홍 기 형 교수 지도

석사학위 청구논문

Visual Encoder를 적용한 딥러닝 기반 보완대체의사소통 그림 상징 시퀀스의 한국어 문장 생성

2025

성신여자대학교 대학원

미래융합기술공학과

이 지 원

Visual Encoder를 적용한 딥러닝 기반 보완대체의사소통 그림 상징 시퀀스의 한국어 문장 생성

홍 기 형 교수 지도

이 논문을 석사 학위 논문으로 제출함

2024년 11월

성신여자대학교 대학원

미래융합기술공학과

이 지 원

인 준 서

이지원의 석사학위 논문으로 인준함

2025년 1월

심사위원장 변 혜 원 (서명 또는 인)

심 사 위 원 오 장 민 (서명 또는 인)

심 사 위 원 홍 기 형 (서명 또는 인)

성신여자대학교 대학원

논 문 개 요

보완대체의사소통(Augmentative and Alternative Communication, AAC)은 언어를 표현하고 이해하는 데 결함이 있는 장애인을 위한 의사소통 보조 도구이며, 대표적으로는 AAC 그림 상징 체계가 있다. 언어 장애인은 1개 이상의 AAC 그림 상징을 선택하여 메시지를 구성할 수 있으며, 이를 AAC 그림 상징 시퀀스라고 한다.

다양한 모바일 어플리케이션을 통해 AAC를 활용한 의사소통을 지원하고 있으나 자연스러운 발화와 전달의 측면에서 언어 장애인과 비장애인이 원활히 의사소통하는 데는 여전히 한계가 있다. 일상 대화, 회의, 주문, 결제, 진료 등 다양한 서비스가 온라인으로 전환됨에 따라 비대면 의사소통의 어려움 또한 증가하였다. 이러한 언어 장애인과 비장애인의 의사소통 문제를 보조하기 위해 AAC 그림 상징 시퀀스와 한국어 문장 간 변환 기능이 요구되며, 관련 연구가 활발히 이어지고 있다. 그러나 연구의 대부분이 AAC 그림 상징의 어휘와 식별자 정보를 기반으로 하며 AAC 그림 상징의 이미지 정보를 활용하지 않는다.

본 논문은 AAC 그림 상징 시퀀스의 이미지를 기반으로 한국어 문장을 생성하는 모델을 제안한다. 상징 시퀀스 이미지 데이터셋 구축을 위해 기존 연구의 상징 시퀀스-한국어 문장 데이터셋을 활용하였다. AAC 상징 시퀀스의 이미지들을 하나의 이미지로 병합하여 상징 시퀀스 이미지-한국어 문장의 최종 데이터셋을 구축하였다. 이미지에서 하나의 상징이 차지하는 영역의 크기를 고려하여, AAC 상징 시퀀스의 최대 길이가 4인 경우와 6인 경우로 분류하고 각각 2×2, 3×2의 그리드 형태로 결합한다. 이미지 특징 추출을 위한

Visual Encoder로는 사전 학습된 합성곱 신경망인 ResNet-101, Inception-v3, EfficientNet-B3 모델을 활용하였다. 추출한 이미지 특징에 기반해 한국어 문장을 생성하는 Text Decoder로는 순환 신경망인 LSTM(Long Short-Term Memory)과 GRU(Gated Recurrent Unit)를 선정하였으며, 문장의 각 단어를 생성하는 시점에 중요한 이미지 영역에 가중치를 둘 수 있도록 Attention Mechanism을 적용하였다. Visual Encoder 3가지, Text Decoder 2가지의 조합으로 총 6가지 모델을 실험하였다. 성능 평가 지표로는 기계 번역과 참조 번역의 유사성을 측정하는 BLEU(Bilingual Evaluation Understudy)를 사용하였다. Visual Encoder로는 ResNet-101을 사용했을 때 우수한 성능을 보였으며, Text Decoder로는 GRU 구조보다 LSTM 구조의 모델을 사용했을 때 전반적으로 더 우수한 성능을 보였다. ResNet-101과 EfficientNet-B3를 Visual Encoder로 사용하고 LSTM을 Text Decoder로 사용한 모델에서는 2×2와 3×2의 병합 이미지에 대해 유사한 성능을 보임을 확인하였다. 6개 모델의 예측 결과에 대한 BLEU 점수는 모두 60 ~ 80점대에 분포하여 고품질의 번역 성능을 보였다.

목 차

그 림 목 차

표 목 차

Ⅰ. 서 론

보완대체의사소통(Augmentative and Alternative Communication, AAC)은 언어 장애인의 의사소통을 보조하기 위한 수단으로, 대표적인 AAC 체계로는 AAC 그림 상징이 있다[1, 2]. AAC 그림 상징은 사용자가 표현하고자 하는 어휘를 그림 또는 사진으로 나타낸 것이다. 언어 장애인은 시각적으로 제시되는 AAC 그림 상징을 보고, 그것이 표상하는 의미를 기억하는 인식(recognition) 과정을 거친다. 해당 상징이 자신이 말하고자 하는 개념과 일치하는 경우 이를 선택하여 메시지를 구성하고, 음성 합성을 이용하여 타인과 의사소통할 수 있다[3]. 사용자의 AAC 활용 능력에 따라 1개 이상의 그림 상징으로 메시지를 구성하며, 이를 AAC 그림 상징 시퀀스라고 한다.

언어 장애인에게 친숙한 AAC 그림 상징 시퀀스는 비장애인이 사용하는 구어 체계와 일치하지 않기 때문에, 실제 상황에서 언어 장애인과 비장애인의 원활한 의사소통을 위한 보조 도구가 필요하다. 특히, 스마트폰, 패드, PC 등 디지털 기기의 보급률이 증가하고 코로나 19 팬데믹 이후 일반 대화, 상품 주문, 금융 거래 등 다양한 생활 양식이 온라인으로 전환되면서 비대면 의사소통의 필요성이 급격히 확대되었다. 이는 언어 장애인에게 기존 AAC 시스템을 넘어, 다양한 디지털 환경에서 적합한 의사소통 도구를 제공할 필요성을 더욱 부각시킨다[4, 5]. 따라서, 언어 장애인에게 친숙한 AAC 그림 상징 시퀀스와 한국어 문장 간 변환 기능을 지원할 필요가 있다.

언어 장애인과 비장애인의 원활한 의사소통을 위해 AAC 그림 상징 시퀀스와 한국어 문장 간 변환 연구가 지속되었다. [6, 7]의 연구는 AAC 그림 상징

시퀀스를 자연스러운 한국어 문장으로 변환하는 것을 목적으로 하며, [7]의 연구에서는 AAC 그림 상징이 여러 의미를 지닐 수 있는 의미적 모호성을 해소하고자 하였다. 두 연구는 공통적으로 AAC 그림 상징의 이미지 정보를 배제하고 상징의 어휘 또는 식별자 정보를 활용하였다. AAC 그림 상징이 갖는 다중 의미를 효과적으로 반영하기 위해서는 AAC 그림 상징의 이미지 정보를 활용할 필요가 있다.

본 논문에서는 AAC 그림 상징 시퀀스의 이미지 정보를 활용하여 한국어 문장을 생성하고자 한다. 시퀀스를 구성하는 각각의 AAC 그림 상징 이미지를 하나의 이미지로 병합하였으며, 모델은 이미지를 해석하기 위한 Visual Encoder와 이미지 정보를 기반으로 텍스트를 생성하기 위한 Text Decoder로 구성하였다. 또한, Attention Mechanism[10]을 적용하여 각 단어를 생성하는 시점에 이미지의 특정 영역에 집중하도록 가중치를 부여해 학습 효율성을 높였다. Visual Encoder는 사전 학습된 합성곱 신경망 모델인 ResNet-101[11], Inception-v3[12], EfficientNet-B3[13]을 사용하고, Text Decoder는 LSTM(Long-Short Term Memory)[14], GRU(Gated Recurrent Unit)[15]를 사용하여 총 6가지 모델을 비교 실험하였다.

본 논문의 흐름은 다음과 같다. 2장에서는 보완대체의사소통 그림 상징을 소개하고, 딥러닝 기반 AAC 그림 상징 시퀀스의 한국어 문장생성에 관한 기존 연구를 다룬다. 또한, 이미지를 이해하고 텍스트를 생성한다는 점에서 본 연구와 유사한 이미지 캡셔닝 모델을 소개한다. 3장에서는 데이터셋 구축 과정을 정리하고, 모델을 설계한다. 기존 연구의 학습 데이터셋을 활용해 보완대체의사소통 그림 상징 시퀀스 이미지와 한국어 문장의 쌍으로 데이터셋을 구축하였으며, 이를 학습하기 위한 6가지 모델을 설명한다. 4장에서는 앞서 설

계한 6가지 모델을 실험하고, BLEU(Bilingual Evaluation Understudy)[16] 점수를 평가 지표로 하여 각 모델의 성능을 비교한다. 5장에서는 본 논문의 결론과 향후 연구를 기술한다.

Ⅱ. 관련 연구 및 이론적 배경

1. 보완대체의사소통 그림 상징

보완대체의사소통(Augmentative and Alternative Communication, AAC)[1]은 구어를 통한 의사소통에 어려움이 있는 장애인을 위한 의사소통 보조 도구이다. AAC 체계는 표정, 제스쳐, 수화 등의 비도구적 상징 체계와 그림, 사진, 물체 등으로 나타낼 수 있는 도구적 상징 체계로 나눌 수 있다[1-3]. 도구적 상징 체계 중 AAC 그림 상징은 그림(Graphic), 식별자(ID), 어휘(Expression)로 구성되며, [그림 1]은 AAC 그림 상징의 예시이다.

[그림 1] 보완대체의사소통 그림 상징의 구성 예시

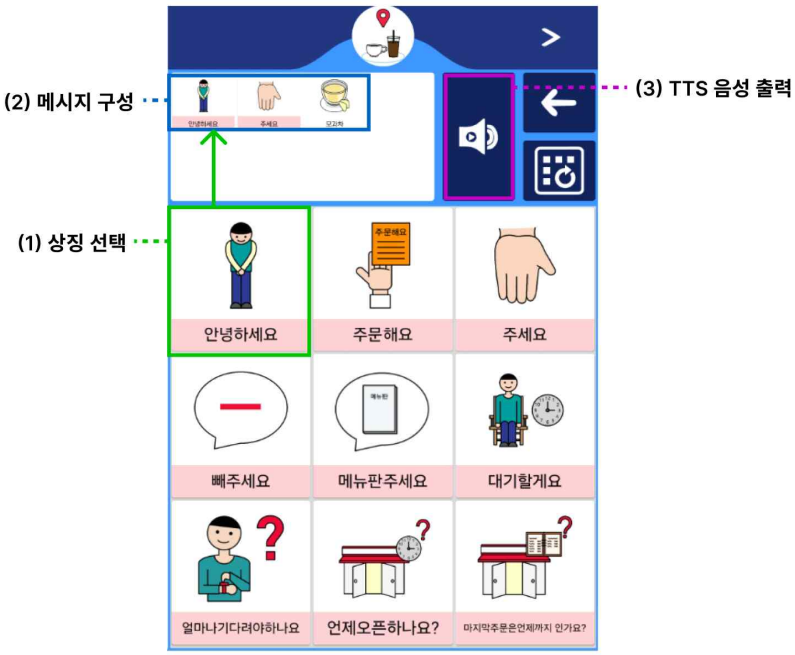
AAC 그림 상징의 어휘(Expression)는 AAC 사용자가 많이 쓰는 표현을 중심으로 단어, 구, 문장 형태로 구성된다. AAC 그림 상징의 그림(Graphic)은 해당 어휘, 개념을 직관적으로 보여주는 도상성(iconicity)을 띈다.

국내에서는 대표적으로 한국형 AAC 그림 상징 체계집[2]이 있으며, [그림 2]는 한국형 AAC 그림 상징의 예시이다. 영유아부터 성인기까지 다양한 연령대의 AAC 사용자에게 필요한 어휘로 구성되며, 국내의 고유 명절, 식생활, 장소명 등 한국 문화를 반영하고 있다. 또한, AAC 사용자의 장애 유형과 다양한 연령대에서 필요한 어휘를 포함하고 있다.

[그림 2] 한국형 보완대체의사소통 상징 체계집의 그림 상징과 어휘 예시

스마트폰, 태블릿 등 모바일 기기의 활용도가 높아짐에 따라 마이토키[17], GeoAAC[18], 나의 AAC[19] 등 다양한 AAC 모바일 어플리케이션 또한 발전하고 있다. 그 중 GeoAAC의 사용 과정은 [그림 3]과 같다. AAC 사용자가 특정 장소와 상황의 의사소통에서 주로 활용하는 어휘를 중심으로 AAC 그림 상징의 보드가 구성되어 있다. AAC 사용자는 보드에 제시된 그림 상징 목록을 보고 자신이 표현하고자 하는 어휘와 일치하는 그림 상징을 선택하여 메시지를 구성한다. 예를 들어, 카페에서 '안녕하세요, 모과차 한 잔 주세요.'를 표

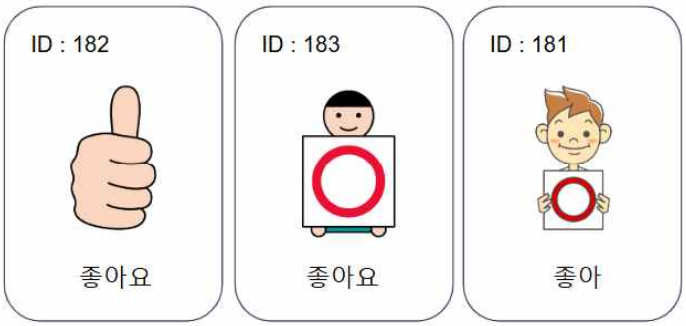
현하기 위해 ‘안녕하세요’, ‘주세요’, ‘모과차’ 상징을 선택할 수 있다. 여기서 구성한 상징 목록을 상징 시퀀스라고 한다. 모바일 앱은 사용자가 선택한 상징 시퀀스 내의 상징 어휘를 순차적으로 음성합성하여 음성으로 출력한다.



[그림 3] GeoAAC 앱의 사용 과정

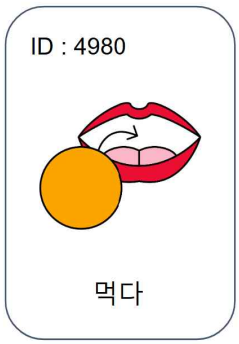
AAC 상징 시퀀스는 1개 이상의 상징으로 구성되며, 이를 통해 다양한 장소와 상황에서 자신의 의사를 표현할 수 있다. AAC 상징 체계는 구어 체계와 구분되는 몇 가지 특징을 지닌다.

AAC 상징과 상징 어휘는 N:M 관계를 갖는다. 먼저, 하나의 어휘에 대해 여러 개의 그림 상징이 존재할 수 있다. [그림 4]는 ‘좋아요’, ‘좋아’라는 선호의 의사를 나타내는 상징이다. 동일한 어휘와 의미에 대해 여러 개의 상징이 존재한다.



[그림 4] ‘좋아요’를 나타내는 그림 상징

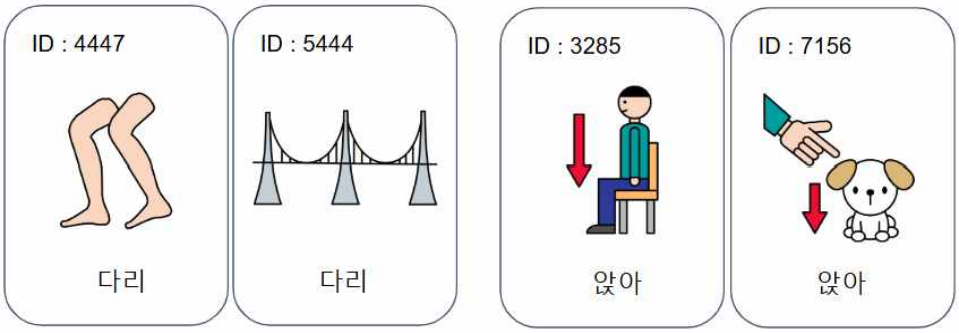
또한, 하나의 그림 상징은 기존의 어휘에서 다양하게 활용될 수 있다. [그림 5]는 ‘먹다’의 어휘를 나타내는 상징이다. 이 상징을 통해 ‘먹어’, ‘먹자’, ‘먹을래’ 등 권유형과 의문형의 표현으로 활용할 수 있으며, ‘먹어요’, ‘드세요’, ‘먹을게요’로 높임법 변환이 나타날 수 있고, ‘먹었어요’, ‘먹을 거에요’와 같이 과거형과 미래형으로 시제 변환이 나타날 수 있다.



[그림 5] ‘먹다’를 나타내는 그림 상징

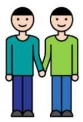
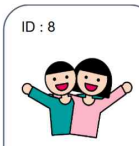
AAC 그림 상징은 [그림 6]과 같이 동일한 어휘더라도 상이한 의미를 지닐 수 있다. ‘다리’는 동음이의어로, 각각 신체의 일부인 다리와 건축 구조물인 다리를 나타낸다. ‘앉아’와 같이 지시, 권유하는 형태의 동일한 표현이더라도, 표

현의 대상이 각각 사람과 강아지로 다른 의미를 나타낸다. 이처럼 AAC 상징의 그림은 경우에 따라 의사소통의 배경과 대상을 포함하고 있어, 같은 어휘에 대해 다른 성격을 띌 수 있다.



[그림 6] 동일한 어휘로 다른 의미를 나타내는 그림 상징

AAC 사용자는 이러한 AAC 상징 그림의 특징에 기반하여 자기 의사를 표현하기 위한 상징 시퀀스를 구성한다. [그림 7]은 하나의 문장을 표현하기 위해 사용될 수 있는 AAC 상징 단위의 어휘와 상징 그림을 나타낸 것이다. 예를 들어, '내일 놀이터에서 같이 놀자'라는 문장에 대해 '내일', '놀이터', '같이', '놀자' 단위의 어휘를 갖는 상징으로 구성할 수 있으며, 구 형태의 '같이 놀자'라는 상징을 사용할 수 있다. 또한, AAC 상징 시퀀스는 AAC 사용자의 선택 과정에 따라 구어적 어순을 따르지 않을 수 있다. 예를 들어 '내일', '놀이터', '같이 놀자'의 상징 시퀀스를 '놀이터', '내일', '같이 놀자'의 순서로 선택할 수 있으며, 이는 원 문장의 의미를 그대로 나타내고 있다.

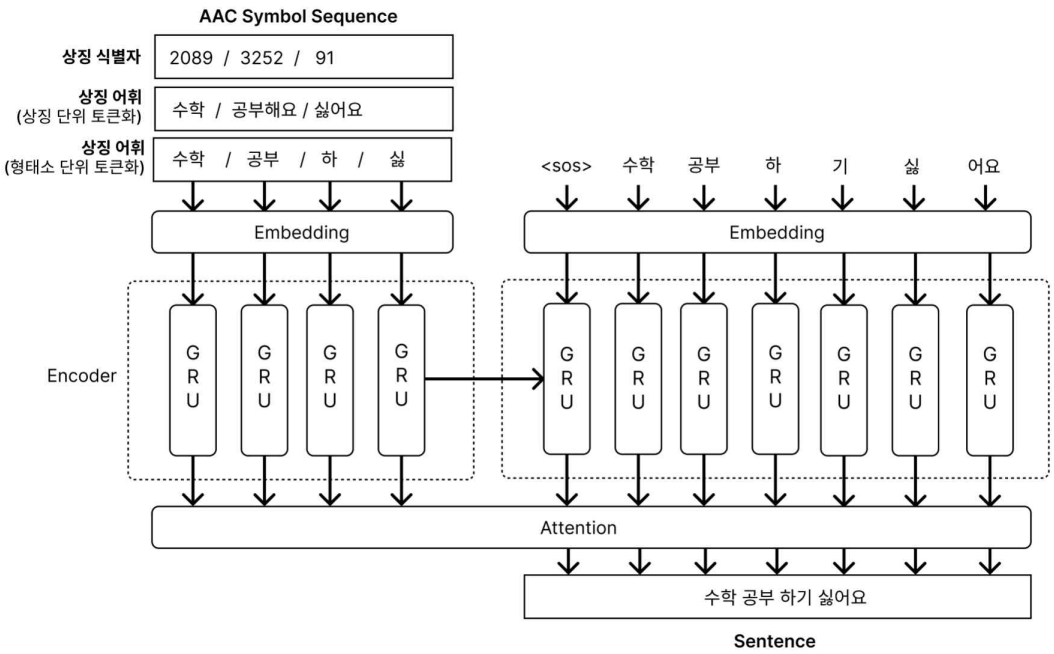
<table>
<tr><td>문장</td><td colspan="4">내일 놀이터에서 같이 놀자.</td></tr>
<tr><td>어휘</td><td>내일</td><td>놀이터</td><td>같이</td><td>놀자</td></tr>
<tr><td>AAC 상징</td><td>ID : 4443
 
ID : 3563
 
ID : 4444
  </td><td>ID : 5548
 
ID : 5549
 
ID : 5550
  </td><td>ID : 5709
  </td><td>ID : 4285
 
ID : 4286
 
ID : 8
  </td></tr>
</table>

[그림 7] '내일 놀이터에서 같이 놀자'를 나타내는 AAC 상징 시퀀스

AAC 앱을 통해 상징 시퀀스를 구성하고 음성합성으로 음성을 산출하는 경우, 상징 어휘를 순차적으로 출력하기 때문에 표현이 매끄럽지 않은 경향이 있다. 또한, 비대면 의사소통 상황에서 비장애인은 AAC 상징에 익숙하지 않기 때문에 AAC 상징 시퀀스로 상대와 메시지를 명확히 주고받는 데에 어려움이 있다. 기존 AAC 앱과 비대면 의사소통 서비스에서 AAC 상징 시퀀스를 한국어 문장으로 변환해 주는 기능이 적용된다면 AAC 사용자인 언어 장애인과 AAC에 익숙하지 않은 비장애인 사이의 의사소통에 큰 도움을 줄 수 있을 것이다.

2. 딥러닝 기반 보완대체의사소통 그림 상징 시퀀스의 한국어 문장 생성

보완대체의사소통(AAC) 그림 상징 시퀀스를 한국어 문장으로 변환하는 딥러닝 기반 연구는 [6, 7]이 있다. [그림 8]에서 볼 수 있듯이, 두 연구에서는 상징 시퀀스 정보로써 상징 식별자 또는 상징 어휘를 사용한다. AAC 상징 체계와 구어 체계 간 변환 작업은 기계 번역과 유사하다. 모델은 Sequence-to-Sequence의 인코더-디코더 구조로 설계하였으며, 인코더와 디코더의 순환 신경망으로는 GRU(Gated Recurrent Unit)[15]를 사용했다.



[그림 8] [6][7]의 상징 어휘, 식별자 기반 상징 시퀀스의 한국어 문장 생성 모델

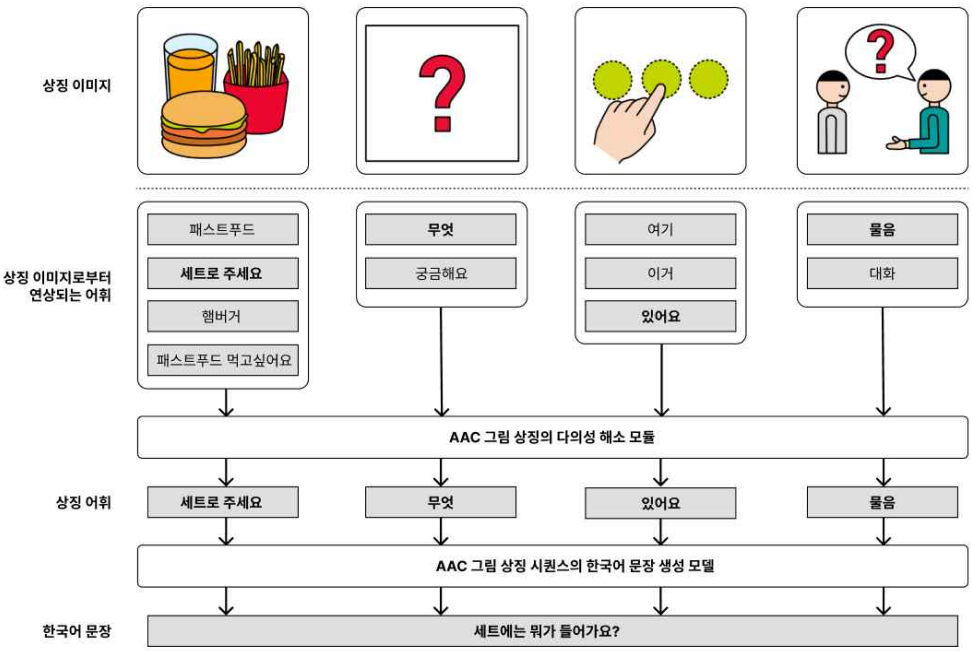
두 연구는 공통적으로 AAC 상징의 토큰화 및 임베딩 기법의 실험을 통해 모델의 번역 성능을 높이고자 하였다. [표 1]은 [6, 7]의 연구에서 실험한 AAC 상징의 정보, 토큰화 및 임베딩 기법을 보여준다. 2021년에 연구한 [6]에서는 AAC 상징의 정보로 상징의 식별자와 상징 어휘를 모두 사용하였다. 또한, 어휘 정보를 사용하는 경우 토큰화의 단위를 상징, 형태소로 구분하였다. 상징 단위로 토큰화하는 경우 Keras 임베딩[20]을 통해 밀집 벡터를 추출하고, 형태소 단위로 토큰화하는 경우 사전 학습된 FastText 모델[21]을 통해 단어의 밀집 벡터를 추출하였다. [6]의 실험 결과, AAC 상징 정보로 상징의 어휘를 사용하고 형태소 단위로 토큰화하여 FastText로 임베딩한 경우에 가장 좋은 성능을 보였다.

2023년에 연구한 [7]에서는 상징의 어휘 정보를 사용하였으며, BERT[22], BART[23], GPT-2[24] 등 대규모 언어 모델을 통해 추출된 텍스트 임베딩을 사용함으로써 그 성능을 높이고자 하였다. 해당 연구에서는 BERT의 임베딩을 사용한 모델이 58.26의 BLEU 점수로 가장 우수한 번역 성능을 보였다.

[표 1] 기존 연구[6][7]의 AAC 상징 정보, 토큰화 및 임베딩 방법

연구년도	AAC 상징	토큰화	임베딩
2021	식별자	상징 단위	Keras
	어휘	상징 단위	Keras
		형태소 단위	FastText
2023		LLM	LLM

또한, [7]의 연구는 상징의 이미지가 문장의 맥락에서 다양한 어휘를 나타낼 수 있다는 점을 고려해 다의성 해소 모듈을 적용하였다. [그림 9]는 [7] 연구에서 제안한 모델의 흐름을 간략화한 것이다. [그림 9]에서 볼 수 있듯이, 상징의 이미지를 보고 연상할 수 있는 어휘는 다양하다. AAC 그림 상징의 다의성 해소 모듈은 상징 이미지로부터 연상되는 어휘들을 모두 입력으로 한다. 상징 시퀀스의 어휘 간 유사도를 측정하고 각 상징으로부터 가장 적합한 어휘를 출력한다. 해당 모듈은 0.354로 정확도가 높진 않으나, 상징의 이미지로부터 사람이 연상하는 다양한 어휘를 반영하고자 했다는 점에서 의의가 있다.



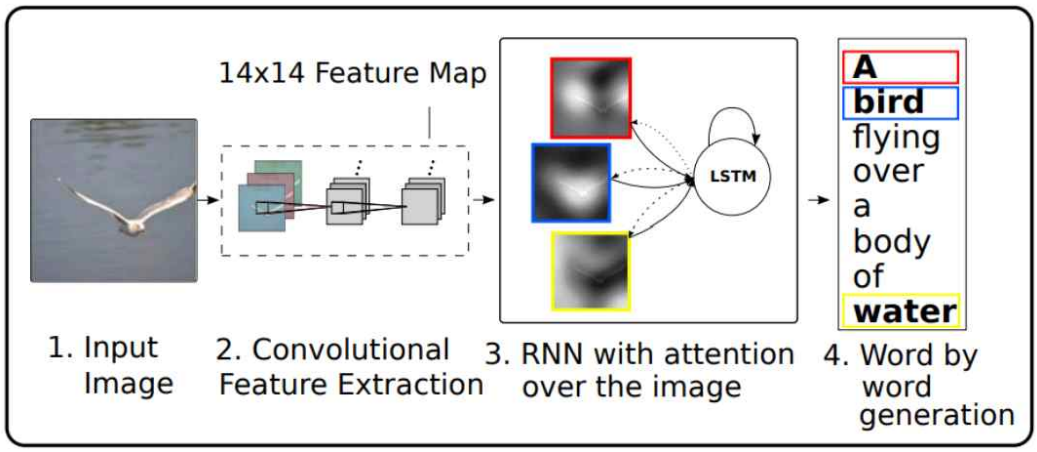
[그림 9] AAC 그림 상징의 다의성 해소 모듈을 적용한 한국어 문장 생성 모델[7]

[7]의 연구에서 볼 수 있듯이, AAC 상징 시퀀스 내에서 AAC 상징의 이미지 자체가 갖는 특징은 자연스러운 한국어 문장을 생성하는 데 있어 중요한 요소이다. AAC 사용자는 시각적 정보에 기반하여 AAC 그림 상징을 선택하

기 때문에, AAC 상징에 대응되는 어휘 외의 표현을 의도할 수 있다. 따라서 본 연구에서는 AAC 상징의 시각적 정보인 그림에 기반하여 한국어 문장을 생성하고자 한다.

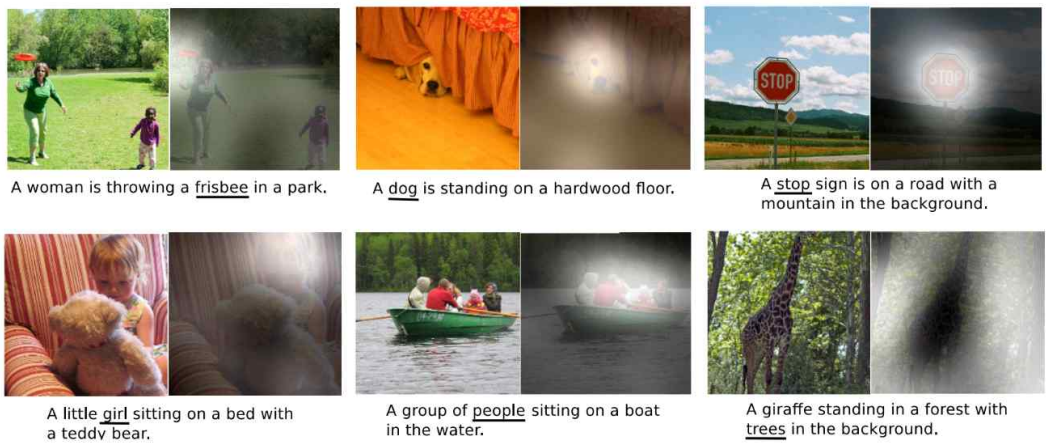
3. 이미지 캡셔닝

본 연구는 이미지 정보로부터 한국어 문장을 생성한다는 점에서 이미지 캡셔닝(Image Captioning) 작업과 연관성이 크다. 이미지 캡셔닝 작업의 대표적인 연구로는 [25]가 있다. [25]는 이미지를 텍스트로 변환하는 문제를 기계 번역의 Sequence-to-Sequence 모델 구조를 응용하여 해결하고자 했으며, [그림 10]은 [25]에서 제안한 모델의 구조이다. Sequence-to-Sequence 모델은 인코더-디코더 구조로 구성된다. 해당 연구에서는 인코더로 합성곱 신경망을 사용하여 공간적 특징 맵(feature map)을 추출한다. 디코더는 LSTM(Long Short-Term Memory)[14]을 활용하여, 이전 시점의 단어와 컨텍스트 정보를 기반으로 다음 정보를 예측한다. 이때 Attention Mechanism을 적용하여 각 단어를 생성하는 시점에 이미지 특징 맵의 각 위치에 가중치를 할당함으로써, 각 단어를 생성할 때 이미지 특정 부분에 집중할 수 있다.



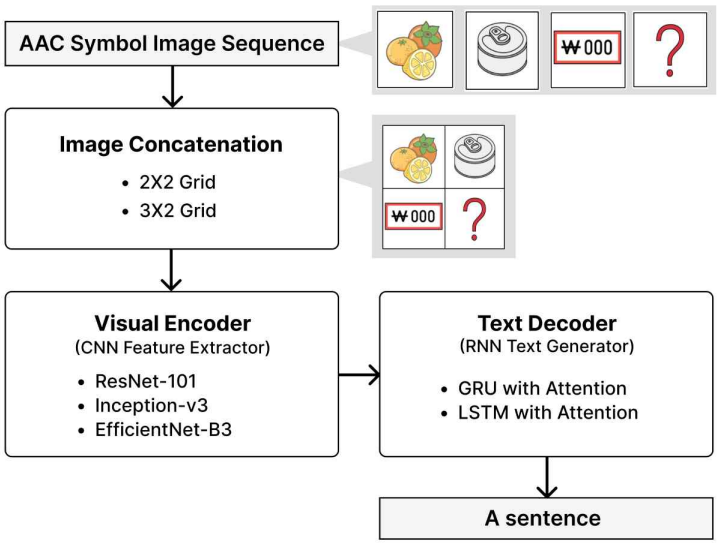
[그림 10] 이미지 캡셔닝 모델 구조[25]

Attention 기법으로는 Soft Attention과 Hard Attention 두가지를 제안하였다. Soft Attention은 이미지의 모든 부분에 대해 가중합을 계산하여 역전파를 통해 학습시키는 반면, Hard Attention은 이미지의 특정 부분을 확률적으로 선택하여 강화학습 알고리즘을 통해 학습된다. [그림 11]은 이미지 캡셔닝 모델에서 각 단어 생성 시점에 집중한 영역을 시각화한 그림[25]이다.



[그림 11] 단어 생성 시점의 집중 영역 시각화 결과[25]

Ⅲ. Visual Encoder를 적용한 딥러닝 기반 AAC 그림 상징 시퀀스의 한국어 문장 생성



[그림 12] Visual Encoder를 적용한 AAC 그림 상징 시퀀스의 한국어 문장 생성 모델

본 연구에서는 AAC 상징 시퀀스의 한국어 문장 변환을 위해, AAC 상징의 이미지 정보를 활용하고자 한다. 따라서 기존 연구의 모델인 Sequence-to-Sequence의 Text Encoder 부분을 Visual Encoder로 변경하였다. 본 연구의 모델 실험 과정은 [그림 12]와 같다. AAC 상징 시퀀스는 여러 개의 AAC 상징으로 구성되어 있으므로, 각 이미지를 병합하여 하나의 이미지로 구성한다. 이를 통해 각 상징이 상징 시퀀스 내에서 가질 수 있는 의미를 효과적으로 반영할 수 있다. 이미지에서 하나의 상징 그림이 차지하는 영역을

고려하여 2×2, 3×2의 그리드 형태로 병합한 경우를 나누어 실험한다.

이미지를 해석하기 위한 Visual Encoder로써, CNN(Convolutional Neural Network)의 마지막 Convolution Layer를 통해 추출된 특징 벡터를 사용한다. CNN 모델은 ResNet-101[11], Inception-v3[12], EfficientNet-B3[13]를 선정하였으며, 각 모델을 사용했을 때 결과를 비교한다. 한국어 문장을 해석하기 위해 Attention Mechanism 기반의 GRU, LSTM 두 개의 모델을 선정하였다.

2×2, 3×2 이미지 병합 크기에 따른 실험을 위해 상징 시퀀스의 최대 길이가 4인 데이터셋과 6인 데이터셋으로 나누었다. 이 2가지 데이터셋에 대해 CNN 모델 3가지, RNN 모델 2가지로 총 6가지 조합의 모델을 실험하고 성능을 평가한다. 본 장에서는 데이터셋의 구축 과정과 모델의 설계 과정을 구체적으로 기술하였다.

1. 데이터셋

본 연구의 학습 데이터셋은 기존 연구[6, 8, 9]에서 구축한 AAC 상징 어휘, 아이디 시퀀스-한국어 문장 쌍의 데이터를 활용하였다. 한국어 문장 데이터는 언어 장애인의 의사소통 능력과 다양한 장소와 상황에서 표현하고자 하는 문장을 고려하여 구성되었다. AI Hub[26]의 한국어 대화 데이터, AI Hub 공개데이터인 KETI의 일상, 오피스 대화 데이터, [27-30]의 논문에서 발췌한 AAC 사용자의 주요 어휘 및 문장, 동화책의 문장을 활용하였으며, [표 2]는 해당 데이터셋을 구성하는 한국어 문장의 출처와 수를 나타낸다.

[표 2] 기존 연구 데이터셋의 한국어 문장 출처 및 데이터 수

출처	문장 데이터 수
AI Hub 한국어 대화(Dialog)	7,638
AI Hub 일상 오피스 대화	2,600
AAC 논문[27-30]	1,665
동화책	3,953

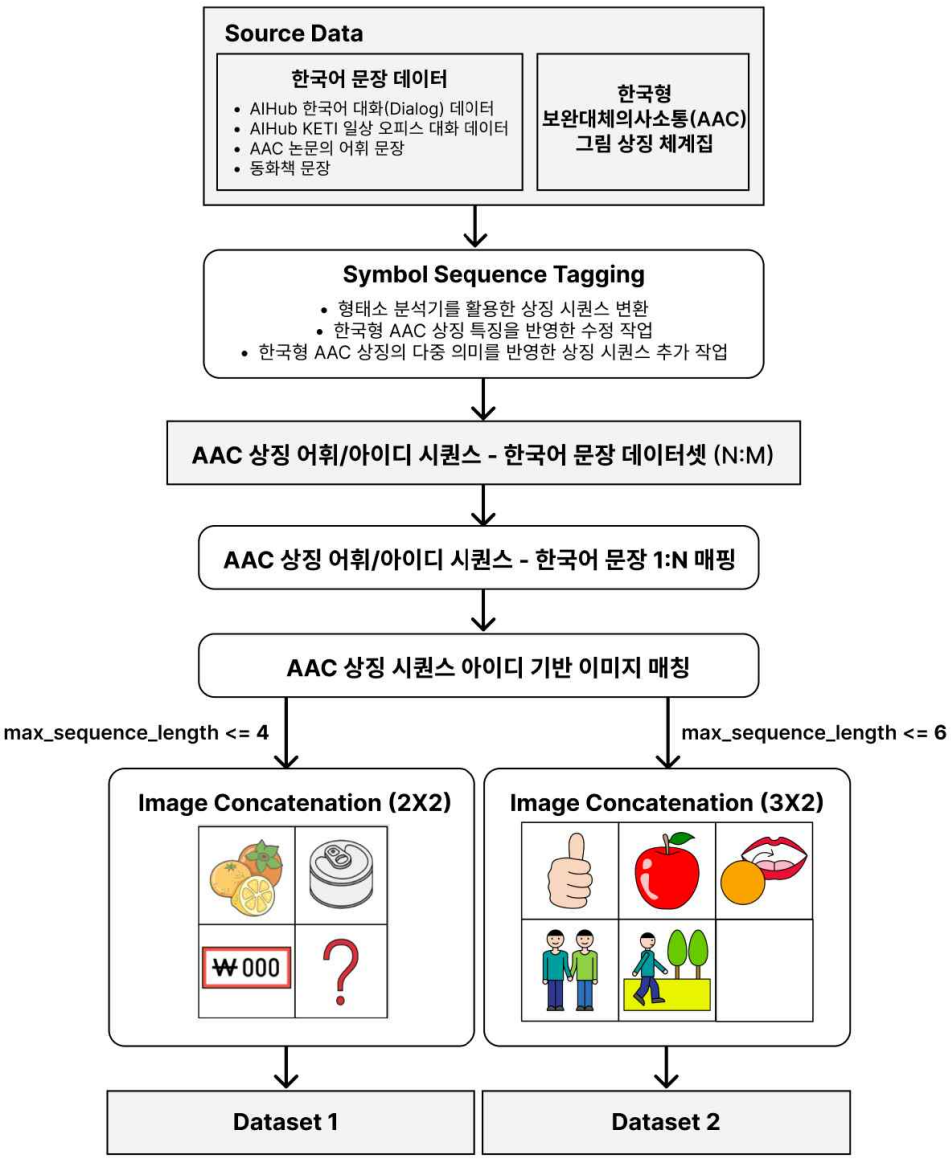
기존 연구의 데이터셋을 구성하는 상징 시퀀스는 한국형 AAC 상징 체계집[2]의 상징 어휘와 아이디를 기반으로 한다. 한국어 문장에 대응되는 상징 시퀀스를 구성하기 위해 3가지의 상징 시퀀스 태깅(Symbol Sequence Tagging) 과정을 거쳤다.

- 형태소 분석기를 활용한 상징 시퀀스 변환 : 한국어 문장을 형태소 분석하여 일치하는 어휘의 상징과 매핑한다.

- 한국형 AAC 상징 특징을 반영한 수정 작업 : 형태소 분석 결과와 상징의 어휘가 일치하더라도, ‘눈’, ‘다리’, ‘밤’과 같은 동음이의어가 있을 수 있으며, 상징의 그림이 포함하는 배경과 대상으로 인해 한국어 문장의 의도와 일치하지 않을 수 있다. AAC 상징 그림의 특징을 반영하여 AAC 상징 시퀀스를 수정한다.

- AAC 상징의 다중 의미를 반영한 상징 시퀀스 추가 작업 : 하나의 문장에 대해 다양한 상징 시퀀스로 구성될 수 있다는 점을 고려해 상징 시퀀스의 추가 작업을 거친다. 예를 들어, ‘놀이터에 가서 놀자’라는 한국어 문장에 대해 ‘놀이터’, ‘가다’, ‘놀다’로 3개의 상징 시퀀스를 사용할 수 있으며, ‘놀이터’, ‘놀다’로 2개의 상징 시퀀스를 사용할 수 있다.

기존 데이터셋의 구축 과정과 본 연구를 위해 추가 작업한 과정은 [그림 13]과 같다.



[그림 13] AAC 상징 이미지 시퀀스 - 한국어 문장 데이터셋 구축 과정

상징 시퀀스 태깅 과정을 거친 데이터셋을 다시 다음의 3가지 작업을 거쳤다.

- AAC 상징 어휘/아이디 시퀀스-한국어 문장 1:N 매핑

기존 연구의 데이터셋은 상징 시퀀스와 문장이 N:M 관계를 갖는다. 해당 데이터를 그대로 사용하여 학습, 평가 데이터로 분리하는 경우 동일한 상징 시퀀스에 대해 학습과 예측을 모두 수행하게 되므로, 정확한 평가가 이루어지기 어렵다. 본 연구에서는 정확한 평가를 위해 상징 시퀀스와 문장을 1:N의 관계로 매핑하였다.

- AAC 상징 시퀀스 아이디 기반 이미지 매칭

상징 아이디를 상징 이미지로 변환하였으며, 이 과정에서 일부 상징 시퀀스에 대해 상징을 제거 또는 추가하였다. 먼저, 기존 데이터셋은 AAC 상징 체계가 단위를 나타내는 상징을 포함하고 있지 않아, [표 3]과 같이 어휘 형태의 AAC를 자체적으로 제작하여 시퀀스에 추가하였다. 예를 들어, '3명 왔습니다.'를 표현하기 위해 '3', '-명', '왔어요' 상징으로 구성하여야 하지만, 인원수의 단위를 나타내는 '-명' 상징에 대한 그림이 존재하지 않았다.

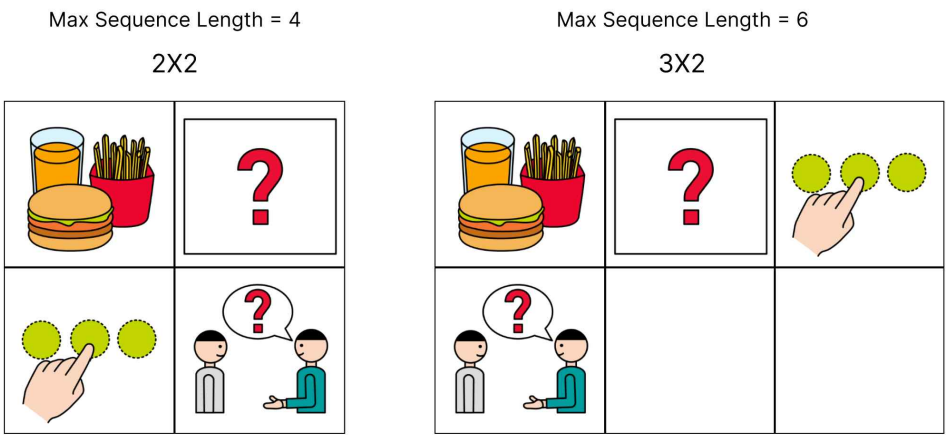
[표 3] 단위 의존 명사를 나타내는 상징 어휘

-분	몇장	-년
-명	-벌	-주
나이	종류	며칠
-학년	몇인분	몇시
몇살	-근	몇분
번호	-퍼센트	몇초

이렇게 상징 시퀀스 내에서 단위 의존 명사를 나타내는 상징이 존재하는 경우, 해당 상징은 시퀀스에서 제외하였다. 추가적으로, 의문문과 평서문에 대한 상징 시퀀스를 구분하기 위해, 한국어 문장이 의문문인 경우 물음표 형태로 표현된 '물음'의 그림 상징을 상징 시퀀스에 포함하였다.

- AAC 상징 시퀀스 최대 길이에 따른 데이터셋 분리 및 상징 이미지 병합

마지막으로, 시퀀스 최대 길이가 4인 경우와 6인 경우로 나누어 이미지를 병합하고 두 가지의 데이터셋을 구축하였다. 기존 데이터셋의 상징 시퀀스는 최소 1개부터 20개까지의 상징으로 구성되어있다. 실제 AAC 사용자는 개인의 표현 능력에 따라 1개~5개 내외의 상징으로 시퀀스를 구성하며, 상징 이미지들을 하나의 이미지로 병합하였을 때 하나의 상징이 차지하는 영역이 작은 경우 이미지 특징 추출이 어려울 수 있다. 따라서, 최대 시퀀스 길이가 4인 경우와 6인 경우로 나누어 각각 2×2, 3×2의 그리드 형태로 병합하였으며 [그림 14]는 그 예시이다. 이때 각 상징의 경계를 나타내기 위해 검은색의 실선(bounding box)을 추가하였다.



[그림 14] 최대 상징 시퀀스 길이에 따른 이미지 병합 결과

최종적으로 구축한 데이터셋(Dataset 1, 2)는 [표 4]와 같다. 최대 시퀀스 길이가 4인 데이터셋(Dataset 1)은 1개부터 4개까지의 상징으로 구성된 상징 시퀀스와 한국어 문장을 포함하며, 상징 시퀀스 기준 총 66,931개이다. 최대 시퀀스 길이가 6인 데이터셋(Datset 2)은 1개부터 6개까지의 상징으로 구성된 상징 시퀀스와 한국어 문장 데이터이며, 총 118,350개이다.

[표 4] 최종 데이터셋 개요

	상징 시퀀스 최대 길이	**데이터 수**
Dataset 1	4	66,931
Dataset 2	6	118,350

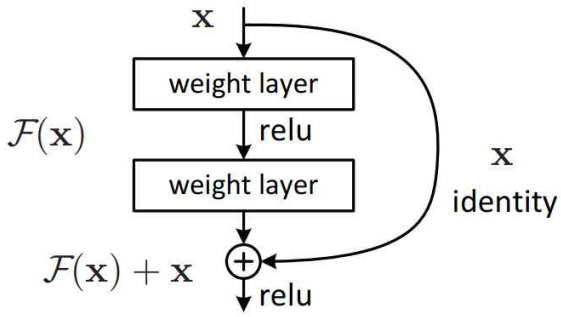
2. 이미지 임베딩 추출을 위한 Visual Encoder

AAC 상징 시퀀스의 이미지의 특징 맵(feature map)을 추출하기 위해 ImageNet[31]으로 사전 학습된 세가지의 합성곱 신경망 모델을 Visual Encoder로 활용하였다. ResNet[11], Inception[12, 32], EfficientNet[13] 모델은 이미지 분류 작업을 통해 학습한 모델로, 마지막 Convolution Layer로부터 유의미한 이미지 특징 벡터를 추출할 수 있다. 본 장에서는 이미지 특징 추출기로 활용한 ResNet, Inception, EfficientNet의 개념과 구조를 설명한다.

1) ResNet(Deep Residual Network)

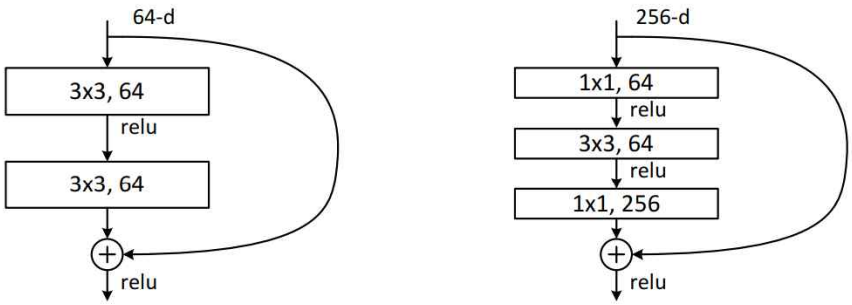
합성곱 신경망 모델은 네트워크 깊이가 깊어질수록 이미지의 복잡한 패턴과 특징을 더 세부적으로 학습할 수 있으나, 모델의 깊이가 증가함에 따라 생기는 기울기 소실(Gradient Vanishing) 문제가 나타난다.

ResNet[11]은 이러한 기울기 소실 문제를 해결하고 신경망의 깊이를 더 늘리는 것에 초점을 맞추어 [그림 15]와 같은 Residual Block을 제안하였다. Residual Block은 이전 층의 입력 x를 현재 층의 출력에 더하는 잔차 연결(Skip Connection)을 통해 학습 과정에서 잔차를 고려하게 된다. 잔차 연결의 경로는 기존 경로와 달리 2개의 기울기 값만을 곱하여 학습이 진행되기 때문에, 네트워크 층이 깊어지더라도 기울기를 소실하지 않고 효과적인 학습이 가능하다.



[그림 15] ResNet의 Residual Block[11]

또한, 깊은 네트워크 구성으로 인해 연산량이 증가하는 것을 해결하기 위해 병목 블록(bottleneck block)을 제안하였으며, [그림 16]과 같은 구조를 보인다. 기존 합성곱 연산을 하기 전에 1×1의 합성곱을 통해 특징 맵의 채널 수를 줄이고, 합성곱 연산 후 다시 1×1 합성곱 연산을 통해 특징 맵의 채널 수를 복원하는 개념으로, 병목 현상과 유사한 형태를 보인다.



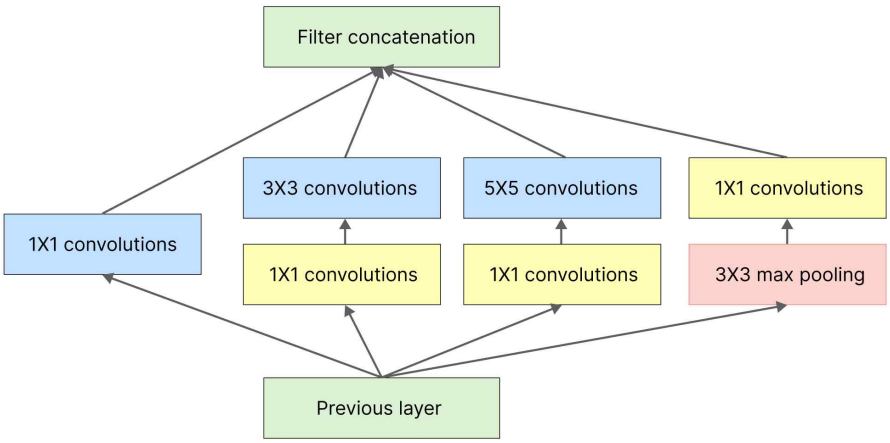
[그림 16] ResNet의 (좌)Residual Block과 (우)Bottleneck Block[11]

모델의 깊이인 층의 개수에 따라 ResNet-18, ResNet-34, ResNet-50 등 다양한 모델이 있으며, 본 연구에서는 이미지 캡셔닝 작업에 주로 사용되는 ResNet-101 모델을 활용한다.

2) Inception(GoogLeNet)

Inception[12, 32]은 하나의 계층 내에서 다양한 크기의 합성곱 필터를 병렬로 연결하여 다양한 크기의 특징을 동시에 학습하는 모델이다. Inception 또한 합성곱 신경망 모델이 깊어질수록 학습 데이터에 과적합(overfitting)되는 문제와 연산량이 증가하는 문제를 해결하고자 하였다.

[그림 17]은 Inception 모델의 핵심 모듈을 나타낸 그림이다. 1×1, 3×3, 5×5 크기의 합성곱 필터를 한 계층에서 병렬로 적용하여 다양한 크기를 특징을 학습할 수 있도록 하였다. 기존 모델은 계층마다 단일 필터를 사용하여 이미지 특징을 추출했기 때문에, 다양한 크기의 객체를 인식하기 어렵다. Inception 모듈을 적용하여 다양한 크기와 방향의 객체를 효율적으로 인식할 수 있는 특징을 추출할 수 있다.

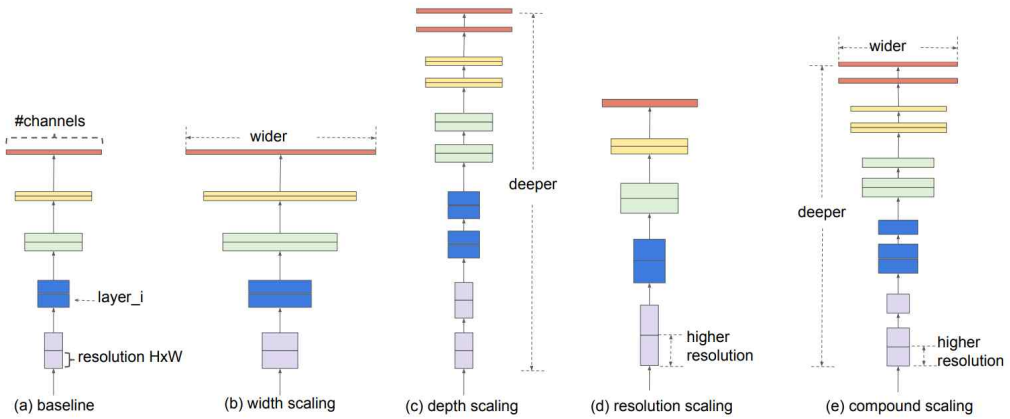


[그림 17] Inception 모듈[32]

층이 깊어질수록 학습 파라미터 수, 연산량이 기하급수적으로 늘어날 것을 고려하여, 연산량이 큰 3×3, 5×5 합성곱 연산 전과 3×3 풀링(pooling) 연산 후에 1×1 합성 곱 연산을 적용하여 채널을 축소하였다.

3) EfficientNet

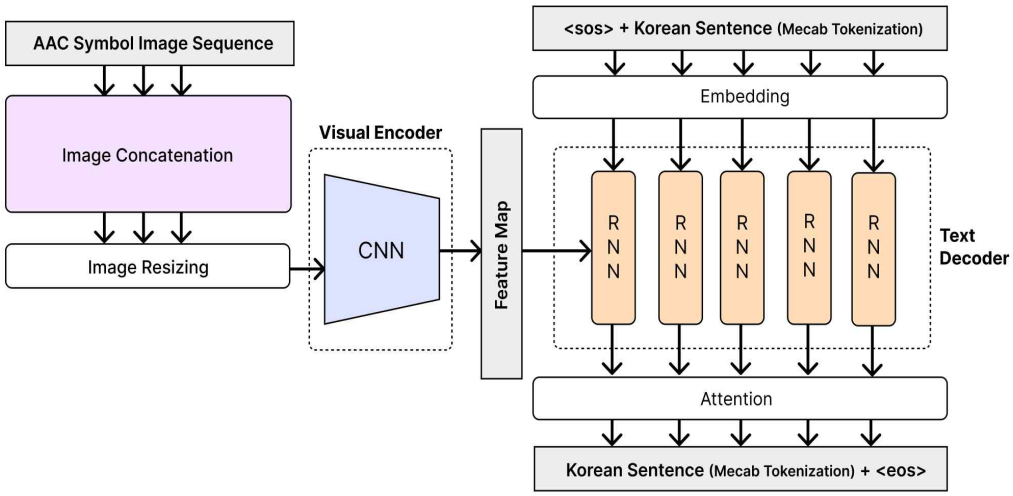
EfficientNet[13]은 기존 합성곱 신경망 모델이 네트워크의 특정 요소만 조정하는 방식으로 연구되어왔다는 점을 문제 삼아, 모델의 성능을 향상시키기 위해 종합적인 요소를 고려하는 Compound Scaling 방식을 제안하였다. [그림 18]은 [13]의 논문에서 모델을 확장할 때 고려하는 변수를 시각화한 것으로, 기존 연구는 모델의 성능 향상을 위해 네트워크의 (b)너비(width), (c)깊이(depth), (d)해상도(resolution)를 각각 고려하고 있으며, EfficientNet은 (e)와 같이 세가지 요소를 모두 고려한다.



[그림 18] 합성곱 신경망 모델의 확장 방식[13]

EfficientNet은 세가지 요소를 모두 고려하여 확장함으로써 성능과 효율성의 균형을 맞추었다. 입력 해상도와 깊이, 너비에 따라 B0 모델부터 B7 모델까지 있으며, 본 연구의 모델로는 데이터의 규모와 연산량을 고려하여 EfficientNet-B3를 채택하였다.

3. AAC 그림 상징 시퀀스의 한국어 문장 생성 모델 설계



[그림 19] AAC 그림 상징 시퀀스의 한국어 문장 생성 모델 설계

본 연구에서 설계한 모델은 [그림 19]와 같다. 최대 길이가 4인 AAC 상징 시퀀스와 최대 길이가 6인 시퀀스로 두 종류의 데이터셋이 있으며, 각각 상징 이미지를 2×2, 3×2의 그리드 형태로 병합(Image Concatenation)하는 과정을 거친다. 이미지는 가로(width) 299픽셀, 세로(height) 299픽셀, 3개의 채널(channel)로 크기를 조정(Image Resizing)한다. Visual Encoder로는 사전 학습된 CNN 모델인 ResNet-101, Inception-v3, EfficientNet-B3를 활용하고, 각 모델의 마지막 합성곱 층(Convolutional Layer)으로부터 이미지 특징 벡터(Feature Maps)를 추출한다. [표 5]는 Visual Encoder의 모델별 입력 이미지 크기, 출력 벡터 크기, 모델의 파라미터 수이다.

[표 5] Visual Encoder의 모델별 입출력 크기 및 파라미터 수

Model	Input Shape (H×W×C)	Output Shape (H×W×C)	Parameters (Millions)
ResNet-101	299×299×3	10×10×2048	44.5
Inception-v3		8×8×2048	27.2
EfficientNet-B3		9×9×1536	12

RNN인 LSTM, GRU를 통해 한국어 문장을 생성하고, Attention Mechanism을 적용하여 문장의 각 단어를 생성하는 시점에 중요한 이미지 영역에 가중치를 두도록 학습한다.

최종적으로 두 종류의 데이터셋에 대해 실험할 모델은 [표 6]과 같다. 합성곱 신경망 모델 3가지, 순환 신경망 모델 2가지의 조합으로 총 6개 모델을 실험한다.

[표 6] 실험 모델

NO.	Visual Encoder	Text Decoder
1	Inception-v3	GRU with Attention
2		LSTM with Attention
3	ResNet-101	GRU with Attention
4		LSTM with Attention
5	EfficientNet-B3	GRU with Attention
6		LSTM with Attention

IV. 모델 실험 및 평가

본 장에서는 실험을 진행한 환경, 모델의 학습 과정과 실험 결과, 성능 평가 결과를 설명한다.

1. 실험 환경

모델의 실험 환경은 [표 7]과 같다.

[표 7] 실험 환경

구분		버전
H/W	CPU	Intel(R) Core(TM) i9-13900KF
	RAM	64.0GB
	GPU	NVIDIA Geforce RTX 3060
S/W	OS	Windows 11
	Python	3.8
	Tensorflow	2.10.0
	CUDA	11.2
	cuDNN	8.1

2. 모델 실험

1) 학습 데이터

데이터셋은 상징 시퀀스 길이가 4인 경우와 6인 경우로 두 종류가 있다. 모델 학습과 평가를 위해 데이터셋을 [표 8]과 같이 9:1 비율로 나누었다. 상징 시퀀스 최대 길이가 4인 데이터셋은 학습 데이터 60,238개, 평가 데이터 6,693개이다. 이미지 내 하나의 상징 크기에 따른 모델의 성능을 평가하기 위해 상징 시퀀스의 최대 길이가 6인 경우에 대해서도 4인 경우의 데이터 수에 맞추어 샘플링(Sampling)하여 사용하였다. 학습 과정을 검증하기 위한 데이터는 학습 데이터의 10%를 추출하여 사용했다.

[표 8] 최종 데이터셋 개요

상징 시퀀스 최대 길이	전체 데이터 수	학습 데이터 수	평가 데이터 수
4	66,931	60,238	6,693
6	118,350	60,238	6,693

2) 모델 학습

앞서 설명한 2가지 데이터셋에 대해 6가지 모델로 실험하였다. 이미지 특징 추출기인 ResNet-101, Inception-v3, EfficientNet-B3는 ImageNet으로 사전 학습된 모델을 추가 학습 없이 그대로 사용하였다. 한국어 문장은 형태소 단위로 토큰화한 후 Keras Embedding[20]을 통해 추출되는 256차원의 단어 벡터를 사용한다. 형태소 분석기는 Python의 한국어 자연어 처리 패키지인 KoNLPY[33]의 Mecab을 사용하였다. 배치 크기는 64로 설정하였고, 최적화(Optimizer) 알고리즘은 Adam, 손실함수(Loss Function)는 Sparse Categorical Cross Entropy를 사용하였으며, 학습률은 0.001로 설정하였다. 반복 학습 횟수(epoch)는 50으로 설정하였으며, 과적합을 방지하기 위해 5번 연속으로 Loss 값이 개선되지 않는 경우 학습을 중단할 수 있도록 조기 종료(Early Stopping) 기법을 적용하였다. [표 9, 10]은 각각 최대 시퀀스 길이가 4, 6인 데이터셋에 대한 모델의 학습 결과이다.

[표 9] 최대 시퀀스 길이가 4인 데이터셋에 대한 모델의 학습 결과

Model	Train		Valid	
	Loss	Accuracy	Loss	Accuracy
Inception-v3+GRU	0.032	0.960	0.171	0.941
Inception-v3+LSTM	0.019	0.964	0.143	0.951
ResNet-101+GRU	0.028	0.969	0.144	0.947
ResNet-101+LSTM	0.022	0.964	0.116	0.944
EfficientNet-B3+GRU	0.027	0.962	0.164	0.944
EfficientNet-B3+LSTM	0.018	0.965	0.137	0.949

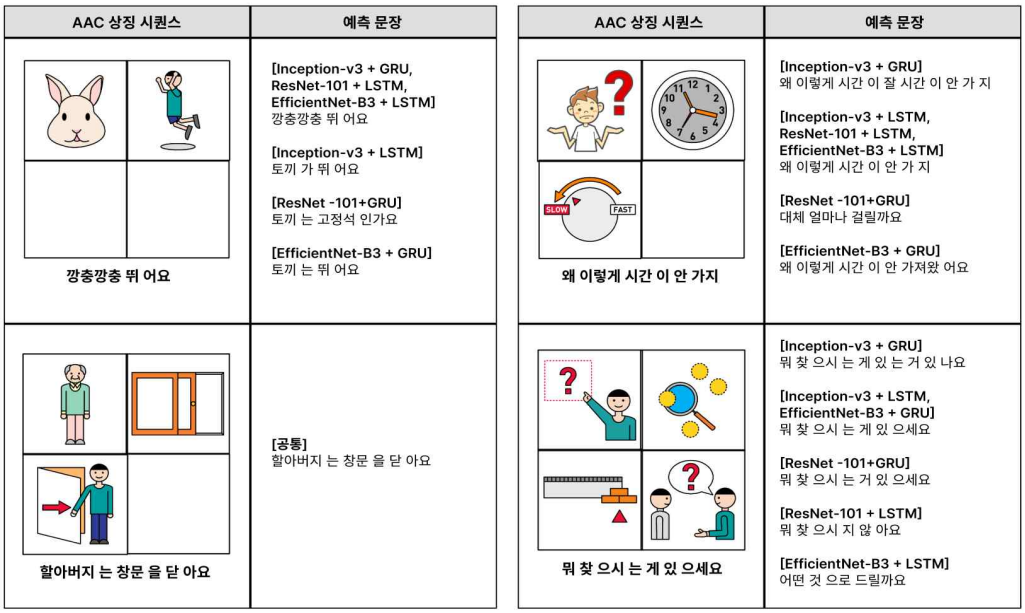
[표 10] 최대 시퀀스 길이가 6인 데이터셋에 대한 모델의 학습 결과

Model	Train		Valid	
	Loss	Accuracy	Loss	Accuracy
Inception-v3+GRU	0.028	0.953	0.166	0.936
Inception-v3+LSTM	0.020	0.957	0.133	0.944
ResNet-101+GRU	0.025	0.955	0.137	0.941
ResNet-101+LSTM	0.014	0.959	0.118	0.947
EfficientNet-B3+GRU	0.025	0.955	0.169	0.938
EfficientNet-B3+LSTM	0.014	0.958	0.136	0.945

3. 모델 평가

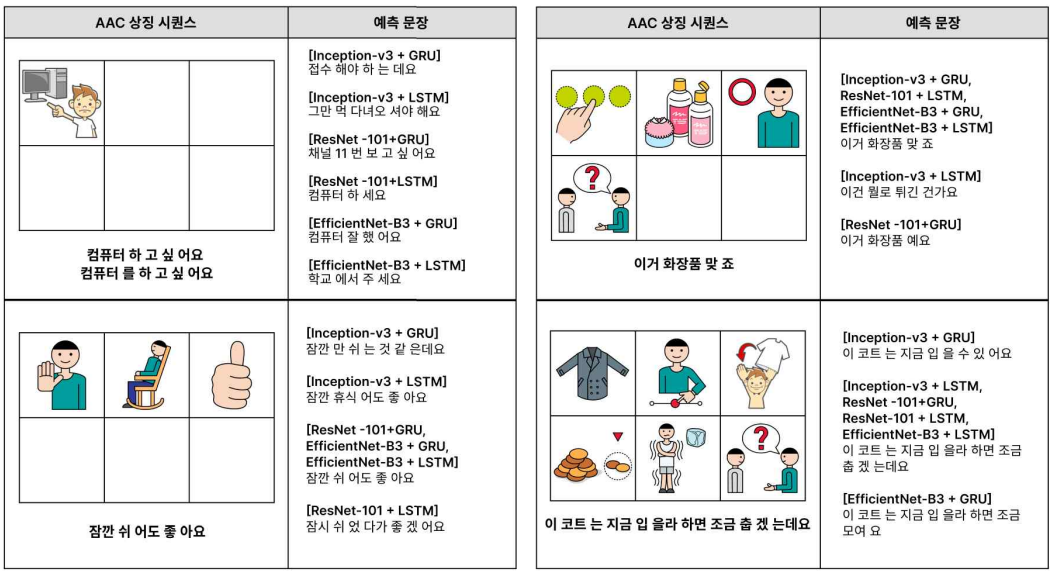
1) 문장 예측 결과

[그림 20]은 최대 길이가 4인 AAC 상징 시퀀스에 대한 한국어 문장이다. 첫 번째 이미지는 '깡충깡충 뛰어요.' 문장을 의미한다. 각 모델은 '깡충깡충'을 나타내고자 한 '토끼' 그림 상징에 대해 '깡충깡충' 또는 '토끼'로 예측한 결과를 확인할 수 있다. '뭐 찾으시는 게 있으세요?'라는 문장의 상징 이미지에 대해 해당 문장과 유사한 의미를 내포하는 문장을 예측하였다. 또한 '물음' 상징의 의미를 반영하여 한국어 문장이 의문문으로 생성된 것을 볼 수 있다.



[그림 20] 최대 시퀀스 길이가 4인 데이터셋에 대한 모델의 한국어 문장 예측 결과

최대 길이가 6인 AAC 상징 시퀀스의 문장 예측 결과는 [그림 21]과 같다. ‘컴퓨터 하고 싶어요.’에 대한 상징 시퀀스에 대해서는 매우 적절하지 않은 문장이 출력되었으며, 짧은 길이의 시퀀스일수록 이러한 결과를 보였다. 3×2의 그리드 병합으로 인해 하나의 상징이 차지하는 이미지 영역이 과도하게 작아지고, 이미지 공간이 과도하게 비어있기 때문에 짧은 길이의 시퀀스 이미지에 대한 정상적인 예측이 비교적 어려웠다. 반면, ResNet-101, EfficientNet-B3를 사용한 모델에서 ‘컴퓨터’ 객체와 관련된 단어를 출력한 결과를 통해 Inception-v3보다 ResNet-101, EfficientNet-B3를 사용한 모델이 작은 이미지 영역을 해석하기 더 적합하다는 점을 확인할 수 있다. 길이가 3 이상인 상징 시퀀스에 대해서는 모든 모델에서 전반적으로 우수한 번역 성능을 보였다.



[그림 21] 최대 시퀀스 길이가 6인 데이터셋에 대한 모델의 한국어 문장 예측 결과

[그림 22]는 모델이 각 단어를 예측하는 시점에 집중(Attention)한 이미지 영역을 시각화한 것이다. 최대 길이가 4인 시퀀스의 이미지에 대해 각각 ‘컬

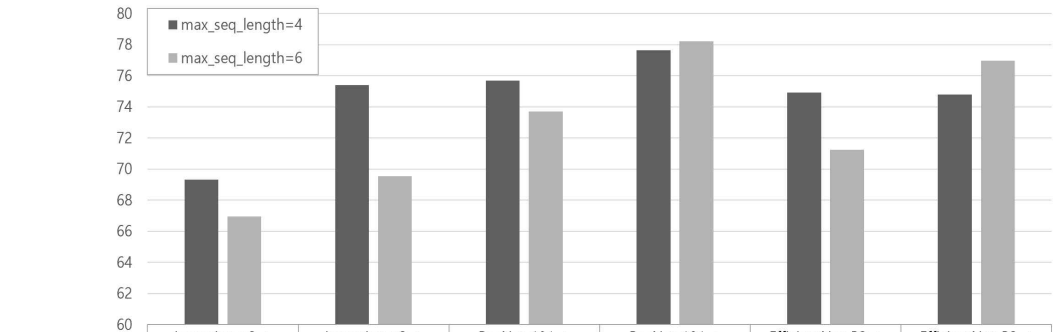
러', '겨울', '손', '자동차' 상징 영역을 집중하였으며, 최대 길이가 6인 시퀀스의 이미지에 대해 각각 '눈물', '물', '아이스', '많이'를 나타내는 상징의 이미지 영역에 집중한 결과를 확인할 수 있다. 그러나 단어와 연관된 이미지 영역 외 공백의 이미지 영역 또한 집중된 결과를 보였다. 최대 시퀀스 길이에 비해 짧게 구성된 시퀀스는 이미지에 공백이 다량 포함되어 있다. 이처럼 짧은 길이의 시퀀스 이미지에 대해 공백을 다량 학습했기 때문에, 최대 시퀀스 길이에 비해 시퀀스 길이가 짧을수록 예측 결과가 부정확한 것으로 보이며 다른 이미지 병합 방식을 시도할 필요성이 있다는 점을 시사한다.

[그림 22] 각 단어 생성 시점의 이미지 집중 영역

2) BLEU 평가 결과

본 연구에서는 모델의 번역 성능을 평가하기 위해 BLEU 점수를 사용했다. BLEU는 기계 번역과 인간 번역의 유사성을 n-gram에 기반하여 특정한 점수이다. 전체 n-gram을 고려하는 BLEU 점수를 기본 측정 지표로 사용하였으며, 단어 수준의 유사성, 문장 흐름의 유사성을 각각 평가하기 위해 BLEU-1, BLEU-2, BLEU-3, BLEU-4를 함께 측정하였다. BLEU의 점수 해석은 [34]를 참고하였다.

[그림 23]은 예측 결과에 대해 BLEU 점수를 평가한 결과이다. 최대 시퀀스 길이(max_seq_length)가 4일 때, Inception-v3와 GRU 구조를 활용한 모델이 69.313으로 가장 낮은 점수를 나타냈으며, ResNet-101과 LSTM 구조의 모델은 77.631로 가장 높은 번역 성능을 보였다. 이러한 결과는 최대 시퀀스 길이가 6인 데이터셋에 대해서도 동일하게 나타난다.



	Inception-v3 + GRU	Inception-v3 + LSTM	ResNet-101 + GRU	ResNet-101 + LSTM	EfficientNet-B3 + GRU	EfficientNet-B3 + LSTM
max_seq_length=4	69.313	75.402	75.691	77.631	74.92	74.806
max_seq_length=6	66.953	69.537	73.696	78.232	71.233	76.97

[그림 23] 모델별 BLEU 평가 결과

Inception-v3가 Visual Encoder인 모델에 대해서는 Text Decoder에 GRU 구조를 사용한 모델보다 LSTM을 사용했을 때 더 좋은 성능을 보였다. 복잡한 이미지 데이터를 장기적으로 처리하기 위해서 LSTM의 구조를 사용하는 것이 적합함을 시사한다. 또한 최대 시퀀스 길이가 6인 데이터셋에 대해서 더 낮은 성능을 보였는데, 이는 Inception-v3 모델이 더 작고 세부적인 구조의 상징 이미지를 해석하는 데 어려움이 있음을 알 수 있다.

ResNet-101, GRU 모델을 사용했을 때는 긴 시퀀스 길이에 대해 더 낮은 성능을 보였지만, 타 모델에 비해 작은 격차를 보인다. ResNet-101, LSTM 모델을 사용했을 때는 긴 시퀀스 길이에 대해 더 높은 성능을 보인다. ResNet-101 모델이 이미지로부터 풍부한 특징을 추출할 수 있으며, LSTM이 장기 의존성 문제를 더 효과적으로 해결할 수 있기 때문에 나타난 결과로 보인다.

EfficientNet-B3를 활용한 모델 또한 ResNet-101 활용 모델과 유사한 경향을 보이지만, 짧은 길이의 시퀀스에 대해서는 GRU, LSTM 모델 간 큰 차이가 나타나지 않았다.

[표 11]은 모델의 번역 성능을 BLEU-1, BLEU-2, BLEU-3, BLEU-4로 측정한 결과이다. 모든 모델에서 단어 수준의 1-gram으로 유사도를 측정한 결과인 BLEU-1이 가장 높은 점수를 보였으며, 문장의 흐름을 평가하는 4-gram으로 유사도를 측정한 결과인 BLEU-4에서 가장 낮은 점수를 보였다.

BLEU 점수를 통해 번역 성능만을 확인했을 때는 ResNet-101과 LSTM을 사용한 모델이 가장 우수한 성능을 보였다. 그러나 연산 비용 측면에서는

ResNet-101보다 EfficientNet-B3가, LSTM보다는 GRU 구조가 효율적일 수 있다. 실제 번역 기능을 실시간의 모바일 프로그램에 적용하기 위해서는 번역 성능, 연산 시간, 메모리 비용을 복합적으로 고려하여 모델을 선정할 필요가 있다.

[표 11] BLEU-1,2,3,4 평가 결과

max_seq_length	Model	BLEU-1	BLEU-2	BLEU-3	BLEU-4
4	Inception-v3 +GRU with Attention	78.596	74.158	70.425	64.567
	Inception-v3 +LSTM with Attention	82.127	78.879	76.934	71.968
	ResNet-101 +GRU with Attention	83.956	80.242	77.096	71.268
	ResNet-101 +LSTM with Attention	**84.401**	**81.294**	**79.262**	**73.998**
	EfficientNet-B3 +GRU with Attention	83.048	79.456	76.430	70.420
	EfficientNet-B3 +LSTM with Attention	82.258	78.600	76.201	71.083
6	Inception-v3 +GRU with Attention	74.995	70.163	66.839	63.526
	Inception-v3 +LSTM with Attention	75.903	71.520	69.592	67.409
	ResNet-101 +GRU with Attention	81.230	77.134	74.027	70.074
	ResNet-101 +LSTM with Attention	**84.221**	**80.761**	**78.702**	**75.635**
	EfficientNet-B3 +GRU with Attention	79.137	74.652	71.347	67.674
	EfficientNet-B3 +LSTM with Attention	**83.042**	**79.333**	**77.231**	**74.588**

V. 결론 및 향후 연구

언어 장애인은 AAC 앱을 통해 다양한 장소와 상황에서 의사소통의 도움을 받고 있으나, 일련의 AAC 상징들로 자신의 메시지를 자연스럽게 표현하는 데 어려움이 있다. 또한, 비대면 의사소통이 필요한 온라인 서비스가 증가함에 따라 비장애인과 소통을 보조할 수 있는 추가적인 도구가 필요하다. 이를 위해 기존 연구에서는 AAC 상징 시퀀스와 한국어 문장 간 변환 기능을 제안하였으며, AAC 상징 어휘, 아이디 정보를 활용하였다. 그러나 의사소통 장애인은 AAC 그림 상징의 이미지를 보고 어휘를 유추하여 선택한다는 점과 AAC 그림 상징 이미지 자체가 문맥상에서 다양한 어휘를 나타낼 수 있다는 점을 고려했을 때 한국어 문장을 생성하는 데 이미지 정보를 활용할 필요가 있다.

본 연구에서는 상징의 그림 정보를 활용하기 위해 Visual Encoder를 적용한 AAC 상징 시퀀스의 한국어 문장 생성 모델을 제안하였다. Visual Encoder로 사전 학습된 합성곱 신경망 모델을 이미지 특성 추출기로 활용하여 이미지를 이해하고, Text Decoder로는 Attention Mechanism 기반의 순환 신경망 모델을 이용하여 한국어 문장을 생성하였다.

AAC 상징 시퀀스 내에서 각 상징이 갖는 의미를 동시 반영하기 위해 최대 시퀀스 길이가 4, 6인 경우로 나누어 각각 2×2, 3×2로 병합하였다. 병합한 이미지 구조에 적합한 이미지 특성 추출기를 알아보기 위해, ResNet-101[11], Inception-v3[12], EfficientNet-B3[13] 모델을 각각 적용하여 실험하고 분석하였다. 또한 텍스트 생성 시 장기 의존성 문제를 해결할 수 있는 GRU[15], LSTM[14] 모델을 실험하였다.

BLEU 점수를 통해 번역 성능을 평가한 결과, Visual Encoder로는 ResNet-101이 가장 우수한 번역 성능을 보였으며, Inception-v3가 가장 낮은 번역 성능을 보였고 EfficientNet-B3는 ResNet-101보다 약간 낮은 성능을 보였다. ResNet-101이 가장 풍부한 이미지 특징을 추출할 수 있다. EfficientNet-B3가 ResNet-101 대비 연산량이 약 1/3 정도로 적기 때문에, 이러한 계산 비용을 생각했을 때 EfficientNet-B3가 충분히 좋은 성능을 보임을 확인하였다. Text Decoder로는 GRU보다 LSTM을 사용했을 때 더 우수한 번역 성능을 보인다. LSTM이 GRU보다 연산량이 많고 두 모델 간 성능은 유사하다고 알려져 있으나 복잡한 구조의 상징 이미지 특징을 처리한다는 점에서 LSTM 구조가 적합하였다.

본 논문에서는 AAC 그림 상징의 이미지를 기반으로 한국어 문장을 생성하였으며, 상징 시퀀스의 이미지에 적합한 Visual Encoder와 Text Decoder의 모델을 확인하였다. 실제 AAC 사용자의 선택 과정을 고려해 시각적 정보를 활용했으며 시퀀스 내에서 달라질 수 있는 상징 이미지의 특성을 고려했다는 점에서 의의가 있다. 본 연구의 모델은 이미지 캡셔닝의 전통적 구조에 기반하기 때문에, 텍스트와 유사한 임베딩을 추출할 수 있는 CLIP[36], BLIP[37] 등 다양한 최신의 멀티모달 학습 모델을 활용해 연구한다면 더 향상된 성능을 기대할 수 있다. 또한, 데이터의 다양성과 현실성의 확보가 필요하다는 점과 대화 맥락에 따라 달라지는 상징 시퀀스의 의미를 반영하지 못했다는 점에서 이를 보완하기 위한 추가 연구가 필요하다.

현재 구축된 데이터는 AI Hub의 일상 오피스 대화 문장을 중심으로 구축되어있으며, 상징 시퀀스가 최대 20개의 상징으로 구성되어 있어, 일반적으로 AAC 사용자가 비대면 상황, 소셜 네트워크 상의 대화에서 사용하는 상징 시

퀀스를 적절하게 포함하고 있다고 보기 어렵다. 따라서, 소셜 네트워크 서비스상 대화를 고려한 AI Hub의 한국어 SNS 멀티턴 대화 문장과 같이 실제 AAC 사용자가 활용하는 상징 시퀀스를 추가 수집하고 상징 시퀀스의 길이를 짧게 구성하여, 데이터셋을 실제와 가깝도록 보완 및 강화할 필요가 있다.

AAC 상징 시퀀스의 한국어 문장 생성에서 중요한 문제 중 하나는 AAC 그림 상징이 갖는 다의성이다. AAC 그림 상징의 다양한 어휘를 나타내는 것은 AAC 사용자의 환경과 대화 맥락의 영향이 크다. 본 연구에서는 대화 중 한 시점의 AAC 상징 시퀀스만으로 한국어 문장을 생성하였기 때문에 본질적인 상징의 다의성을 해소했다고 보기 어렵다. 실제 대화형 서비스에 적용하기 위해서는, 이전 대화의 맥락을 고려한 그림 상징 시퀀스의 문장 생성을 시도할 필요가 있다. 이전 대화의 텍스트 특징을 함께 데이터로 사용하여 멀티 모달 학습을 시도한다면, 이전 대화 맥락을 함께 고려할 수 있어 그림 상징의 다의성을 보다 효과적으로 해소하는 한국어 문장 생성을 기대할 수 있을 것이다.

참 고 문 헌

[1] D. R. Beukelman and P. Mirenda, “Augmentative and alternative communication: Supporting children and adults with complex comminication needs,” 3rd ed. Baltimore, MD: Brookes Publishing, 2006.

[2] 박은혜, 김영태, 홍기형, 연석정, 김경양. “이화-AAC 상징체계개발연구,” 보완대체의사소통연구, vol. 4, no. 2, pp. 19-40, 2016.

[3] S. Shin, Y. Kim, E. Park, “A Study on the Verification of AAC Graphic Symbols Focusing on Nouns, Adverbs, and Verbs,” Communication Sciences & Disorders, vol. 22, no. 3, pp. 597 - 607, Sep. 2017.

[4] J.-S. Kwon, “The Mediating Effect of Digital Literacy Competency between Social Support of the Disabled and Level of Changes in Digital Information Service on the Disabled after COVID-19,” Journal of Digital Contents Society, vol. 24, no. 7, pp. 1545 - 1554, Jul. 2023.

[5] I. A. Khan and N. W. Paliwal, “ChatGPT and digital inequality: A rising concern,” Scholars Journal of Applied Medical Sciences, vol. 11, no. 09, pp. 1646 - 1647, Sep. 2023.

[6] 조희. “한국형 AAC 그림 상징 시퀀스의 딥러닝 기반 텍스트 생성,” 국내석사학위논문 성신여자대학교 대학원, 2021.

[7] 안서영. “딥러닝 기반 보완대체의사소통 상징의 다의성을 반영한 상징 시퀀스의 한국어 문장 변환.” 국내석사학위논문 성신여자대학교 일반대학원, 2023.

[8] 이주현. “딥러닝 기반 대화 문장의 보완대체의사소통 상징 시퀀스 변환.” 국내석사학위논문 성신여자대학교 일반대학원, 2021.

[9] 유세희. “다중 의미를 갖는 보완대체의사소통 상징의 구문분석을 통한 딥러닝 기반 문장의 상징 시퀀스 변환.” 국내석사학위논문 성신여자대학교 일반대학원, 2023.

[10] D. Bahdanau, K. Cho, and Y. Bengio, “Neural machine translation by jointly learning to align and translate,” CoRR, vol. abs/1409.0473, 2014.

[11] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” 2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR), pp. 770 - 778, 2016.

[12] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 2818 - 2826, 2016.

[13] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” International Conference on Machine Learning (ICML), pp. 6105 - 6114, May 2019.

[14] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, vol. 9, no. 8, pp. 1735 - 1780, 1997.

[15] K. Cho, B. Merrienboer, C. Gulcehre, F. Bougares, H. Schwenk, and Y. Bengio, “Learning phrase representations using RNN encoder-decoder for statistical machine translation,” 2014.

[16] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, “BLEU: a method for automatic evaluation of machine translation,” Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp. 311 - 318, 2002.

[17] 리드스피커코리아, “마이토키,” [Online]. Available: https://www.mytalki

e.co.kr/. [Accessed: Nov. 5, 2024].

[18] 조희, 홍기형, “GeoAAC, 위치기반 보완대체의사소통 모바일 앱,” 보완대체의사소통연구, vol. 8, no. 1, pp. 87 – 117, 2020.

[19] 엔씨문화재단, “나의 AAC,” [Online]. Available: https://www.myaac.or.kr/. [Accessed: Nov. 5, 2024].

[20] Keras, “Embedding layer,” [Online]. Available: https://keras.io/api/layers/core_layers/embedding/. [Accessed: Nov. 5, 2024].

[21] FastText, “Word Vectors for 157 languages,” [Online]. Available: https://fasttext.cc/docs/en/crawl-vectors.html. [Accessed: Nov. 5, 2024].

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” North American Chapter of the Association for Computational Linguistics, 2019.

[23] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, “BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2019.

[24] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, “Language models are unsupervised multitask learners,” 2019.

[25] 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,” Proceedings of the 32nd International Conference on Machine Learning, vol. 37, pp. 2048 – 2057, 2015.

[26] AI Hub, “AI Hub,” [Online]. Available: https://aihub.or.kr/. [Accessed: Nov. 5, 2024].

[27] 천춘경. “보완 · 대체 의사소통 (AAC) 체계 활용을 위한 지역사회중심의 기초어휘 및 문장 조사,” 국내석사학위논문 단국대학교 대학원, 2000.

[28] 김수미. “AAC를 활용한 함께 책 읽기 중재가 복합의사소통장애 학생의 의미 관계 표현과 어휘다양도 변화에 미치는 효과,” 국내석사학위논문 창원대학교 대학원, 2019.

[29] 박은혜. “보완/대체의사소통체계를 위한 기초어휘조사: 뇌성마비 초등 저학년 학생을 중심으로,” 특수교육논총, vol. 13, no. 1, pp. 91-115, 1996.

[30] 이정은, 박은혜. “보완· 대체의사소통체계 적용을 위한 상황 중심 핵심어휘 개발 연구,” 재활복지, vol. 4, pp. 64- 122, 2000.

[31] 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, pp. 248 - 255, 2009.

[32] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1 - 9, 2015.

[33] E. L. Park and S. Cho, “KoNLPy: Korean natural language processing in Python,” Proceedings of the 26th Annual Conference on Human & Cognitive Language Technology, Chuncheon, Korea, Oct. 2014.

[34] Google Cloud, “Evaluate models using AutoML Translation,” [Online]. Available: https://cloud.google.com/translate/automl/docs/evaluate?hl=ko. [Accessed: Nov. 5, 2024].

[35] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, “Learning transferable visual models from natural language supervision,”

2021.

[36] J. Li, D. Li, C. Xiong, and S. Hoi, “BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation,” 2022.

ABSTRACT

Deep Learning-Based Korean Sentence Generation from Augmentative and Alternative Communication Pictorial Symbol Sequences Using Visual Encoders

Jiwon Lee
Department of Future Convergence Technology Engineering
Graduate School of
Sungshin University

Augmentative and Alternative Communication(AAC) is a communication tool for people with difficulties in communication. The most representative AAC is a pictorial symbol system. People with language disorder select at least one symbol to make their own message, which is called an AAC symbol sequence.

Many mobile applications support communication of using AAC, but they are weak in online communication since they were developed with a focus on face-to-face communication. As many services such as conversations, meetings, orders, payments are converted to online services, there are also increasing difficulties in non-face-to-face communications. To solve these problems, converting between AAC symbol sequences and Korean sentences is a crucial feature, and active research is ongoing in this area. However, most of the research has been done based on the textural information of AAC symbols such as their

expression vocabularies and identifiers.

This paper proposed Korean sentence generation models based on the visual information (images) of AAC symbol sequences. For training data, the symbol sequence-Korean sentence dataset from previous studies was utilized. The images of AAC symbol sequences were merged into a single image to build the symbol sequence image-Korean sentence dataset. Considering the area occupied by each symbol in the image of a symbol sequence, the dataset was classified based on the maximum length of AAC symbol sequences, with cases of length 4 and 6, and each was combined into grid formats of 2×2 and 3×2. To extract image features, we used pre-trained convolutional neural networks such as ResNet-101, Inception-v3 and EfficientNet-B3 as visual encoders. We also used recurrent neural networks such as LSTM (Long Short Term Memory) and GRU(Gated Recurrent Unit) as text decoders for generating Korean sentences based on the image features. An attention mechanism was applied to prioritize key image regions when generating each text from combined AAC images. A total of six models were experimented with, combining the three visual encoders and two text decoders. As an evaluation metric, we used BLEU(Bilingual Evaluation Understudy) scores which measures the similarity between machine generated translations and reference translations. Among the visual encoders, ResNet-101 tended to show the best performance, while in the case of text decoders, models with LSTM generally outperformed those with GRU. Note that similar results were observed for both 2×2 and 3×2 combined AAC images in ResNet-101 and EfficientNet-B3 with LSTM as the text decoder. The proposed six models achieved BLEU scores ranging from 60 to 80, indicating high-quality translation performance.