 나타낸다. Table 3 안의 굵은 글씨는 각 모델 안에서 성능이 높은 점수를 나타낸다. 두 모델 모두에서 자가 교열 트랜스포머를 사용한 모델이 우수했다. FRC-MRC 모델의 경우 BLEU 1 스코어를 제외한 모든 스코어가 올랐

표 1 선행 연구와의 성능 비교
Table 1 Performance comparison with previously studied models

Metric	B@1	B@4	M	R	C	S
LSTM[12]	-	31.9	25.5	54.3	106.3	-
SCST[20]	-	34.2	26.7	55.7	114	-
LSTM-A[22]	78.6	35.5	27.3	56.8	118.3	20.8
Up-Down[9]	79.8	36.3	27.7	56.9	120.1	21.4
RFNet[23]	79.1	36.5	27.7	57.3	121.9	21.2
ICSA[19]	80.2	38	28.6	58.4	**128.6**	22.1
GCN-LSTM[24]	80.5	38.2	28.5	58.3	127.6	22
SGAE[25]	**80.8**	38.4	28.4	58.6	127.8	22.1
ORT[17]	80.5	38.6	28.7	**58.7**	128.3	22.6
FRC-MRC+SRT	79.8	38.3	**29**	58.5	128.2	**22.8**
MRC-FRC+SRT	80.4	**38.7**	**29**	**58.7**	128.4	**22.8**

표 2 이미지 특징 추출기의 성능 비교
Table 2 Performance comparison of image feature extractors

Model	KL Divergence Loss						CIDEr-D Score Optimization					
Feat. Extractor	B@1	B@4	M	R	C	S	B@1	B@4	M	R	C	S
FRC	71.6	29.4	24.9	48.2	99.2	19.6	80.1	38.3	28.7	58.5	127.2	22.6
MRC	58.1	18.3	18.4	39.2	58.8	12.4	66.6	25.3	21.9	47.3	82.8	14.8
FRC-FRC	68.7	27.9	24.2	48.5	99.4	18.4	75.1	34.5	27.7	54.1	123.3	21.6
FRC-MRC	**76.7**	**35.1**	**27**	**55.8**	**110.7**	**20.5**	79.8	38.3	**29**	58.5	128.2	**22.8**
MRC-FRC	70.8	29.8	24.9	49.5	101	19.2	**80.4**	**38.7**	**29**	**58.7**	**128.4**	**22.8**

표 3 디코더 모듈의 성능 비교
Table 3 Performance comparison of decoder modules

Model		CIDEr-D Score Optimization					
Feat. Extractor	Dec	B@1	B@4	M	R	C	S
FRC-MRC	Base	80.1	38.2	28.6	58.4	126.6	22.4
	+ SRT	79.8	**38.3**	**29**	**58.5**	**128.2**	**22.8**
MRC-FRC	Base	80.1	38.2	28.8	58.5	127.8	22.7
	+ SRT	**80.4**	**38.7**	**29**	**58.7**	**128.4**	**22.8**

표 4 인코더 디코더 결합 방법론의 성능 비교
Table 4 Performance comparison of methods combining an encoder and a decoder

Model		CIDEr-D Score Optimization					
Feat. Extractor	Methods	B@1	B@4	M	R	C	S
FRC-MRC	Stacked attn	79.8	38.3	**29**	58.5	128.2	22.8
MRC-FRC		**80.4**	**38.7**	**29**	**58.7**	**128.4**	**22.8**
FRC-MRC	Stacked layer	79.8	38.1	28.6	58.2	126.2	22.4
MRC-FRC		74.7	34.1	27.7	54	122.7	21.5
FRC+MRC	Parallel attn	78.7	36.9	28.2	56.6	128	21.9

고, MRC-FRC 모델은 모든 스코어에서 성능의 향상을 보인다. 특히, CIDEr-D 스코어가 두 모델에서 크게 증가하였다. 이를 통해 자가 교열 트랜스포머는 모델 성능 개선에 긍정적인 영향을 보여준다.

Table 4는 3.4.1에서 언급했던 인코더와 디코더의 결합 방법론에 대한 비교 실험이다. Fig. 3의 (a)는 Table 4의 methods에서 Stacked attn에 해당하고 (b)는 Stacked layer, 그리고 (c)는 Parallel attn을 가리킨다. 결과를 살펴보면, Stacked attn의 경우가 다른 방법론들에 비해 모든 스코어에서 높은 점수를 보인다. Stacked layer를 사용한 경우는 FRC-MRC 모델이 MRC-FRC 모델보다 성능이 높았다. 그리고 Parallel을 사용한 모델은 스코어들을 종합적으로 보았을 때, Stacked layer와 비슷한 성능을 나타낸다. 예를 들면 Parallel 모델은 METEOR 스코어만 가장 높고, 그 외에는 FRC-MRC 모델과 근사하지만 낮은 점수를 보인다. 그러므로 실험

적으로 보았을 때, 주의 기제 기법을 쌓는 (a) 방법론이 제안하는 모델에서 효과적이란 것을 알 수 있다.

4.4 정성적 평가

Fig. 7은 본 연구의 비교실험 모델들의 이미지에 대한 캡션 생성 예시이다. (a)는 캡션이 올바르게 생성된 경우이고, (b)는 모델이 잘못된 캡션을 생성한 경우이다. 모델의 디코더는 모두 자가 교열 트랜스포머를 사용한다. 생성된 캡션과 같이, 다중 특징 추출기를 사용한 모델이 주어진 이미지의 특징들을 더욱 세부적으로 포착하여 서술하고 있다.

(a)의 첫 번째 이미지는 세 가지 모델 모두 이미지를 잘 설명한 것을 볼 수 있다. 생성된 캡션을 보면, FRC+SRT 모델은 새가 작다는 것과 벤치가 하얗다는 특징을, FRC-MRC+SRT 모델은 새가 작다는 것과 벤치가 나무로 만들어 졌다는 것을, 그리고 MRC-FRC+SRT 모델은 노란 새가 나무로 만들어진 벤치에 있다는 특징을 잘 잡아냈다. 그러나 FRC+SRT 모델은 이미지에 대한 캡션을 대체적으로 잘 생성하지만, 다중 특징 추출기를 쓴 모델이 생성한 캡션의 퀄리티가 더 높은 경우가 많다. 두 번째, 세 번째 이미지의 경우, 나머지 두 모델은 제트기의 정확한 색상과 개수, 양 이외의 동물인 개를 포착한 것에 비해, FRC+SRT 모델이 생성한 캡션은 이미지에 대해 비교적 일반적인 설명을 출력하였다. 이를 통해 다중 특징 추출기의 사용은 이미지의 특징을 포착하는 성능의 향상을 보여준다.

(b)의 이미지들은 잘못된 캡션을 생성한 예시들을 나타낸다. 첫 번째 이미지에서 FRC+SRT 모델과 MRC-FRC+SRT 모델은 이미지 안의 객체를 세부적으로 찾아냈지만, 캡션의 주체가 자전거를 타는 사람인 이미지에서 중요하지 않는 서핑 보드를 포착하여 캡션 생성에 영향을 끼쳐 틀린 묘사를 한다. 또한, 모델이 이미지 안의 객체를 비슷한 다른 객체와 혼동하여 잘못 생성하는데, 두 번째 이미지에서는 이미지에 없는 바나나와 물고기라는 객체를 인식하여 잘못된 캡션을 생성하고 세 번째 이미지는 인물이 여자임에도 불구하고 남성으로 인식하고 우산을 야구 방망이로 다르게 인식하고 있다.

Fig. 8은 자가 교열 트랜스포머의 유무에 대한 예시이다. 예시의 Base는 MRC-FRC 특징 추출기를 사용하고, 디코더는 기존의 트랜스포머를 사용한 모델을 나타낸다. 생성된 캡션과 같이, 자가 교열 트랜스포머를 사용한 모델이 이미지를 더욱 자세히 설명하고 있다.

왼쪽에서 첫 번째 이미지를 보면, 남자가 파란색 곰인형을 안고 누워있는 이미지이다. 트랜스포머를 쓴 모델이 생성한 예시를 번역하면 “곰인형을 안고 의자에 누워있는 남자” 이지만, 자가 교열 트랜스포머를 사용한 모델은 “곰인형을 안고 침대에 누워있는 남자”이다. 전자의 모델에서는 “의자에 누워있다”와 같은 어색한 표현을 생성하였으나, 본 모델에서는 “침대에 누워있다”는 캡션을 생성하여 단어 간의 연결성을 개선했음을 보여준다. 두 번째 이미지의 경우, 두 모델의 캡션은 어순이

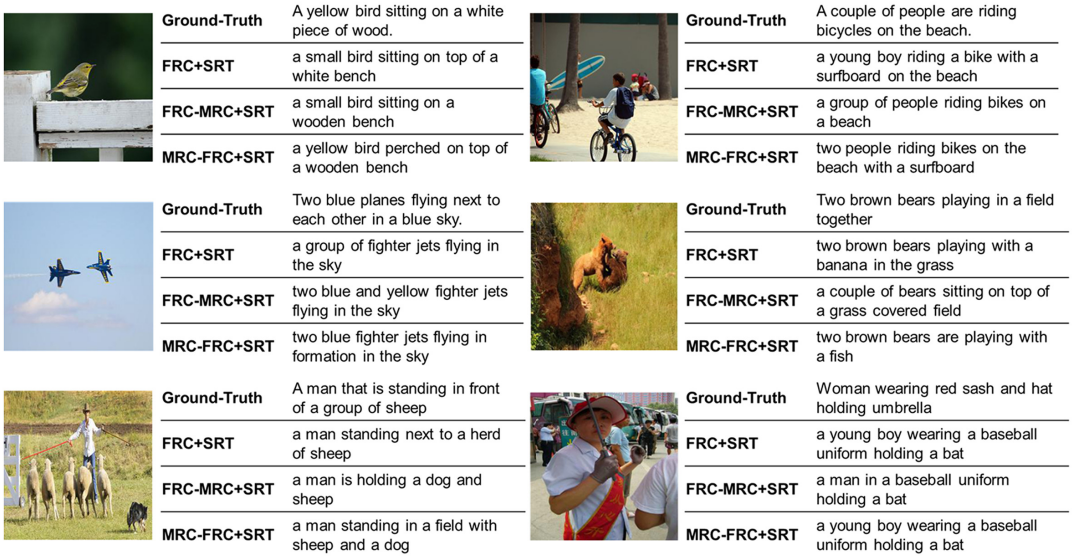

그림 7 특징 추출기에 따른 모델이 생성한 문장 예시

Fig. 7 Examples of the Sentences Generated by Models According to Feature Extractors

Base a man laying in a chair holding a teddy bear
+SRT a man laying on a bed with a teddy bear

Base a man standing on a tennis court with a racket
+SRT a man holding a tennis racket on a tennis court

Base a woman standing in front of an umbrella
+SRT a woman sitting on top of a small shack

Base a group of stuffed teddy bears sitting on a table
+SRT a group of stuffed animals sitting on a table with books

그림 8 자가 교열 트랜스포머를 이용한 비교 예시
Fig. 8 Comparative Examples using Self-revising Transformer

다르지만 대체로 비슷한 단어들을 활용하고 있다. 세 번째 이미지에서는 앞의 모델은 객체를 잘못 포착했지만 제안하는 모델은 비슷한 객체를 설명하는 예시이다. 정답 예시의 경우 해당 물체를 정자 및 오두막으로 표현하였다. 트랜스포머 모델을 쓴 모델은 우산이라고 설명하였지만, 본 연구가 제안하는 모델은 비슷한 단어인 판잣집이라고 설명한다. 마지막 이미지는 생성된 캡션을 번역하면 “탁자 위에 앉아 있는 곰 인형들”과 “책과 함께 탁자 위에 앉아 있는 봉제 완구들”이다. 두 예시를 비교하면 마찬가지로 자가 교열 트랜스포머를 사용한 모델이 이미지를 잘 설명하고 있다. 예시를 통해 자가 교열 트랜스포머의 사용은 생성된 문장의 연결성 및 이미지 설명의 정확도를 높여주는 것을 볼 수 있다.

5. 결 론

본 연구에서는 세부적인 묘사와 더불어 캡션의 높은 완성도를 위해, 다중 관점 인코더와 자가 교열 트랜스포머를 제안한다. 이미지 캡션 생성의 선행 연구는 하나의 특징 추출기를 통해 순환 신경망을 언어 모델로 사용하여 캡션을 생성한다. 그러나 하나의 특징 추출기를 사용하여 단일 관점을 갖고, 순환 신경망의 사용으로 장기 의존성을 갖는 한계가 있다. 이러한 한계를 해결하기 위해 다중 관점을 갖는 인코더를 제안하여 서로 다른 각도의 이미지 정보를 모델에 제공한다. 또한, 순환 신경망을 사용하지 않고 자가 교열 트랜스포머를 제안함으로써, 캡션의 연결성과 정확도를 얻는다. 제안하는 모델의 성능을 검증하기 위해 MSCOCO 데이터셋으로 실험하고, 여러 스코어를 통해 정량적 평가를 진행하여 선행 연구보다 성능이 우수함을 보였다. 그리고 예시를 통해 제안한 방법론이 모델의 성능 개선에 긍정적인 영향을 끼침을 보여준다.

향후에는 스택 멀티모달 주의 기제 안의 특징 추출기의 순서에 대한 중요성을 확인하기 위해 모델의 구조를 결정지을 수 있는 메타 학습 기법을 활용해볼 예정이다. 또한, MSCOCO와 같은 일반적인 이미지에 대한 데이터셋이 아닌 특정한 성격을 갖는 도메인의 데이터셋에 대해 실험을 수행할 계획이다.

References

[1] S. Wu, J. Wieland, O. Farivar, and J. Schiller, "Automatic alt-text: Computer-generated image descriptions for blind users on a social network service," *Proc. of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*, pp. 1180-1192, 2017.

[2] V.-K. Vo-Ho, Q.-A. Luong, D.-T. Nguyen, M.-K. Tran, and M.-T. Tran, "A Smart System for Text-Lifelog Generation from Wearable Cameras in Smart Environment Using Concept-Augmented Image Captioning with Modified Beam Search Strategy," *Applied Sciences*, pp. 1886, 2019.

[3] B. Mishra, D. Garg, P. Narang, and V. Mishra, "Drone-surveillance for search and rescue in natural disaster," *Computer Communications*, 2020.

[4] A. Krizhevsky, I. Sutskever, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, pp. 1097-1105, 2012.

[5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 1-9, 2015.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *Proc. of the IEEE conference on computer vision and pattern recog-*

nition, pp. 770-778, 2016.

[8] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, pp. 91-99, 2015.

[9] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 6077-6086, 2018.

[10] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 3156-3164, 2015.

[11] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[12] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," *International conference on machine learning*, pp. 2048-2057, 2015.

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," *Proc. of the IEEE international conference on computer vision*, pp. 2961-2969, 2017.

[14] I. Sutskever, O. Vinyals, and Q.V. Le, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, pp. 3104-3112, 2014.

[15] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 375-383, 2017.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, pp. 5998-6008, 2017.

[17] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, "Image captioning: Transforming objects into words," *Advances in Neural Information Processing Systems*, pp. 11137-11147, 2019.

[18] X. Zhu, L. Li, J. Liu, H. Peng, and X. Niu, "Captioning transformer with stacked attention modules," *Applied Sciences*, pp. 739, 2018.

[19] L. Zhenru, L. Yaoyi, and L. Hongtao, "Improve Image Captioning by Self-attention," *International Conference on Neural Information Processing*, *Springer*, *Cham*, pp. 91-98, 2019.

[20] S.J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7008-7024, 2017.

[21] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 4566-4575, 2015.

[22] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," *Proc. of the IEEE International Conference on Computer Vision*, pp. 4894-4902, 2017.

[23] W. Jiang, L. Ma, Y.-G. Jiang, W. Liu, and T. Zhang, "Recurrent fusion network for image captioning," *Proc. of the European Conference on Computer Vision (ECCV)*, pp. 499-515, 2018.

[24] T. Yao, Y. Pan, Y. Li, and T. Mei, "Exploring visual relationship for image captioning," *Proc. of the European conference on computer vision (ECCV)*, pp. 684-699, 2018.

[25] X. Yang, K. Tang, H. Zhang, and J. Cai, "Auto-encoding scene graphs for image captioning," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 10685-10694, 2019.

[26] L. Ruotian, "A Better Variant of Self-Critical Sequence Training," *arXiv preprint arXiv:2003.09971*, 2020.

[27] S. Yan, Y. Hua, and N.M. Robertson, "Off-Policy Self-Critical Training for Transformer in Visual Paragraph Generation," *arXiv preprint arXiv: 2006.11714*, 2020.

[28] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 580-587, 2014.

[29] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," *2009 IEEE conference on computer vision and pattern recognition*, *Ieee*, pp. 248-255, 2009.

[30] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, and D.A. Shamma, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International journal of computer vision*, pp. 32-73, 2017.

[31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 2818-2826, 2016.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C.L. Zitnick, "Microsoft coco: Common objects in context," *European conference on computer vision*, *Springer*, pp. 740-755, 2014.

[33] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," *Proc. of the IEEE conference on computer vision and pattern recognition*, pp. 3128-3137, 2015.

[34] D.P. Kingma and J. Ba, "Adam: A method for

stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
[35] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," *Proc. of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311-318, 2002.
[36] S. Banerjee and A. Lavie, "METEOR: An automatic metric for MT evaluation with improved correlation with human judgments," *Proc. of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65-72, 2005.
[37] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," *Text summarization branches out*, pp. 74-81, 2004.
[38] P. Anderson, B. Fernando, M. Johnson, and S. Gould, "Spice: Semantic propositional image caption evaluation," *European Conference on Computer Vision, Springer*, pp. 382-398, 2016.

이 지 은

2019년 인천대학교 컴퓨터공학부(학사) 2019년~현재 연세대학교 컴퓨터과학과 석박사통합과정. 관심분야는 빅데이터마이닝 & 기계 학습

박 진 욱

2016년 서울시립대학교 통계학과(학사) 2017년~현재 연세대학교 컴퓨터과학과 석박사통합과정. 관심분야는 빅데이터마이닝 & 기계 학습

박 상 현

1989년 서울대학교 컴퓨터공학과 졸업(학사). 1991년 서울대학교 대학원 컴퓨터공학과(공학석사). 2001년 UC-LA 대학원 컴퓨터과학과(공학박사). 1991년~1996년 대우통신 연구원. 2001년~2002년 IBM T. J. Watson Research Center Post-Doctoral Fel-low. 2002년~2003년 포항공과대학교 컴퓨터공학과 조교수. 2003년~2006년 연세대학교 컴퓨터과학과 조교수. 2006년~2011년 연세대학교 컴퓨터과학과 부교수. 2011년~현재 연세대학교 컴퓨터과학과 교수. 관심분야는 데이터베이스, 데이터마이닝, 바이오인 포매틱스, 빅데이터 마이닝 & 기계 학습