홍 기 형 교수 지도

석사학위 청구논문

한국형 AAC 그림 상징 시퀀스의 딥러닝 기반 텍스트 문장생성

2021

성신여자대학교 대학원

컴퓨터학과

조 희

한국형 AAC 그림 상징 시퀀스의 딥러닝 기반 텍스트 문장생성

홍 기 형 교수 지도

이 논문을 석사학위논문으로 제출함

2020년 11월

성신여자대학교 대학원

컴퓨터학과

조 희

인 준 서

조희의 석사학위 논문으로 인준함

2020년 11월

심사위원장 ..... 서명 또는

심 사 위 원 ..... 서명 또는

심 사 위 원 ..... 서명 또는

성신여자대학교 대학원

논문개요

보완대체의사소통(Augmentative and Alternative Communication; AAC)은 그림, 몸짓과 같은 비구어적 방법을 사용한 의사소통 방법으로, 구어를 사용한 의사소통에 어려움이 있는 사람들은 AAC 상징을 사용하여 자신의 의사를 표현한다. 모바일기기와 소프트웨어의 발전으로 AAC 사용자들의 어플리케이션 활용도가 높아져, AAC 어플리케이션 이외에도 다양한 사회적 네트워크 서비스 사용에 대한 욕구가 클 것으로 생각된다. 하지만 현실적으로 일반인들의 AAC 이해도가 높지 않아, 일상생활에서 AAC 상징을 사용한 의사소통에 어려움이 있다.

본 논문은 국내 AAC 사용자들이 다양한 모바일 플랫폼에서 원활한 의사소통을 할 수 있도록, 딥러닝을 활용하여 한국형 AAC 그림 상징 시퀀스를 한국어 문장으로 생성하는 것을 목적으로 한다. 학습을 위한 데이터로 한국형 AAC 그림 상징 시퀀스 데이터를 구축하였으며, 상징 식별자 기반 시퀀스와 상징 어휘 기반 시퀀스로 표현할 수 있다. 전처리 과정에서 토큰화 방법으로 상징 단위 토큰화 방법과 형태소 단위 토큰화 방법을 사용하였으며, 학습 모델로 게이트 순환 유닛을 사용한 시퀀스-투-시퀀스 모델과 어텐션 메커니즘을 적용한 모델 2가지를 활용하여 토큰화 방법에 따른 모델별 문장생성 성능을 비교하였다. 결과적으로 상징 어휘 기반 시퀀스는 형태소 단위 토큰화 방법을 사용한 어텐션 기반 시퀀스-투-시퀀스 모델에서 가장 높은 성능을 보인 것을 확인하였다.

목 차

표 목 차

그 림 목 차

I. 서 론

의사소통은 사회를 살아가면서 사람과 관계를 만드는데 기본 요소이며 인간이 누려야 할 권리이다. 인간은 사회적 욕구가 있어 의사소통을 통해 다른 사람들과 관계를 만들며 살아가고 싶어한다. 하지만 의사소통 장애인들은 구어를 사용한 의사소통에 한계가 있어, 사회 구성원들과 소통하며 관계를 만드는 것에 많은 어려움을 느낀다.

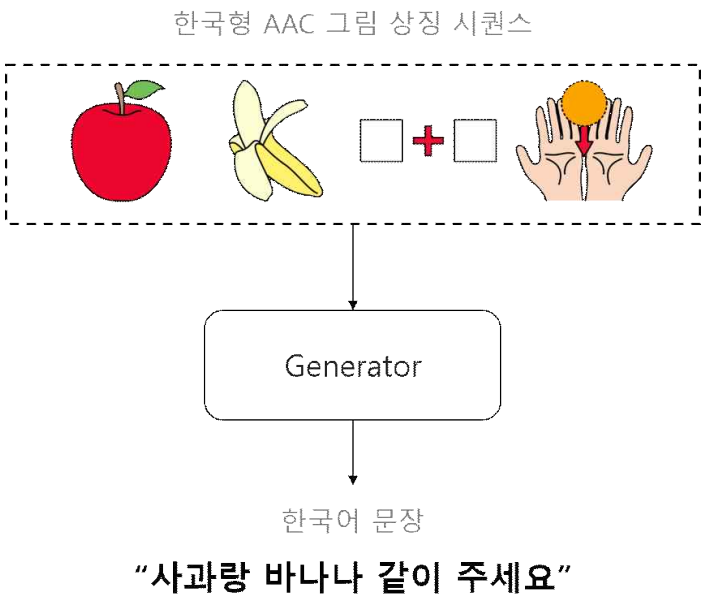
보완대체의사소통(Augmentative and Alternative Communication; AAC)은 구어를 사용한 의사소통에 장애를 보이는 사람들에게 의사소통의 기회를 주기 위하여 그림과 같은 대체적인 방법을 사용한 의사소통 방법을 말한다[1]. 의사소통 장애인은 AAC 구성 요소 중 하나인 상징(Symbol)을 사용하여 자신의 의사를 표현하며, 다양한 상징체계 중 그림 상징을 가장 많이 사용한다[2].

AAC 그림 상징은 오프라인 공간에 있는 상대방과의 대화 상황뿐만 아니라 모바일 메신저(예: 카카오톡)와 같이 온라인 공간에서도 사용할 수 있다. [3]의 AAC 현장 적용을 살펴보면 AAC 사용자들이 그림 상징들을 나열하여 편지쓰기, 자기소개 등 짧은 문장의 형태로 자신의 의사를 표현하는 것을 알 수 있다. [4]의 연구에서는 AAC 어플리케이션과 모바일 메신저를 연동하여 원거리 상황에서도 AAC를 활용한 의사소통이 가능하다는 것을 알 수 있다. 하지만 사회에는 AAC를 모르는 일반인(비장애인)이 대다수이기 때문에 일상생활에서 AAC 그림 상징을 사용하는 데 어려움이 있다.

본 논문에서는 AAC를 위한 어플리케이션, 카카오톡과 같은 사회적 네트워크 서비스(Social Networking Service; SNS) 등 다양한 온라인 플랫폼에서 AAC 사용자의 AAC 상징들을 이용한 의사 표현을 일반인이 이

해하는 텍스트 문장으로 변환하는 모델을 제안한다. 제안한 모델은 시퀀스-투-시퀀스 모델을 기반으로 기존의 순환 신경망 대신 게이트 순환 유닛을 사용하며, 중요한 단어에 더 집중하는 어텐션 메커니즘을 적용한다. 학습에 사용되는 데이터는 한국형 AAC 상징 체계집[2, 5]과 한국어 대화 문장 데이터[6]를 바탕으로 한국형 AAC 그림 상징 시퀀스 데이터를 구축하여 사용한다.

[그림 1]은 한국형 AAC 그림 상징 시퀀스를 한국어 문장으로 생성하는 예시를 나타낸 것이다. '사과', '바나나', '합쳐요', '주세요'의 어휘로 구성된 한국형 AAC 그림 상징 시퀀스를 입력으로 하여, 자연스러운 한국어 문장인 "사과랑 바나나 같이 주세요"를 출력하는 모델(Generator)이다.



[그림 1] 제안 모델 예시

본 논문의 구성은 다음과 같다. 2장에서는 AAC 상징을 문장으로 생성하는 관련 연구와 제안 모델에 필요한 이론 배경에 대해 알아보고 3장에서는 한국형 AAC 그림 상징 시퀀스 데이터 구축 과정을 살펴본다. 4장에서는 한국형 AAC 상징 시퀀스의 딥러닝 기반 문장생성을 위한 전처리 과정과 제안한 모델을 설명한다. 5장에서는 실험 환경, 과정, 결과 그리고 모델의 성능 평가를 진행하고 마지막으로 6장에서는 결론과 향후 연구에 대해 제시한다.

II. 관련 연구 및 이론 배경

1. 보완대체의사소통

1) 보완대체의사소통 구성 요소 및 구문 특징

보완대체의사소통(AAC)은 말이나 글을 사용한 의사소통에 어려움을 느끼는 사람들(예: 말실행증, 뇌성마비, 지적장애 등)의 의사소통 능력을 보완하거나 대체하기 위하여 그림, 몸짓이나 손짓과 같은 비언어적인 방법을 사용한 의사소통 방법을 뜻한다[1]. AAC의 사용은 일시적으로 대체적인 의사소통 방법이 필요할 때 사용되기도 하지만, 영구적인 결함으로 인해 AAC를 하나의 언어처럼 사용하기도 한다[7].

AAC를 사용하기 위해서는 상징(Symbols), 도구(Aids), 기법(Techniques), 전략(Strategies) 총 4가지 요소가 필요하다[7].

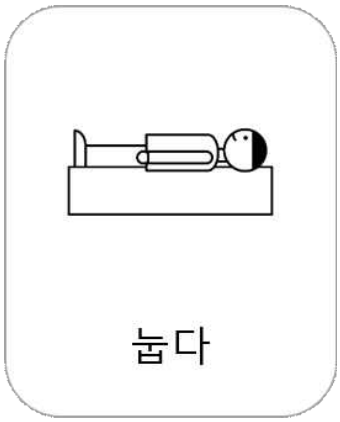
- 상징은 언어의 어휘나 문장에 해당하는 대체적인 표현 방법을 뜻한다. 모형, 그림과 같은 도구적(Aided) 상징과 몸짓, 수화와 같은 비도구적(Unaided) 상징으로 구분할 수 있다.
- 도구는 상징체계를 담기 위한 물리적인 도구를 뜻한다. 전자적인 성능에 따라, 전자기능이 없는 AAC, 단순한 전자기기를 사용한 AAC, 첨단 전자기기를 사용한 AAC 등으로 분류한다.
- 기법은 상대방에게 자신의 의사를 전달하는 방법으로, 직접 메시지를 선택하는 방법과 프로그램을 사용한 스캐닝 방법 등이 있다.
- 전략은 AAC 사용자가 자신의 의사소통 메시지를 전달할 때의 정확

성, 시간의 효율성 등을 높이는 방법을 뜻한다. 예를 들어 AAC 사용자를 위한 문장 예측 전략 등이 이에 해당한다.

AAC 사용자 중 복합의사소통장애아동들의 AAC 구문은 다음과 같은 특징이 있다[7].

- 1~2개의 단어로 이루어진 메시지가 주를 이루어 대부분 단문으로 자신의 의사를 표현한다.
- 문법보다는 내용 중심의 단어들이 많이 사용되어 동사나 관사 등 생략되는 경우가 흔하다.
- 이 외에도 상징으로 표현되지 않은 메시지를 전달할 때는 메타적인 언어 전략을 사용한다.

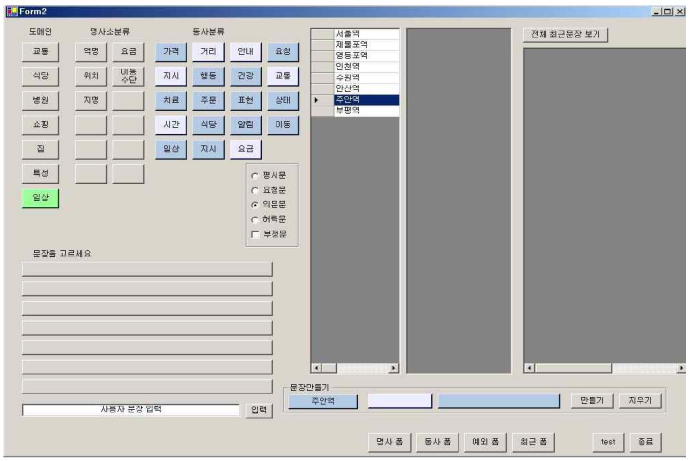
AAC 상징을 사용한 의사소통은 구어와 기능적인 차이가 있다. 예를 들어 [그림 2]는 한국형 AAC 상징 체계집의 '눕다' 상징이다. '눕다' 상징은 사람이 누워있는 그림으로 표현되어있어, '눕다' 상징 하나로 '사람이 누워 있어요' 라는 문장을 충분히 표현할 수 있다.



[그림 2] '눕다' 상징

2) 보완대체의사소통을 활용한 문장생성 연구 동향

AAC 사용자는 대체적인 표현인 상징(예: 그림, 사진, 글자 등)을 사용하기 때문에, AAC에 대한 사전지식이 없는 상대방은 AAC 사용자의 의사 표현을 이해하는 데 어려움을 느껴 대화를 이어가는데 다소 시간이 걸린다. AAC 사용자와의 원활한 의사소통을 위하여, 자연어 처리 분야 기술을 접목한 AAC에 관한 연구는 지속적으로 발전되어 왔다.



[그림 3] 명사에 연결 가능한 동사 종류

[8]의 연구에서는 언어장애인들의 원활한 일상적인 대화를 위하여 어휘정보를 구축하고, 사용자가 선택한 어휘를 바탕으로 문법 패턴을 사용한 문장생성 시스템을 구축하였다. 시스템의 문장생성 단계는 총 9단계로 구성되어 있어, 언어장애를 포함한 중복장애가 있는 사용자들에게는 다소 어려움이 있다. 또한, 구축된 어휘정보만 사용하기 때문에 생성할 수 있는 문장의 수가 제한적이다. [그림 3]은 [8]에서 구축한 시스템으로 명사에 연결 가능한 동사 종류를 제시한 것이다.

[그림 4] 보완대체의사소통 도구

[9]의 연구에서는 이미지에 대한 의미와 범주를 고려한 이미지 사전을 구축하고, 동사에 초점을 맞춰 좀 더 자연스러운 형태의 문장을 생성하는 시스템을 구축하였다. [그림 4]는 [9]의 연구에서 구축한 문장 생성시스템이다. 하지만 부사, 관형어 단어들은 이미지로 표현하기 어려워 배제되었으며, 다른 단어나 문장을 수식하는 것도 고려하지 않아 3형식 이상의 문장생성이 불가능하다.

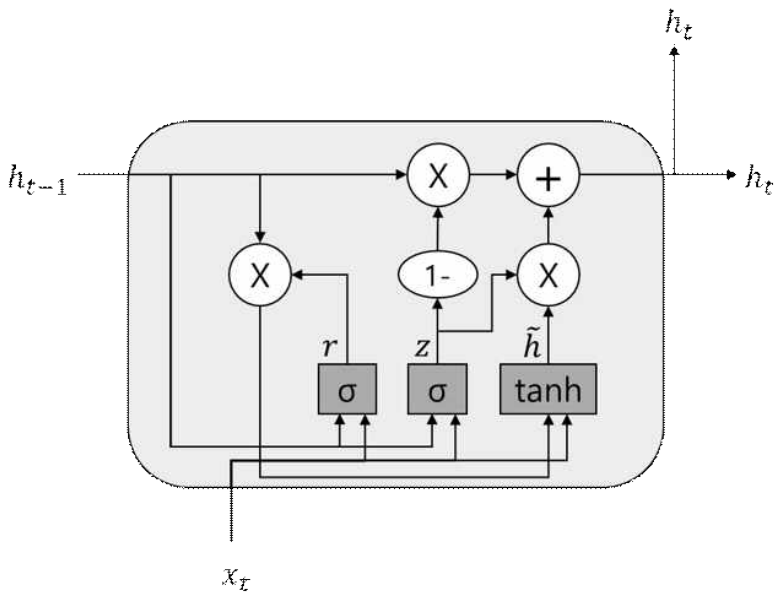
이외에도 [10]의 연구는 그림과 어휘로 이뤄진 아이콘을 정의하고 문법적 체계를 갖췄지만, 아이콘 언어가 활용 영역에 의존적인 형태이며 아이콘이 다중적 의미와 품사에 대한 애매성을 나타낸다. [11]의 연구에서도 사용자가 선택한 단어를 바탕으로 문법적 함수와 사용자에 대한 지식 기반 정보 등을 활용하여 문장을 생성하지만, 제한된 도메인에서만 사용할 수 있고 한국어를 지원하지 않아 한국 사회에서 사용하는 데 어려움이 있다.

2. 딥러닝 이론 배경

1) 게이트 순환 유닛(Grated Recurrent Unit)

순환 신경망(Recurrent Neural Network; RNN)은 출력값이 이전 스탭(Step)의 연산 결과에 영향을 받는 순환적인 구조로 이루어진 인공 신경망으로, 문장과 음성 같은 시퀀스 데이터를 처리하는 연구에 널리 활용되고 있다. 하지만 순환 신경망은 데이터들의 시점 간격이 커질수록 앞에서 전달받은 정보를 점차 잃어 뒷부분까지 영향을 미치지 못하는 장기 의존성(Long-Term Dependency) 문제를 가지고 있다[12]. 이 문제로 시퀀스 길이가 긴 데이터를 학습하는 데 한계를 보인다. 이를 해결하기 위해 장단기 메모리(Long Short-Term Memory; LSTM)[13], 게이트 순환 유닛(Grated Recurrent Unit; GRU)[14] 등 제안된 방법 중 본 논문에서는 GRU를 사용하여 기존 RNN의 문제점을 보완하고자 한다.

GRU는 기존 RNN 구조에 리셋 게이트(Reset gate)와 업데이트 게이트(Update gate)를 추가한 RNN의 종류로, 장기 의존성 문제를 해결하면서 LSTM 구조보다 간단화 시켜 계산 시간을 줄였다[14]. 매개변수와 계산량이 적기 때문에 데이터의 양이 적거나 모델 설계 시 반복적인 일을 해야 하는 경우 적합하다[15]. [그림 5]는 GRU의 계산 그래프이다. 여기서 σ는 시그모이드(sigmoid) 활성화 함수, tanh은 하이퍼볼릭 탄젠트 함수(Hyperbolic Tangent Function; tanh), x_t는 t 시점의 입력값, h_{t-1}와 h_t는 t-1 시점과 t 시점의 은닉 상태의 출력값을 의미한다.



[그림 5] GRU 계산 그래프 [15]

r은 이전 은닉 상태의 출력 값 h_{t-1}을 얼마나 적용할지 정하는 리셋 게이트로, 이전의 상태를 얼마나 무시할지 결정하는 역할을 한다. 수식 (1)은 리셋 게이트 r의 수식으로, W_x^r, W_h^r은 x_t와 h_{t-1}의 가중치, b^r은 편향을 의미하며 시그모이드 활성화 함수를 통해 0과 1 사이의 값을 출력하도록 한다.

$$r = \sigma(x_t W_x^r + h_{t-1} W_h^r + b^r) \tag{1}$$

수식 (2)는 새로운 은닉 상태($\tilde{h}$)를 나타낸다. 여기서 W_x, W_h는 가중치, b는 편향, $\odot$은 곱셈을 뜻한다. 만약 r이 0이면 이전 은닉 상태 h_{t-1}은 완전히 무시되어 새로운 은닉 상태($\tilde{h}$)는 t 시점의 입력 값 x_t만으로 결정된다. $\tilde{h}$는 활성화 함수로 tanh를 사용하여 -1부터 1 사이의 값을 출력한다.

$$\tilde{h} = \tanh(x_t W_x + (r \odot h_{t-1}) W_h + b) \tag{2}$$

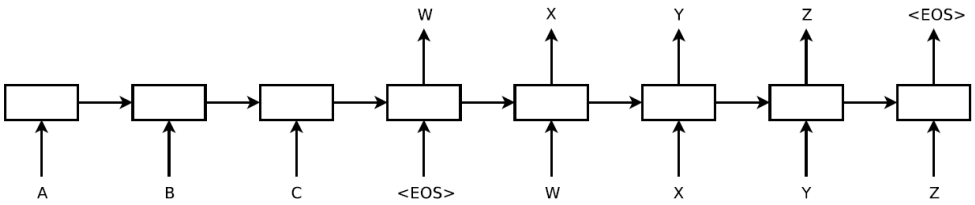
업데이트 게이트 z는 t 시점의 은닉 상태를 갱신하는 게이트로 과거와 현재 정보의 갱신 비율을 시그모이드 활성화 함수를 통해 결정한다. 업데이트 게이트 z는 수식 (3)과 같으며 W_x^z, W_h^z는 가중치, b^z는 편향을 나타낸다.

$$z = \sigma(x_t W_x^z + h_{t-1} W_h^z + b^z) \tag{3}$$

업데이트 게이트 z는 과거의 은닉 상태의 정보를 삭제하고 새로 추가된 정보에 가중치를 부여한다. 수식 (4)의 $(1-z) \odot h_{t-1}$이 정보를 삭제하는 연산이고, $z \odot \tilde{h}$ 가 새로 추가된 정보에 가중치를 부여하는 연산이다. 만약 z가 1이라면 h_{t-1}은 전부 잊게 되고 $\tilde{h}$를 모두 기억하게 된다.

$$h_t = (1-z) \odot h_{t-1} + z \odot \tilde{h} \tag{4}$$

2) 시퀀스-투-시퀀스(Sequence-to-Sequence)



[그림 6] 시퀀스-투-시퀀스 모델 [16]

시퀀스-투-시퀀스(Sequence-to-Sequence; Seq2Seq) 모델은 입력된 시퀀스로부터 다른 도메인의 시퀀스로 출력하는 모델로 챗봇, 기계 번역, 음성 인식 등 입력과 출력이 시계열 데이터인 도메인에 많이 적용되는 모델이다[16]. Seq2Seq 모델은 순환 신경망을 기반으로 한 인코더(Encoder)와 디코더(Decoder) 이루어져 있으며, 인코더-디코더(Encoder-Decoder) 모델이라 불린다[15].

인코더는 입력 시퀀스 $X=[x_1,x_2,...,x_n]$ 를 구성하는 토큰 $x_1,x_2,...,x_n$을 순서대로 순환 신경망의 입력으로 하며, 각 순환 신경망을 통해 입력 시퀀스의 특징을 반영하는 값으로 추출한다. 마지막 시점의 순환 신경망 출력인 은닉 상태(Hidden state) h는 디코더에 필요한 정보가 하나의 벡터로 응축된 것으로 문맥 벡터(Context vector) v로 간주한다. [그림 6]은 Seq2Seq 모델의 예시이다. 입력 시퀀스로 문장 “ABC”를 문자 단위로 토큰화하고 임베딩 과정을 통해 벡터화된 값을 순환 신경망에 입력한다. 문장의 끝을 나타내는 “<EOS>” 토큰이 나오면 문맥 벡터 v를 출력한다.

디코더는 문맥 벡터 v를 초기 상태로 활용하고, 출력 시퀀스 $Y=[y_1,y_2,...,y_m]$ 를 구성하는 토큰 $y_1,y_2,...,y_m$을 각 순환 신경망의 입력으로 한다. 디코더는 t 시점의 출력 값을 t+1 시점의 입력 값으로 사용하여, t+1 시점의 순환 신경망 출력이 토큰 y_{t+1}로 출력될 확률이 최대가 되도록 학습을 진행한다. 예를 들어 출력 시퀀스 문장 “WXYZ” 의 첫 번째 문자 ‘W’와 문맥 벡터 v를 순환 신경망의 입력으로 사용하여, 그다음 시퀀스인 ‘X’가 출력되는 확률이 최대가 되도록 학습한다. 디코더도 마찬가지로 “<EOS>” 토큰이 나올 때까지 이 과정을 반복하여 출력 시퀀스로 변환한다.

전체 모델을 식으로 표현하면 수식 (5)와 같다. 길이 T의 입력 시퀀스와 길이 T’의 출력 시퀀스가 주어졌을 때 문맥 벡터 v를 활용하여, 입력 시퀀스에 대한 출력 시퀀스의 조건부 확률 $p(y_1,...,y_{T'}|x_1,...,x_T)$ 를 구하는 것이다[16].

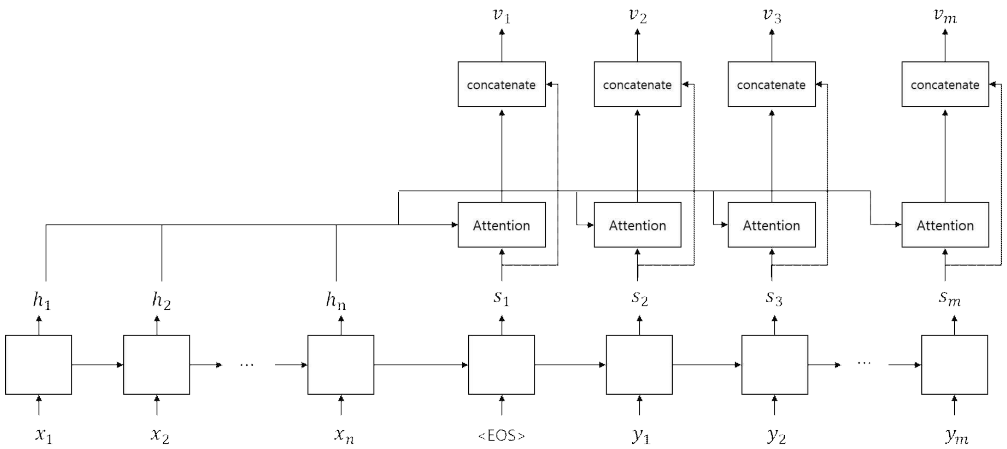
$$p(y_1,...,y_{T'}|x_1,...,x_T)=\prod_{t=1}^{T} p(y_t|v,y_1,...,y_{t-1}) \quad (5)$$

본 논문에서는 Seq2Seq 모델을 활용하여 입력 시퀀스로 한국형 AAC 그림 상징 시퀀스를 받아 인코더를 통해 상징 시퀀스의 특징을 파악하고, 디코더를 통해 한국어 문장 시퀀스로 출력하고자 한다.

3) 어텐션 메커니즘(Attention Mechanism)

기존 순환 신경망에 기반한 Seq2Seq 모델은 입력 시퀀스를 하나의 고정된 크기의 벡터로 표현하기 때문에 입력 시퀀스의 길이가 길어질수록 정보 손실이 발생하며, 순환 신경망의 문제인 기울기 소실(Gradient vanishing) 문제도 존재한다. 이를 해결하기 위해 중요한 부분에 집중하게 하는 기법인 어텐션 메커니즘(Attention Mechanism)을 활용한다[17].

어텐션 메커니즘의 기본 아이디어는 디코더에서 출력 시퀀스를 예측하는 매 시점(Time step)마다, 인코더의 전체 입력 시퀀스를 다시 한번 참조하는 것이다. 이때, 전체 입력 시퀀스를 동일한 비율로 참조하는 것이 아니라 해당 시점에서 출력 시퀀스에 비교적 큰 영향력을 주는 입력 시퀀스 부분에 더 집중한다. 다음 [그림 7]은 어텐션 메커니즘을 적용한 Seq2Seq 모델을 도식화한 것이다.



[그림 7] 어텐션 메커니즘을 적용한 Seq2Seq 모델

어텐션 메커니즘은 4 가지 과정으로 나타낼 수 있다. 첫 번째 과정으로 디코더의 현재 시점 t의 은닉 상태 s_t 와 인코더의 모든 은닉 상태 $h1$, h_2, ..., h_n 값의 유사 정도를 계산하여 어텐션 점수(Attention Score)를 구한다. 어텐션 점수를 구하는 방법으로 닷-프로덕트 어텐션(Dot-Product Attention)을 사용하여 설명하도록 한다. 어텐션 점수를 구하는 함수는 수식 (6)이며, 수식 (7)의 e^t는 인코더의 모든 은닉 상태의 어텐션 점수 모음을 정의한 것이다.

$$score(s_t, h_i) = s_t^T h_i \tag{6}$$

$$e^t = [s_t^T h_1, ..., s_t^T h_n] \tag{7}$$

두 번째 과정으로 e^t에 Softmax 함수를 적용해 0과 1 사이의 값으로 정규화된 확률 분포인 어텐션 분포(Attention Distribution) α^t를 얻는다. α^t를 식으로 정의하면 수식 (8)과 같다. 이때 어텐션 분포 값 각각은 어텐션 가중치(Attention Weight)라고 한다.

$$\alpha^t = softmax(e^t) \tag{8}$$

세 번째 과정으로 인코더의 은닉 상태와 어텐션 가중치를 가중합(Weight Sum)을 하여 어텐션의 최종 결과값인 어텐션 값(Attention Value) a_t 를 계산한다. a_t는 인코더의 문맥을 포함하고 있어 문맥 벡터라고도 불린다.

마지막으로 어텐션 값 a_t와 디코더의 t 시점의 은닉 상태 s_t를 연결하여 하나의 벡터 v_t로 만든다. v_t는 예측 연산의 입력으로 사용되며, 인코더의 문맥 정보를 활용하여 더 잘 예측할 수 있도록 한다.

III. 한국형 AAC 그림 상징 시퀀스

1. 한국형 AAC 상징 체계집

한국형 AAC 상징 체계집은 한국 사회와 문화에 필요한 어휘를 바탕으로 개발된 AAC 그림 상징 체계집으로, 약 1만여 개의 그림 상징으로 구성되어 있다 [2, 5]. 한국형 AAC 상징은 상징 이미지와 상징 어휘로 구성되어 있으며, 한국 AAC 사용자를 위해 개발된 모바일, PC 등 다양한 AAC 어플리케이션에 많이 활용되고 있다[18-20]. 한국형 AAC 상징 체계집은 다음과 같은 특징을 가지고 있다.

- 아동부터 성인까지 전 연령대에서 사용 가능한 어휘, 다양한 장애 유형의 사용자들을 고려한 어휘, 한국 사회와 문화를 반영하는 어휘의 그림 상징들로 구성되어 있다. [그림 8]의 '설날' 상징과 '김치' 상징은 한국 사회와 문화를 반영하는 상징 중 하나이다.

[그림 8] 한국 문화와 사회를 반영한 상징

- 낱말 이외에 AAC 사용자들이 자주 사용하는 표현들을 짧은 구나 문장 형태의 상징으로 개발하였다. 예를 들어 “더 먹고 싶어요” 와 “놀이터에 갈래요” 는 AAC 사용자들이 자주 사용하는 문장으로 [그림 9]와 같이 문장형 상징으로 제작되었다.

[그림 9] 문장 형태의 상징

- 사용 정도가 높다고 판단되는 상용구 표현은 높임말과 반말 2가지 형태의 어휘로 제작되었다. [그림 10]의 왼쪽 ‘사랑해’ 상징은 반말 어휘이며, 오른쪽 ‘사랑해요’ 상징은 높임말 어휘이다.

[그림 10] 반말과 높임말 상징

- 아동이나 유아 AAC 사용자를 위하여 기존에 제작된 상징 중 일부를 새로운 컨셉으로 제작한 상징들이 있다. [그림 11]은 동일한 '같이놀자' 어휘이지만 왼쪽 상징은 전 연령대에서 사용이 가능한 상징이며, 오른쪽 상징은 아동 연령을 고려한 상징이다.

[그림 11] 연령을 고려한 상징

- 동사의 원형을 사용하지 않고 활용어를 사용한 (해요체 등) 상징들이 있으며, 사용자의 특성과 상황을 고려하여 상징의 어휘는 동일하지만 상징의 이미지가 다른 상징들이 있다. [그림 12]의 왼쪽 '넣어요' 상징은 일반적인 상황에서 사용할 수 있는 상징이지만, 오른쪽 '넣어요' 상징은 애완동물의 먹이를 통에 넣는 상황에서 사용할 수 있는 상징이다.

[그림 12] 특정 상황을 고려한 상징

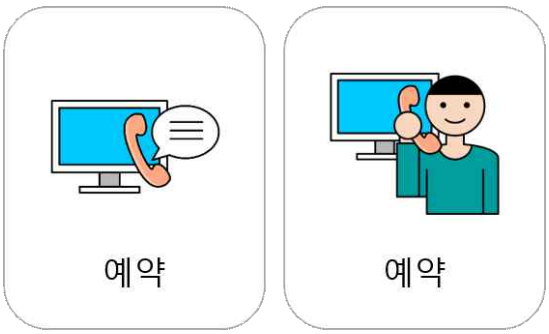
2. 한국형 AAC 그림 상징 시퀀스 데이터 구축

본 논문에서는 한국형 AAC 상징 체계집을 사용하여 한국어 대화 문장을 표현하는 한국형 AAC 그림 상징 시퀀스 데이터를 구축하였다. 한국형 AAC 상징 체계집에는 동일한 어휘를 가진 서로 다른 상징들이 있어, 이를 구분하기 위해 상징 식별자(ID)를 사용하였다. 한국어 대화 문장으로 AAC 사용자가 주로 사용하는 문장[21-24]과 AI Hub에서 제공하는 한국어 대화 데이터[6]를 사용하여 약 13,000개의 문장 데이터를 확보하였다. 확보한 문장 데이터 중 AAC 상징으로 표현할 수 없는 다소 복잡한 문장과 중복된 문장들을 제거하여 9,473개의 문장 데이터를 사용하였다. 한국어 문장에 대한 한국형 AAC 그림 상징 시퀀스를 다음과 같이 구축하였다.

- 한국형 AAC 그림 상징으로 동사를 표현할 때, 동사 원형의 어휘를 가진 상징을 사용한다. 만약 존재하지 않으면 활용형 어휘의 상징을 사용한다. 예를 들어 “줬다” 또는 “드리다”는 “주다”의 활용어이다. 하지만 한국형 AAC 상징 체계집에는 “주다” 어휘의 상징이 존재하지 않아, “주다”의 활용어인 [그림 13]의 ‘주세요’ 상징을 사용한다.

[그림 13] ‘주세요’ 상징

- AAC 구문은 동사가 생략된다는 특징을 고려하여 한국어 문장에서 동사를 표현할 때, 동사형 상징(활용형 포함)이 존재하지 않으면 명사형 상징을 사용한다. 즉, 상징의 어휘에 "-하다"를 붙여 동사로 표현이 가능하면 해당 상징을 사용한다. 예를 들어 [그림 14]의 '예약' 상징은 문장에서 "예약하다", "예약해요" 등에 사용된다.

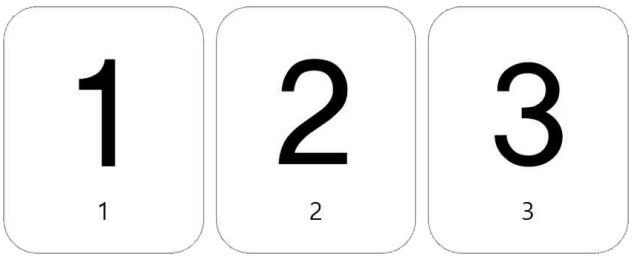


[그림 14] '예약' 상징

- 숫자 표현 중 화폐 단위는 한국형 AAC 상징 체계집의 화폐 단위 상징(예: 만원, 오천원. 천원 등)을 사용한다. 예를 들어 "15,000원" 또는 "만 오천원"은 [그림 15]처럼 '만원' 상징과 '오천원' 상징으로 표현한다.

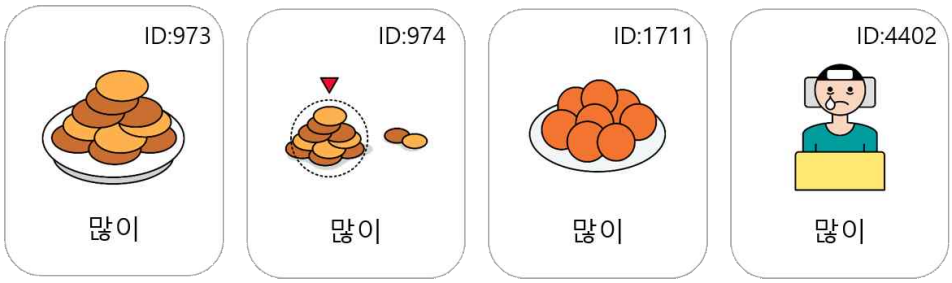
[그림 15] '만원'과 '오천원' 상징

- 화폐 단위 이외의 숫자 표현은 한국형 AAC 상징 체계집에 있는 숫자 상징을 사용한다. 숫자 상징으로 표현할 수 없는 경우 자릿수마다 0~9로 각각 표현한다. 예를 들어 숫자 “123”은 한국형 AAC 상징 체계집에 존재하지 않으므로, [그림 16]과 같이 숫자 상징 ‘1’, ‘2’, ‘3’으로 자릿수마다 따로 표현한다.



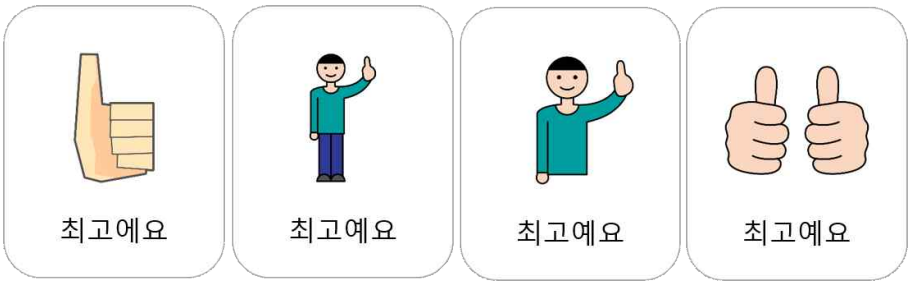
[그림 16] 숫자 123 표현

- 상징 어휘는 적합하지만 상징 이미지가 문맥에 맞지 않는 경우, 해당 문맥에 적합한 상징만 사용한다. 이때, 문맥에 적합한 상징은 상징 식별자(ID)를 사용하여 구분한다. 수의 정도를 의미하는 “많이”는 [그림 17]에서 상징 식별자(ID)가 973, 974, 1711인 ‘많이’ 상징을 사용하고, 아픈 정도의 “많이”는 [그림 17]에서 상징 식별자(ID)가 4402인 상징을 사용한다.



[그림 17] ‘많이’ 상징

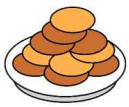
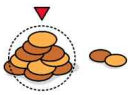
- 한국형 AAC 상징의 어휘로 모든 한국어 문장을 표현할 수 없으므로, 그림 상징에 따라 상황에 적합한 상징을 사용한다. 예를 들어 “제일”의 어휘를 가진 상징이 없으므로 ‘최고에요’ 또는 ‘최고예요’ 상징을 사용한다. ‘최고에요’와 ‘최고예요’ 상징은 [그림 18]과 같이 가장 으뜸인 것을 표현하는 상징으로, 상황에 따라 “제일”의 의미로 사용될 수 있다. 또 다른 예로, [그림 19]에서 보인 것처럼 ‘카카오페이’ 상징이 없으므로 ‘카카오톡’ 상징과 ‘계산해요’ 상징을 같이 사용하여 표현한다.



[그림 18] ‘최고에요’와 ‘최고예요’ 상징

[그림 19] “카카오페이”를 표현하는 상징

한국형 AAC 그림 상징 시퀀스 데이터는 2가지 방식, 상징 어휘 기반 시퀀스와 상징 식별자 기반 시퀀스로 표현할 수 있다. 예를 들어 문장 “비가 많이 내려 추워요.”를 2가지 방식으로 표현하면 [표 1]과 같다. 상징 어휘 기반 시퀀스는 [‘비’, ‘많이’, ‘내려요’, ‘추워요’] 1개이다. 하지만, 상징 식별자 기반 시퀀스는 [‘6681’,‘973’,‘6651’,‘497’]를 포함하여 총 9개의 시퀀스로 구축된다. 하나 이상의 시퀀스가 나오는 이유는 한국형 AAC 상징 체계집에 ‘많이’와 ‘추워요’ 어휘를 가진 상징이 하나 이상 존재하기 때문이다.

상징 어휘	비	많이	내려요	추워요
상징 식별자(ID)	ID:6681  비	ID:973  많이	ID:6651  내려요	ID:497  추워요
		ID:974  많이		ID:498  추워요
		ID:1711  많이		ID:499  추워요

[표 1] “비가 많이 내려 추워요”의 한국형 AAC 그림 상징 시퀀스

• 상징 식별자(ID) 기반 시퀀스

9,473개의 한국어 문장에 대한 상징 식별자 기반 시퀀스 데이터는 140,759개로 구축되었으나, 입력 데이터(상징 ID)의 개수와 예측 데이터(한국어 문장)의 수는 14.8배 차이로 예측 데이터의 수가 적다는 특징이 있다. 즉, 동일한 한국어 문장에 대해 많은 시퀀스들이 생성되었다. 예를 들어 [표 2]의 “좋은 생각이에요.”에서 “좋은”은 1개의 ‘좋아’ 상징과 2개의 ‘좋아요’ 상징으로 나타낼 수 있으며, “생각이에요.”는 ‘아이디어’ 상징으로 표현하여 총 3개의 시퀀스로 표현할 수 있다(그림 20 참조). 실제 데이터에서 “물 내려주세요.” 는 총 14개의 시퀀스로 표현되었다.

상징 식별자(ID)	한국어 문장
['181', '8379']	좋은 생각이에요.
['182', '8379']	좋은 생각이에요.
['183', '8379']	좋은 생각이에요.
['853', '3562', '188']	물 내려주세요.
...	...
['854', '3562', '186']	물 내려주세요.
...	...

[표 2] 상징 식별자 기반 시퀀스 데이터

[그림 20] 문장 “좋은 생각이에요.” 에 사용된 상징

- 상징 어휘 기반 시퀀스

9,473개의 한국어 문장에 대한 상징 어휘 기반 시퀀스 데이터는 총 10,484개로 구축되었다. 예를 들어 [표 3]의 “좋은 생각이에요.”에서 “좋은”은 ‘좋아’ 상징의 어휘와 ‘좋아요’ 상징의 어휘로 표현하며, “생각이에요.”는 ‘아이디어’ 상징의 어휘를 사용하여 총 2개의 시퀀스로 표현할 수 있으며, 한국어 문장 “물 내려주세요.”는 [‘물’,‘내려요’,‘주세요’] 1개로 표현된다.

상징 어휘	한국어 문장
[‘좋아’,‘아이디어’]	좋은 생각이에요.
[‘좋아요’,‘아이디어’]	좋은 생각이에요.
[‘물’,‘내려요’,‘주세요’]	물 내려주세요.
…	…

[표 3] 상징 어휘 기반 시퀀스 데이터

한국형 AAC 그림 상징은 AAC 사용자의 특징과 상황에 맞게 다른 어휘로 사용되기도 한다[18]. 예를 들어 [그림 21]과 같이 '아이디어' 상징을 사용자 어휘 '생각하다'로 사용할 수 있다. 즉, 동일한 상징 이미지이지만 사용자에 따라 '아이디어' 또는 '생각하다' 등 다른 어휘로 사용될 수 있다. 따라서 동일한 한국형 AAC 그림 상징 시퀀스라도 AAC 사용자에 따라 시퀀스의 의미가 달라질 수 있다.

[그림 21] '생각하다' 상징 제작

IV. 한국형 AAC 그림 상징 시퀀스의 딥러닝 기반 문장 생성

본 장에서는 학습의 용도에 맞는 토큰화(Tokenization) 방법, 자연어를 컴퓨터가 이해할 수 있게 표현하는 단어 임베딩(Word Embedding) 방법, GRU를 이용한 어텐션 기반 Seq2Seq 모델 그리고 모델 평가 방법 순서로 한국형 AAC 그림 상징 시퀀스를 한국어 문장으로 생성하는 과정에 대하여 살펴본다.

1. 토큰화(Tokenization)

토큰화는 텍스트 데이터를 사용할 때 용도에 맞게 처리하는 텍스트 전처리(Text preprocessing) 과정 중 하나로 주어진 코퍼스에서 토큰 단위로 나누는 작업을 말한다. 토큰은 학습성능에 영향을 미치므로 의미있는 단위로 정의해야 한다. 한국어는 교착어이기 때문에 의미를 지닌 어근을 학습시키기 위해서 보통 형태소 단위로 토큰화를 한다[25].

본 논문에서는 두 가지 토큰화 방법을 시도하여 한국형 AAC 그림 상징 시퀀스에 적합한 토큰화 방법을 찾아내고자 한다. 첫 번째 방법으로 AAC 상징이 하나의 언어처럼 사용된다는 특징을 고려하여 상징 어휘 혹은 상징 식별자(ID) 자체를 하나의 토큰으로 정의하는 상징 단위 토큰화 방법이다. AAC 상징의 어휘는 한국어이므로 두 번째 방법은 한국형 AAC 상징 시퀀스 전체를 하나의 문장으로 정의하고 이를 형태소 단위로 토큰화하는 방법이다. 한국어 대화 문장 데이터도 형태소 분석을 통해 토큰화를 진행한다.

예를 들어 한국어 문장이 “좋은 생각이에요.”, 상징 어휘 기반 시퀀스가 [‘좋아요’, ‘아이디어’], 상징 식별자 기반 시퀀스가 [′182′, ′8379′]인 경우 토큰화 결과는 다음과 같다. 상징 단위 토큰화 방법을 사용한 경우, 상징 어휘 기반 시퀀스는 ‘좋아요’, ‘아이디어’ 로 토큰화가 되며 상징 식별자 기반 시퀀스는 ‘182’, ‘8379’ 로 토큰화가 된다. 형태소 단위 토큰화 방법을 사용한 경우, 상징 어휘 기반 시퀀스는 ‘좋’, ‘아요’, ‘아이디어’로 토큰화가 되며 한국어 문장은 ‘좋’, ‘은’, ‘생각’, ‘이’, ‘에요’ 로 토큰화가 된다.

형태소 분석은 한국어 정보처리를 위한 파이썬 패키지 KoNLPY[26]를 사용한다. KoNLPY는 총 5개의 형태소 분석기(Hannanum, Kkma, Komoran, Mecab, Twitter)로 이루어져 있다. 본 논문에서는 띄어쓰기와 실행 시간에 더 높은 성능을 보인 Mecab을 사용하여 형태소 분석을 진행한다.

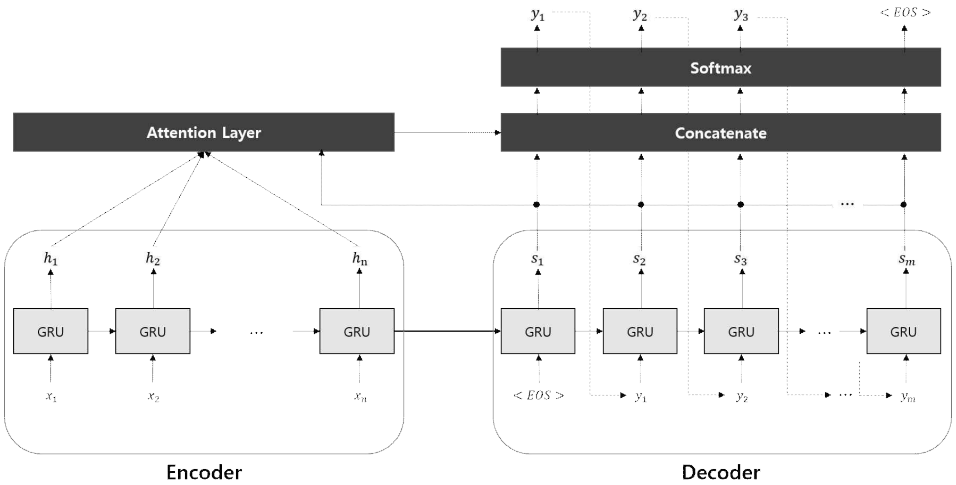
2. 단어 임베딩(Word Embedding)

토큰화한 결과를 컴퓨터가 효율적으로 처리할 수 있도록 자연어를 숫자로 변환하는 단어 임베딩(Word Embedding) 과정을 거쳐야 한다[26]. 단어 임베딩은 단어를 밀집 벡터의 형태로 표현하는 방법을 말하며, 이 밀집 벡터를 임베딩 벡터(Embedding vector)라고 한다. 단어 임베딩 방법은 자연어 처리의 성능에 영향을 미치기 때문에 FastText[27], Word2Vec[28], Glove[29] 등 많은 방법론이 제안되었다. 또한, 대량의 코퍼스를 가지고 Word2vec, FastText, Glove등을 통해 미리 훈련된 임베딩 벡터들을 사용함으로써 훈련 데이터가 적은 경우에 성능 개선을 기대할 수 있다.

본 논문에서는 토큰화 방법에 따라 다른 단어 임베딩 방법을 사용하고자 한다.

- 상징 단위 토큰화는 AAC 상징을 하나의 언어처럼 사용한 것이므로, 각 토큰에 고유한 정수값을 부여하고 이를 밀집 벡터로 변환하여 인공 신경망을 통해 단어 벡터를 학습하는 방법을 사용한다.
- 형태소 단위 토큰화는 한국어의 언어적 특징을 반영한 것으로, 한국어에 좋은 성능을 보인 Fasttext[27]로 사전 훈련된 Fasttext 임베딩 벡터를 사용한다.

3. GRU를 이용한 어텐션 기반 Seq2Seq 모델



[그림 22] GRU를 이용한 어텐션 기반 Seq2Seq 모델 구조

본 논문에서는 [그림 22]와 같이 어텐션 메커니즘을 적용한 Seq2Seq 모델을 사용하여 한국형 AAC 그림 상징 시퀀스를 한국어 문장으로 생성한다. Seq2Seq 모델은 인코더-디코더 구조로 이루어져 있으며 인코더에는 토큰화한 한국형 AAC 그림 상징 시퀀스, 디코더에는 한국어 문장을 순차적으로 입력한다. 인코더의 모든 은닉 상태와 해당 시점의 디코더 은닉 상태를 Attention Layer에 전달한다. Attention Layer에서는 예측해야 할 단어와 연관이 있는 부분에 얼마나 더 집중할지 계산하고, 인코더의 문맥을 포함한 문맥 벡터를 구한다. Attention Layer에서 어텐션 점수는 Bahdanau[17]가 제안한 방법으로 계산하였다. 그다음 문맥 벡터와 임베딩된 디코더의 단어 벡터를 연결(Concatenate)하여 하나의 벡터로 만들고, 그 값을 해당 시점의 새로운 입력으로 사용하여 Softmax를 통해 출력 시퀀스를 예측한다.

V. 실험 및 평가

본 장에서는 실험을 진행한 환경, 실험 과정과 각 과정의 결과를 확인하고 모델의 성능을 평가하였다. 전체적인 실험은 상징 어휘 기반 시퀀스를 사용하여 진행하며, 마지막으로 상징 식별자 기반 시퀀스를 사용한 학습결과를 고찰하였다.

1. 실험 환경

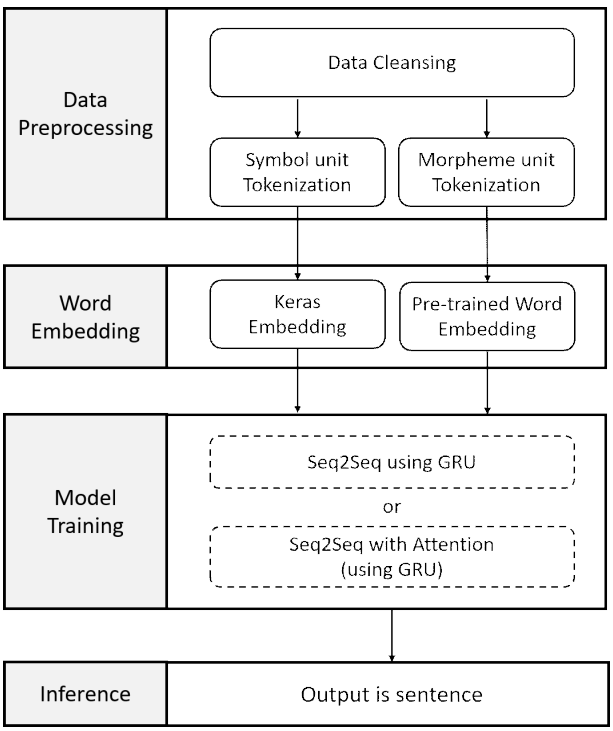
모델 구현 시 설정된 실험 환경은 다음 [표 4]와 같다.

	구분	버전
H/W	CPU	Intel(R) Core(TM) i9-9900K CPU @ 3.60GHz
	GPU	NVIDIA TITAN Xp 16G
	RAM	32 GB
S/W	OS	Windows 10
	Python	3.6.8
	TensorFlow	2.0.0

[표 4] 실험 환경

2. 실험 과정 및 결과

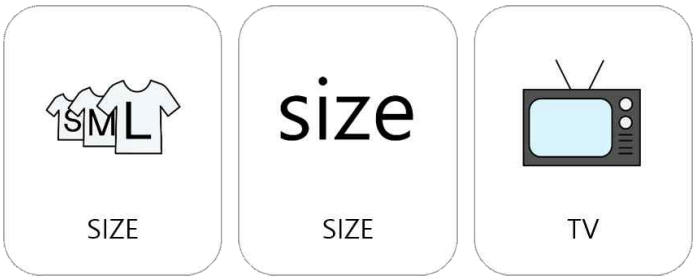
실험은 [그림 23]과 같이 데이터 전처리(Data Preprocessing), 단어 임베딩(Word Embedding), 모델 훈련(Model Training), 추론(Inference) 순서로 진행한다. 데이터 전처리에서는 텍스트 데이터를 사용하기 위한 작업으로 정제(Cleansing), 토큰화(Tokenization), 불용어 처리를 진행한다. 토큰화는 상징 단위 토큰화 방법과 형태소 단위 토큰화 방법을 사용하며, 각 방법에 따라 다른 단어 임베딩을 적용한다. 모델 학습은 GRU를 사용한 Seq2Seq 모델과 어텐션 메커니즘을 적용한 모델을 실험한다. 전체적인 실험 과정에서는 상징 어휘 기반 시퀀스를 사용하며, 추가로 상징 식별자 기반 시퀀스를 사용한 실험을 진행하여 학습결과를 확인한다.



[그림 23] 실험 과정

1) 데이터 전처리(Data Preprocessing)

텍스트 처리에 불필요한 문자와 기호를 제거하고 영어를 한국어로 바꿔주는 정제(Data Cleansing) 과정을 거친다. 단어 임베딩 과정에서 사전 훈련된 한국어 임베딩 벡터들을 사용하기 때문에 영어로 된 상징 어휘들을 한국어로 바꿔준다. 예를 들어 [그림 24]의 'TV'는 '텔리비전'으로 'size'는 '크기'로 바꿔준다.



[그림 24] 영어 어휘 상징

그다음 상징 어휘 기반 시퀀스(Source data)를 2가지 방식으로 토큰화하여 실험하도록 한다. [표 5]는 토큰화를 하기 전 데이터로, 문장 "제일 맛있는 반찬이 뭐예요?"의 상징 어휘 기반 시퀀스는 '최고예요', '맛있어요', '반찬', '뭐예요' 이다. 3장에서 설명한 것처럼 "제일"은 '최고예요' 상징으로 표현하였다.

한국형 AAC 그림 상징 시퀀스	
상징 어휘 기반 시퀀스 (Source data)	[‘최고예요’, ‘맛있어요’, ‘반찬’, ‘뭐예요’]
한국어 문장 (Target data)	제일 맛있는 반찬이 뭐예요?

[표 5] 토큰화 전 데이터

[표 6]은 상징 어휘 기반 시퀀스를 상징 단위로 토큰화하고, 한국어 문장은 KoNLPY[26]에서 제공하는 Mecab을 사용하여 형태소 단위로 토큰화한 결과이다. 상징 단위 토큰화 방법은 상징의 어휘가 문장형인 경우에도 하나의 토큰으로 정의한다. 예를 들어 ‘최고예요’, ‘맛있어요’, ‘뭐예요’ 상징은 문장형이지만 하나의 토큰으로 정의한다.

상징 어휘 기반 시퀀스 (Source data)	최고예요 / 맛있어요 / 반찬 / 뭐예요
한국어 문장 (Target data)	제일 / 맛있 / 는 / 반찬 / 이 / 뭐 / 예요 / ?

[표 6] 상징 단위 토큰화

[표 7]은 상징 어휘 기반 시퀀스를 하나의 문장으로 정의하고, Mecab을 사용해 형태소 단위로 토큰화한 결과이다. 문장형 어휘를 가진 한국형 AAC 상징의 경우 형태소 단위로 토큰화가 적용되며, 불용어 처리 과정에서 토큰의 품사가 조사와 어미인 경우 제거된다. 예를 들어 '맛있어요' 상징은 형태소 단위인 '맛있'과 '어요' 로 토큰화가 적용되며, 품사가 어미인 '어요'는 제거된다.

상징 어휘 기반 시퀀스 (Source data)	최고 / 예요 / 맛있 / 어요 / 반찬 / 뭐 / 예요
한국어 문장 (Target data)	제일 / 맛있 / 는 / 반찬 / 이 / 뭐 / 예요 / ?

[표 7] 형태소 단위 토큰화

2) 단어 임베딩 (Word Embedding)

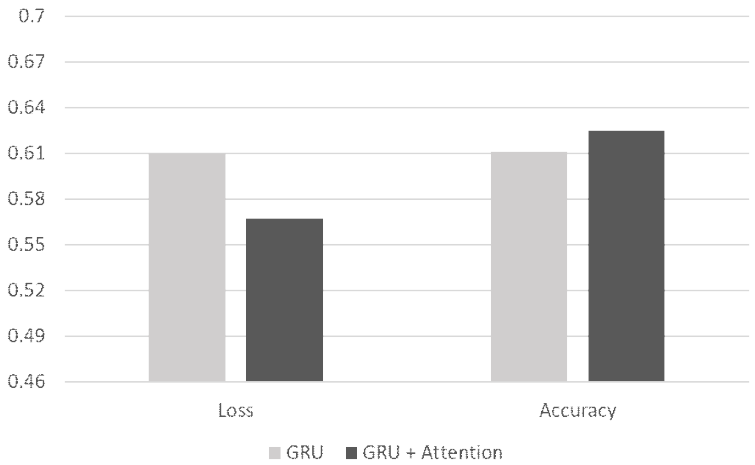
토큰화 과정을 마친 후, 토큰화 방법에 따라 다른 단어 임베딩(Word Embedding)을 진행한다. 상징 단위로 토큰화한 경우 Keras의 Embedding을 사용하여 단어를 밀집 벡터로 만들고 신경망을 통해 단어 벡터를 학습하는 방법을 사용한다. 임베딩 차원은 [25]의 임베딩 차원의 수에 따른 성능 비교 실험에서 높은 성능을 보인 100차원으로 정의한다. 형태소 단위로 토큰화한 상징 어휘 기반 시퀀스와 한국어 문장은 사전에 300차원으로 훈련된 한국어 Fasttext 임베딩 벡터를 적용한다.

3) 모델 학습 (Model Training)

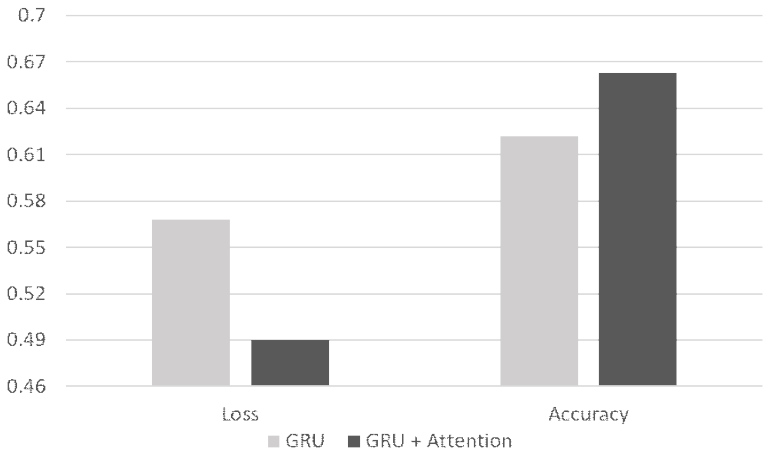
본 논문에서는 상징 어휘 기반 시퀀스 데이터 10,484개를 무작위로 섞은 후 80%는 학습 데이터, 10%는 검증 데이터, 10%는 테스트 데이터로 사용하였으며, GRU를 사용한 Seq2Seq 모델과 어텐션 메커니즘을 적용한 모델의 학습을 진행하였다. 모델 학습은 배치(Batch) 크기는 16, 최적화(Optimizer) 알고리즘은 Adam, 학습률은 0.001로 설정하였다. 에폭(Epoch)은 100으로 설정하였지만, Keras에서 제공하는 학습 조기 종료(Early Stopping) 기능을 사용하여 과적합이 발생하면 중간에 종료해 과적합(Overfitting)을 방지하였다.

[그림 25]는 상징 어휘 기반 시퀀스를 사용한 상징 단위 토큰화 방법에 따른 학습결과로, 어텐션 메커니즘을 적용한 모델이 GRU만 사용한 모델보다 손실(Loss)은 7.58% 줄었으며 정확도(Accuracy)는 2.29% 향상되었다. [그림 26]은 형태소 단위 토큰화 방법에 따른 학습결과로, 어텐션

메커니즘을 적용한 모델이 GRU만 사용한 모델보다 손실은 15.92% 줄었으며 정확도는 6.59% 향상되었다. 즉, GRU를 사용한 어텐션 기반 Seq2Seq 모델이 더 높은 성능을 보인 것을 확인할 수 있다.



[그림 25] 상징 단위 토큰화를 사용한 모델의 학습결과



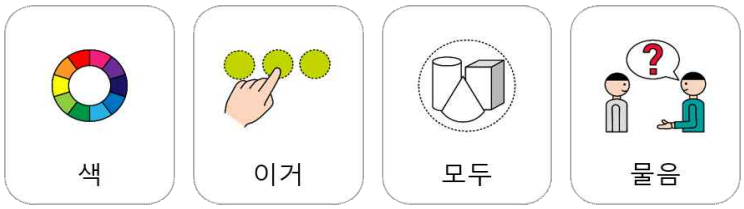
[그림 26] 형태소 단위 토큰화를 사용한 모델의 학습결과

4) 추론 (Inference)

토큰화 방법에 따른 GRU를 사용한 Seq2Seq 모델과 어텐션 메커니즘을 적용한 모델의 추론결과는 다음과 같다.

- 상징 단위 토큰화를 사용한 모델별 추론

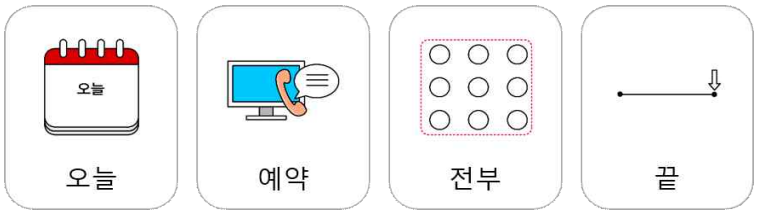
[표 8]은 상징 단위 토큰화를 사용한 모델의 추론결과이다. GRU를 사용한 Seq2Seq 모델과 어텐션 메커니즘을 적용한 모델의 추론결과를 살펴보면 문장의 의미는 파악할 수 있지만, 어텐션 메커니즘을 적용한 모델의 결과가 더 자연스러운 문장인 것을 확인할 수 있다. [표 8]의 ②의 경우 GRU를 사용한 Seq2Seq 모델의 추론결과인 “사이즈 는 ?”은 완벽한 문장 형태를 이루지 않았으며, 어텐션 메커니즘을 적용한 모델의 추론결과인 “사이즈 는 어떻게 되 나요 ?” 는 상징 시퀀스의 의미에 적합한 문장인 것을 확인하였다. [표 8]의 ③의 경우 GRU를 사용한 Seq2Seq 모델의 추론결과는 문장형이지만 ‘모두’ 상징의 의미가 “하나”로 생성되어 적합하지 않으나, 어텐션 메커니즘을 적용한 모델은 ‘모두’ 상징의 의미와 비슷한 단어인 “전부”로 생성된 것을 확인하였다.

①	
AAC 그림 상징 시퀀스	
GRU	이게 제일 많이 나가 요
GRU + Attention	이게 제일 많이 나가 요
②	
AAC 그림 상징 시퀀스	
GRU	사이즈 는 ?
GRU + Attention	사이즈 는 어떻게 되 나요 ?
③	
AAC 그림 상징 시퀀스	
GRU	색상 은 이거 하나 예요 ?
GRU + Attention	색상 은 이거 전부 예요 ?

[표 8] 상징 단위 토큰화를 사용한 모델별 추론결과

- 형태소 단위 토큰화를 사용한 모델별 추론

[표 9]는 형태소 단위 토큰화를 사용한 모델의 추론결과이다. [표 9]의 ②의 경우 GRU를 사용한 모델의 추론결과는 “사이즈 로 큰 거 주 세요” 로, 문법적으로 어색한 문장으로 생성되었다. 하지만 어텐션 메커니즘을 적용한 모델의 결과는 “큰 사이즈 로 주 세요” 로, ‘커요’ 상징과 ‘크기’ 상징이 문장에서 ‘큰 사이즈’로 생성된 것을 확인하였다. 또한 [표 9]의 ③에서 GRU를 사용한 모델의 추론결과는 시작 단어를 제외하고 시퀀스에 맞지 않은 문장으로 생성되었으며, 어텐션 메커니즘을 적용한 모델의 추론결과는 상징 시퀀스의 의미에 적합한 한국어 문장인 “오늘 예약 은 다 끝 났어요” 로 생성된 것을 확인하였다.

①	
AAC 그림 상징 시퀀스	
GRU	이거 는 겨울옷 인가요 ?
GRU + Attention	이거 는 겨울옷 인가요 ?
②	
AAC 그림 상징 시퀀스	
GRU	사이즈 로 큰 거 주 세요
GRU + Attention	큰 사이즈 로 주 세요
③	
AAC 그림 상징 시퀀스	
GRU	오늘 은 행사 다 되 셨습니다
GRU + Attention	오늘 예약 은 다 끝 났어요

[표 9] 형태소 단위 토큰화를 사용한 모델별 추론결과

5) 상징 식별자 기반 시퀀스를 사용한 실험

추가 실험으로 상징 식별자 기반 시퀀스를 사용하여 어텐션을 적용한 모델을 학습하였다. 토큰화는 상징 단위 토큰화 방법, 단어 임베딩은 Keras Embedding을 사용하여 단어 벡터를 학습하는 방법을 적용하였다. [표 10]은 학습결과로 훈련 손실(Loss)은 0.0068 검증 손실(Val_loss)은 0.0187이며, 훈련 정확도(Accuracy)는 0.9937 검증 정확도(Val_Accuracy)는 0.987로, 상징 어휘 기반 시퀀스를 사용한 모델에 비해 매우 높은 성능을 보였다.

Loss	Accuracy	Val_loss	Val_Accuracy
0.0068	0.9937	0.0187	0.987

[표 10] 상징 식별자 기반 시퀀스 데이터 학습결과

높은 성능을 보인 이유는 다음과 같이 2가지로 분석하였다.

① 상징 식별자 기반 시퀀스 데이터의 입력 데이터 개수와 예측 데이터 개수는 약 14.8배 차이로, 예측 데이터의 수가 적어 정답을 예측할 확률이 높아져 우수한 성능을 보인 것으로 판단하였다.

② 데이터를 무작위로 섞는 작업에서 빈도수가 많은 데이터가 학습과 평가 데이터에 균등하게 분포되어있지 않아, 높은 정확도에 영향을 준 것으로 파악하였다.

위와 같은 이유로 상징 식별자 기반 시퀀스는 추후 서로 다른 다수의 한국어 문장이 존재하는 데이터가 추가되었을 때도 높은 성능을 보일지 의문이 있어, 본 논문에서는 상징 어휘 기반 시퀀스를 사용하여 실험을 진행하였다.

3. 평가

1) 평가 방법

BLEU 점수(Bilingual Evaluation Understudy Score)를 사용하여 모델의 성능을 평가한다. BLEU 점수는 번역에 대한 성능을 측정하는 방법 중 하나이며, 번역된 문장(Candidate)과 정답에 해당하는 실제 문장(Reference)들이 얼마나 유사한지 비교하는 방법을 사용한다[30]. 본 논문에서 실제 문장(Reference)들을 구성하기 위해 AAC 그림 상징 시퀀스 데이터 구축에 사용된 한국어 문장들을 사용하였다.

BLEU 점수는 단어의 순서를 고려하는 n-gram 정밀도를 기반으로 계산되며, 짧은 문장의 경우 점수가 높게 측정될 수 있어 짧은 문장 길이에 대한 패널티(Brevity Penalty; BP)를 적용한다. 문장 길이에 대한 패널티를 적용하는 방법은 수식 (9)이며, BP를 적용한 최종 BLEU 점수를 계산하는 방법은 수식 (10)이다.

$$BP = \begin{cases} 1 & \text{if } c > r \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases} \tag{9}$$

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N} w_n \log\ p_n\right) \tag{10}$$

2) 평가 결과

모델 \ 토큰화 방법	상징 단위 토큰화	형태소 단위 토큰화
GRU	0.247	0.287
GRU+Attention	**0.27**	**0.30**

[표 11] 토큰화 방법에 따른 BLEU 점수

[표 11]은 상징 어휘 기반 시퀀스를 사용하여 각 토큰화 방법과 모델에 따른 BLEU 평가 결과이다. 상징 단위 토큰화 방법에서 GRU를 사용한 모델의 BLEU 점수는 0.247이며 어텐션 메커니즘을 적용한 모델의 BLEU 점수는 0.27로 9.31% 향상되었다. 형태소 단위 토큰화 방법에 따른 방법에서 GRU를 사용한 모델의 BLEU 점수는 0.287이며 어텐션 메커니즘을 적용한 모델의 BLEU 점수는 0.30로 4.53% 향상되었다. 추가 실험으로 상징 식별자 기반 시퀀스를 사용한 어텐션 기반 Seq2Seq 모델의 BLEU 점수는 0.888로 가장 높은 점수를 얻었지만, 앞서 설명한 데이터 특징과 데이터를 섞는 과정에서의 제약으로 상징 식별자 기반 시퀀스는 사용하지 않는다.

평가 결과를 통해 상징 어휘 기반 시퀀스는 형태소 단위 토큰화 방법이 더 적합하다는 것을 알 수 있었으며, GRU를 사용한 어텐션 기반 Seq2Seq 모델이 BLEU 점수 0.30으로 가장 높은 성능을 보인 것을 확인하였다. 구글에서 제공하는 BLEU 점수 해석[31]에 의하면, 0.20~0.29는 요점은 명확하지만 문법적 오류가 있음을 의미하고 0.30~0.40은 이해할

수 있는 양호한 번역을 의미한다. GRU를 사용한 어텐션 기반 Seq2Seq 모델의 BLEU 점수 0.3은 사람이 이해할 수 있는 양호한 번역 결과임을 알 수 있었으며, 이외의 실험한 모델들은 BLEU 점수는 0.2 이상 0.29 이하로 요점은 명확하지만 다소 문법적 오류가 있다는 번역 결과임을 알 수 있었다.

VI. 결론 및 향후 연구

온라인 공간과 같은 비대면 의사소통 상황에서 대부분 일반인은 AAC에 대한 이해도가 높지 않아, 언어장애인이 사용하는 AAC 그림 상징만으로 상대방의 의사를 이해하는 데 어려움이 있다.

본 논문에서는 언어장애인과 일반인의 원활한 비대면 의사소통을 위해 AAC 그림 상징들을 이용한 의사 표현을 일반인이 이해하는 텍스트 문장으로 변환하는 딥러닝 모델을 제안하였다. 딥러닝 모델의 학습을 위한 데이터로 한국형 AAC 그림 상징 시퀀스 데이터를 구축하였으며, 상징 어휘 기반 시퀀스와 상징 식별자 기반 시퀀스의 2가지 형태로 구축하였다. 실험 모델로 GRU를 사용한 Seq2Seq 모델과 어텐션 메커니즘을 적용한 모델을 활용하였으며 토큰화에 따른 모델별 성능을 비교하는 실험을 진행하였다.

상징 어휘 기반 시퀀스 실험에서 AAC 상징의 특징을 고려한 토큰화 방법을 알아보기 위해 상징 단위 토큰화 방법과 형태소 단위 토큰화 방법을 시도하였으며, 두 모델의 성능을 비교하였다. 결과적으로 GRU를 사용한 어텐션 기반 Seq2Seq 모델에서 더 좋은 성능을 보였으며, 추론결과와 BLEU 점수 그리고 사용자에 따라 상징의 어휘가 다르다는 특징을 고려하여 형태소 단위 토큰화 방법이 적합하다는 것을 확인하였다.

상징 식별자 기반 시퀀스를 사용한 모델의 실험 결과는 상징 어휘 기반 시퀀스를 사용한 모델에 비해 BLEU 점수에서 매우 높은 성능을 보였다. 하지만 상징 식별자 기반 시퀀스는 입력 데이터의 수와 예측 데이

터의 수가 크게 차이 나기 때문에 정답확률이 높아진 것으로, 하나의 상징 식별자 기반 시퀀스에 대하여 서로 다른 다수의 한국어 문장이 존재하는 데이터에서도 이와 같은 성능을 보일지는 의문이 있다. 따라서 향후 많은 데이터를 확보한 후 다시 실험해볼 가치가 있으며, 데이터 빈도수를 고려한 셔플링 방법을 적용하면 처음 본 데이터에 대해서도 잘 작동하는 일반화된 모델을 얻을 수 있을 것이다.

본 논문은 한국 AAC 분야에서 AAC 상징을 사용한 문장생성 연구의 기초를 제공할 것으로 기대한다. 또한, 실험 모델의 BLEU 점수 결과는 사람이 이해할 수 있는 정도의 성능을 보였으며, AAC 어플리케이션뿐만 아니라 SNS와 같은 다양한 온라인 플랫폼에 적용한다면 AAC 사용자와 비장애인의 비대면 상황에서 더 원활한 의사소통이 가능할 것으로 기대한다.

본 논문에서 한국형 AAC 그림 상징 시퀀스 데이터 구축을 위하여 사용한 한국형 AAC 기본 상징 체계집은 약 만 개의 상징으로, 일상대화에 필요한 어휘를 전부 나타낼 수 없다는 한계점이 있었다.

- 시간 단위(예: 시, 분, 초), 사람 단위(예: 명, 인분) 등 의사소통에 자주 사용되는 단위 어휘들이 부족하다.
- 일상적인 대화에서 활용형 상징 어휘를 응용하는 데 어려움이 있다. 예를 들어 한국어 문장인 “날짜를 보세요”에서 “보세요”를 표현할 때 가장 적합한 상징은 ‘보여주세요’ 상징으로 “보세요”와 의미적인 차이가 있다.
- 일반적으로 사용할 수 있는 상징의 어휘가 특정한 상황에서만 사용할 수 있는 상징 이미지로 표현되어있다. 예를 들어 ‘알겠어요’ 상징

의 어휘는 일반적으로 사용될 수 있는 어휘지만, 상징의 이미지는 요리사 모자를 쓴 사람으로 특정 상황(예: 식당, 빵집 등)에서만 사용된다.

- 한국형 AAC 상징 체계집의 한계점을 보완하기 위하여 임의로 대체 어휘 및 상징을 합성하여 사용하였다.

이상의 약점을 보완한 한국형 AAC 상징 체계집의 보완 구축이 필요하다.

또한, 일반인이 사용한 한국어 문장 데이터를 바탕으로 데이터를 구축하였기 때문에 다음과 같은 한계점이 있다.

- 일반인의 대화 데이터에는 긴 문장들이 다수 있어, 한국형 AAC 그림 상징 시퀀스 데이터에서 상징 5개 이상으로 이루어진 것이 약 30%를 차지한다. 즉, 장애인 사용자가 표현하기에는 긴 시퀀스 데이터로 구성되어 있다.
- 한국형 AAC 그림 상징 시퀀스 데이터는 단일 대화체 문장으로, 이전 대화를 고려하지 않는다.

한국형 AAC 상징 체계집의 보완 구축과 더불어 필요한 연구는 의사소통 장애인들이 자주 사용하는 문장들을 바탕으로 다양한 그림 상징 시퀀스 데이터를 구축하고 AAC 전문가의 검증을 받는 것이 필요하다. 더 나아가 사용자 데이터(예: 사용자의 현재 위치, 감정 등)와 이전 대화 데이터를 활용하여, 문장생성 연구를 진행한다면 대화 문맥(예: 높임말, 시제 등)을 고려한 자연스러운 문장을 생성할 수 있을 것이다. 이와 더불어 BERT, GPT-3 등 최신 언어 모델을 사용한 연구를 진행한다면 더 높은 성과를 기대할 수 있을 것이다.

참고문헌

[1] K. Y. Kim and E. H. Park, “A study on family involved intervention factors in augmentative and alternative communication(AAC),” S*pec. Educ. Res.*, vol. 10, no. 3, pp. 219, 2011.

[2] E. Park, Y. Kim, K.-H. Hong, S. Yeon, K. Kim, and J. Lim, “Development of Korean ewha-AAC symbols: Validity of vocabulary and graphic symbols,” *AAC Res. Pract.*, vol. 4, no. 2, pp. 19, 2016.

[3] 남기화, 엄숙희, 이은영, 이정은, 조혜림, 정은지, 최순지. “언택트로 온-텍트하다,” *한국보완대체의사소통 추계학술대회 발표집*, pp. 191-229, 2020.

[4] H.-J. Yun and H.-J. Park, “Use of mobile messengers through connection of AAC with kakao talk,” *AAC Res. Pract.*, vol. 3, no. 2, pp. 167, 2015.

[5] S. Yeon, Y. Kim, and E. H. Park, “Transparency and name agreement of Korean ewha-AAC symbols – nouns, verbs, and adjectives –,” *AAC Res. Pract.*, vol. 4, no. 1, pp. 45, 2016.

[6] 한국정보화진흥원, Aihub.or.kr. [Online]. Available: https://www.aihub.or.kr/. [Accessed: 15-Nov-2020].

[7] Y. T. Kim, “Using AAC for children with speech-language disorders,” *AAC Res. Pract.*, vol. 2, no. 1, 2014.

[8] E.-J. Hwang and H.-K. Min, “A method of sentence generation for augmentative and alternative communication,” *KIPS Trans.*

PartB, vol. 12B, no. 3, pp. 323 - 328, 2005.

[9] J. Ryu and K.-R. Han, "Implementation of augmentative and alternative communication system using image dictionary and verbal based sentence generation rule," *KIPS Trans. PartB*, vol. 13B, no. 5, pp. 569 - 578, 2006.

[10] K.-N. Choo, Y.-S. Woo, and H.-K. Min, "Implementation of iconic language for the language support system of the language disorders," *KIPS Trans. PartB*, vol. 13B, no. 4, pp. 479 - 488, 2006.

[11] N. Tintarev, E. Reiter, R. Black, and A. Waller, "Natural language generation for augmentative and assistive technologies," in *Natural Language Generation in Interactive Systems*, A. Stent and S. Bangalore, Eds. Cambridge: Cambridge University Press, 2014, pp. 252 - 278.

[12] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans.* Neural Netw., vol. 5, no. 2, pp. 157 - 166, 1994.

[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735 - 1780, 1997.

[14] K. Cho et al., "Learning phrase representations using RNN encoder - decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[15] 사이토 고키. "밑바닥부터 시작하는 딥러닝 2: 파이썬으로 직접 구현하며 배우는 순환 신경망과 자연어 처리," 한빛미디어, 서울. 2019.

[16] 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.

[17] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv [cs.CL]*, 2014.

[18] Y. Jang and K.-H. Hong, "An HTML5 based AAC board making system," *KIISE Trans. Comput. Pract.*, vol. 21, no. 5, pp. 365 - 372, 2015.

[19] H. Cho and K.-H. Hong, "GeoAAC, A location-based augmentative and alternative communication mobile application," *AAC Res. Pract.*, vol. 8, no. 1, pp. 87 - 117, 2020.

[20] "마이토키," M ytalkie.co.kr. [Online]. Available: http://www.mytalkie.co.kr/. [Accessed: 15-Nov-2020].

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

[22] 천춘경. "보완·대체 의사소통 (AAC) 체계 활용을 위한 지역사회 중심의 기초어휘 및 문장조사," 국내석사학위논문 단국대학교 대학원, 2000.

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

[24] 이정은, 박은혜. "보완· 대체의사소통체계 적용을 위한 상황 중심 핵심어휘 개발 연구," *재활복지*, vol. 4, pp. 96- 122, 2000

[25] Choi, Sanghyuk, Jinseok Seol, and Sang-goo Lee. "On word embedding models and parameters optimized for korean," *한국어정보학회: 학술대회논문집*, pp. 252 - 256, 2016.

[26] 박은정, 조성준. "KoNLPy: 쉽고 간결한 한국어 정보처리 파이썬 패키지," *제 26 회 한글 및 한국어 정보처리 학술대회 논문집*, pp. 1-4, 2014.

[27] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. Assoc. Comput. Linguist.*, vol. 5, pp. 135 - 146, 2017.

[28] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv [cs.CL]*, 2013.

[29] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

[30] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics - ACL '02*, 2001.

[31] "Evaluating models," Google.com. [Online]. Available: https://cloud.google.com/translate/automl/docs/evaluate. [Accessed: 29-Dec-2020].

ABSTRACT

Sentence Generation for Korean AAC Symbol Sequences based on deep learning

Cho Hee
Department of Computer Science
Graduate School of
Sungshin University

Augmentative and Alternative Communication(AAC) is a method of communication using non-verbal methods such as images and gestures. People with language impairment use the AAC symbols to express their intentions. As smart devices and mobile applications become popular, the number of AAC applications has been increasing. Current AAC applications are effective in face-to-face communication. However, they are not suitable for non-face-to-face communication on various social network services because ordinary people have a low understanding of AAC.

The purpose of this study is to generate the Korean sentence from the Korean AAC symbol image sequences based on deep learning so that Korean AAC users can communicate easily with ordinary people on various mobile platforms. For training data, we constructed the Korean AAC symbol sequences. The Korean AAC symbol sequences

can be expressed in the sequences based on symbol identifier(id) or symbol vocabulary. We tokenized the AAC sequences by using morpheme and AAC symbol. Then, using the tokenized AAC sequences, we conducted and compared two deep learning models, the Sequence-to-Sequence model using GRU and the Sequence-to-Sequence model with attention. As a result, we found that the Sequence-to-Sequence model with attention using the morpheme unit tokenization was the best for translating the AAC symbol sequence to the Korean sentences.