



국민대학교
소프트웨어융합대학
소프트웨어학부

캡스톤 디자인 I

종합설계 프로젝트

프로젝트 명	AID(AI Doctor)
팀 명	캡스톤 디자인 1 12팀
문서 제목	중간보고서

Version	1.0
Date	2022-03-31

팀원	황교민 (조장)
	장민혁
	조상연
	허진우
지도교수	-

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

CONFIDENTIALITY/SECURITY WARNING

이 문서에 포함되어 있는 정보는 국민대학교 소프트웨어융합대학 소프트웨어학부 및 소프트웨어학부 개설 교과목 캡스톤 디자인I 수강 학생 중 프로젝트 "AID"를 수행하는 팀 "캡스톤 디자인1 12팀"의 팀원들의 자산입니다. 국민대학교 소프트웨어학부 및 팀 " 캡스톤 디자인1 12팀 "의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

문서 정보 / 수정 내역


Filename	중간보고서-AID.doc
원안작성자	장민혁, 조상연, 허진우, 황교민
수정작업자	장민혁, 조상연, 허진우, 황교민

수정날짜	대표수정자	Revision	추가/수정 항목	내 용
2022-03-31	황교민	1.0	최초 작성	

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

목 차

1	프로젝트 목표	4
2	수행 내용 및 중간결과	6
2.1	배경 지식	6
2.1.1	Transformer(Attention is all you need).....	6
2.1.2	GPT(Generative Pre-Training).....	10
2.1.3	BERT(Bidirectional Encoder Representations from Transformers).....	15
2.2	계획서 상의 연구 내용	19
2.3	수행내용	21
3	수정된 연구내용 및 추진 방향	23
3.1	수정사항	23
4	향후 추진계획	24
4.1	향후 계획의 세부 내용	24
5	고충 및 건의사항	24
6	참고 문헌	24

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

1 프로젝트 목표


우울증은 생각의 내용, 사고 과정, 동기, 의욕, 관심, 행동, 수면, 신체 활동 등 전반적인 정신 기능이 지속적으로 저하되어 일상생활에도 악영향을 미치는 상태를 의미합니다. 우울증은 단순한 정신질환으로 생각하기 쉽지만 신체적으로도 두통이나 위장질환 등을 유발할 수도 있는, 정신적인 고통만이 존재하는 질병은 아닙니다. 코로나 발생 이후 세계 각국에서 우울증과 불안증의 발생이 2 배 이상 증가했습니다. 특히 한국에서는 우울증 유병률이 36.8%로 발표되었습니다.

심리부검 결과에 따르면 사람들이 자살을 택한 주요 원인 중 32.2%는 우울증이었습니다. 자살의 결정적 요인이 아니더라도 88.4%는 정신 질환을 앓고 있었던 것으로 드러났습니다.

연령별 우울증 점유율을 살펴보면 2016 년 기준 60 대 이상 노인이 43.8%를 차지하며, 독거 노인의 경우 30.2%이 우울하다고 답했습니다. 이는 노인 부부(16.4%)에 비해 2 배 이상 높은 수치입니다. 특히, 한국의 노인 자살률은 10 년 넘게 OECD 국가 중 압도적인 1 위를 차지하고 있습니다.

국가에서도 이를 해결하기 위한 정책들을 시행하고 있습니다. 특히 독거 노인 방문 돌봄 서비스나 기초 심리 검사에 근거한 유형별 상담 서비스를 지원하며 노인의 경제적, 신체적 정신적 문제를 해결하고자 합니다. 하지만 현재 정책에는 두가지 한계점이 있습니다. 첫째로, 우울증 환자는 자신이 우울증인 것을 알지 못하고 일상 생활에서 상당히 위축되어 기능이 떨어질 때까지도 자신의 기분 문제에 대해 호소하지 않을 뿐더러, 더욱이 몸이 불편한 노인 스스로 자신의 심리 상태를 파악해서 상담 받기는 어려운 일입니다. 둘째로, 앞으로 증가할 모든 독거노인들에게 돌봄 서비스를 제공하기란 비용 및 인력 차원에서 제한되는 부분입니다. 이러한 한계를 해소하기 위해 정부와 협력하여 기업에서도 노력하고 있습니다.

18 년도 이후부터는 IoT 기술이 돌봄 서비스에서도 활용되고 있습니다. 대표적인 예로 KT 의 IoT 기반 위치 트래커, 안심 LED 솔루션, 그리고 TV 시청 형태를 통한 모니터링 시스템이 있습니다. 하지만, 이는 모두 독거 노인의 정신적 건강을 모니터링하는 서비스가 아닌 독거 노인의 위치 파악을 통한 신체적 건강 모니터링을 주로 집중하고 있습니다. 즉, IoT 기술의 활용을 통해서 많은 독거 노인이 혜택을 누릴 수 있게

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31


되었지만, 감정 및 정신적 건강 모니터링에 집중하지 않아 정작 가장 중요한 조기발견은 어렵다는 한계점을 가집니다.

우울증은 조기에 발견되면, 정신과 질환 중에서 가장 치료가 잘 되는 질환 중의 하나이므로 빠른 발견과 적절한 치료를 받는다면 대다수의 사람들이 정상적인 일상으로 회복할 수 있습니다. 따라서 우리는 1인 가구(특히 독거노인) 인원의 감정상태를 모니터링하여 우울증을 조기에 발견하고 빠르게 치료할 수 있도록 돕는 서비스를 소개합니다.

우리 기술은 독거 노인이 ai 스피커에 말을 걸게 되면 ai 서버에서 독거 노인의 음성을 이용해서 감정 분석 및 알맞은 응답을 생성합니다. 이를 음성 합성을 통해서 음성을 통해 독거 노인에게 적절한 응답을 출력합니다.

독거 노인의 음성과 음성을 통해 분석한 감정들은 모두 DB에 저장되며, 2주간의 대화 목록을 분석해서 특정 Threshold 이상의 우울이 감지된다면 보호자에게 알리를 줍니다. 만약 보호자가 알리를 받지 않는다면 그 다음날 알람을 다시 보내며, 보호자가 알리를 확인할 때까지 알람을 보냅니다. 만약 보호자가 알람을 받는다면 그 즉시 최근 7일간의 데이터를 제외한 나머지를 삭제하고, 다시 2주간의 데이터를 수집해 분석합니다.

이를 통해서 독거 노인의 우울증이 심각해지기 이전에 알리를 주어 적절한 치료를 받게 할 수 있을 것으로 기대됩니다.

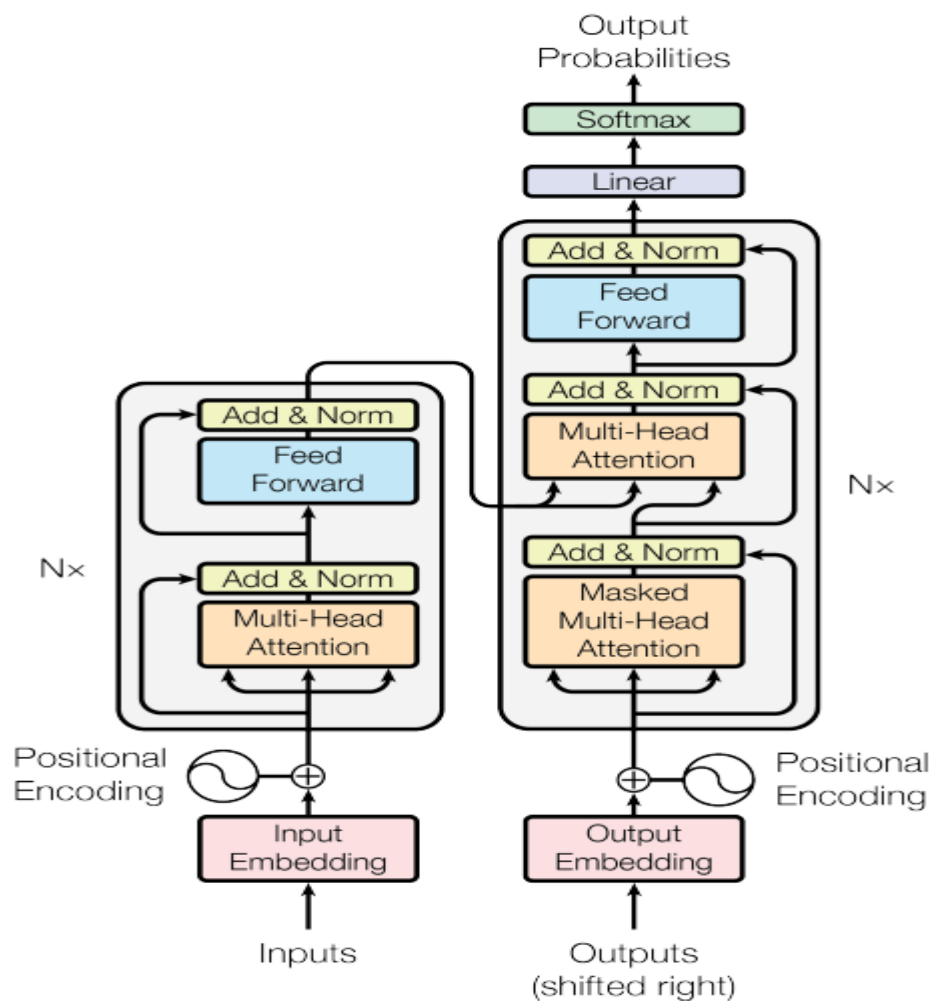
 <div> <p>국민대학교</p> <p>소프트웨어학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

2 수행 내용 및 중간결과


2.1 배경 지식

2.1.1 Transformer(Attention is all you need)

Transformer 모델은 기존 RNN모델의 문제점을 개선하고자 나온 self-attention 기반 모델입니다. RNN은 기계 번역 task에서 문장을 encoding할 때, 고정된 encoding vector로 표현하기 때문에 정보의 손실이 발생합니다. 따라서 Transformer 모델에서는 고정된 vector로 표현하지 않고, 문장 전체에 대한 encoding vector로 나타냅니다.



[그림 1] Transformer 구조

	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

[그림 1]은 Transformer의 구조입니다. Transformer의 핵심 요소는 1. Positional Encoding 2. Multi-head attention 3. Feed Forward Network입니다.

1. Positional Encoding


Input sequence가 순차적으로 들어가는 RNN과 다르게, Transformer는 input sequence 전체가 모델의 input으로 한 번에 들어가게 됩니다. 그리고 내부적으로 token간의 position을 learnable하게 배울 수 있는 parameter 혹은 장치가 전혀 없기 때문에 사람이 명시적으로 token의 위치를 알려주는 작업이 필요합니다. 이를 위해서 Transformer 논문에서는 sin과 cos함수의 결합을 통해서 absolute positional encoding을 진행합니다.

$$PE_{(pos,2i)} = \sin(pos/10000^{2i/d_{model}})$$

$$PE_{(pos,2i+1)} = \cos(pos/10000^{2i/d_{model}})$$

[그림 2] Transformer absolute positional encoding

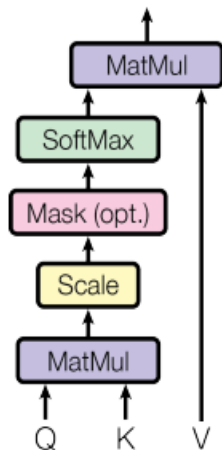
[그림 2]는 Transformer에서 사용되는 positional encoding 공식입니다. 위의 식을 통해서 각 token의 위치를 unique하게 나타내는 역할을 합니다.

 <div> <p>국민대학교</p> <p>소프트웨어학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

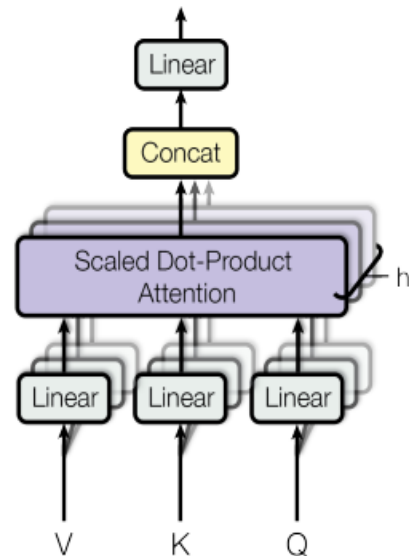
2. Multi-head attention

Attention은 가중치 값을 통해서 모델의 표현력을 높이는 방법입니다. Transformer에서는 다양한 Attention 방법 중 self-attention을 이용해서 표현력을 높입니다.

Scaled Dot-Product Attention




Multi-Head Attention



[그림 3] Transformer Attention

[그림 3]는 Transformer에서 사용된 Attention 방식입니다. Multi-head attention을 진행하기 위해서 input embedding으로부터 linear layer를 통해서 Q, K, V matrix를 생성합니다.

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

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

[그림 4] scaled-dot product

생성된 Q, K, V를 기반으로 [그림 4]의 수식을 통해서 token 별 가중치를 계산합니다. 이를 V와 matrix multiplication을 통해서 Attention을 진행합니다. Multi-head attention에서는 input embedding을 head의 개수로 chunking하고 chunking한 matrix에 대해서 scaled-dot product를 진행해서 여러 관점에서 바라본 vector를 계산합니다.

3. Feed Forward Network

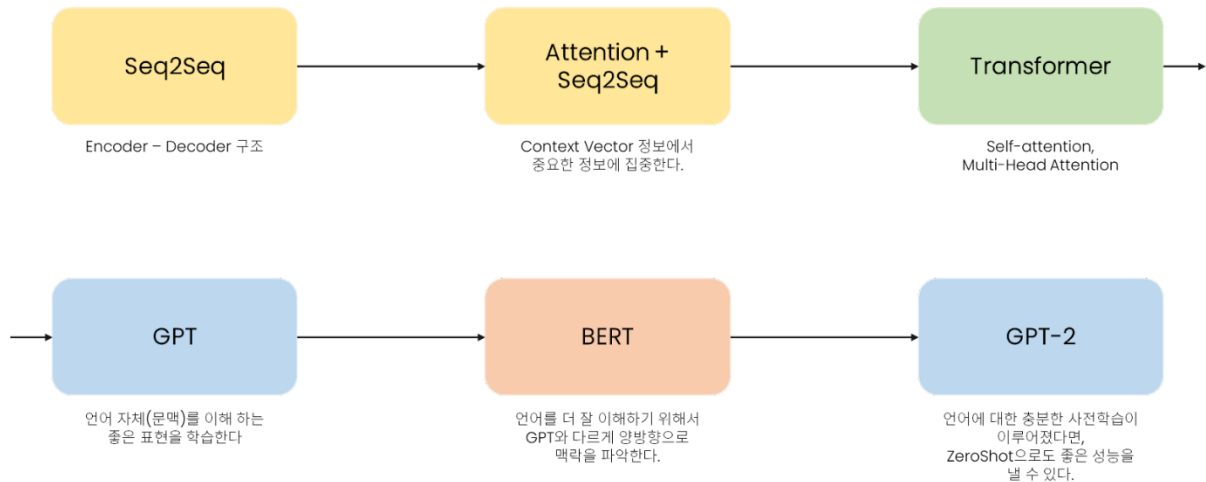
Feed Forward Network는 2개의 mlp 연산과 2개의 GELU로 구성된 Network입니다. Multi-head attention을 통해서 token간의 연산을 했다면, Feed Forward Network를 통해서 token의 embedding내의 learnable한 연산을 진행합니다.

Transformer 모델에서는 2. Multi-head attention을 통해서 token간의 연산을 진행하고, 3. Feed Forward Network를 통해서 하나의 token 내부의 연산을 진행합니다.

Transformer는 Encoder와 Decoder 구조로 이루어져 있습니다. [그림 3]의 왼쪽 부분은 Encoder이고, 오른쪽 부분은 Decoder입니다. Encoder와 Decoder는 Multi-head attention과 Feed Forward Network로 이루어진 블록이 여러 개 쌓여 있는 구조입니다.

Decoder에서는 왼쪽의 단어들을 이용해서 다음에 나올 단어를 예측하는 방식으로 작동합니다. 따라서 Decoder에서는 Masked-Multi-head attention을 통해서 현재 참고할 수 있는 token만을 이용해 다음 단어를 예측합니다.

Transformer 구조 등장 이후, 대부분의 NLP Task에서 Backbone으로 많이 사용되고 있습니다. 이는 이후 소개할 BERT와 GPT 모델에도 영향을 주었습니다.




[그림 5] NLP의 역사

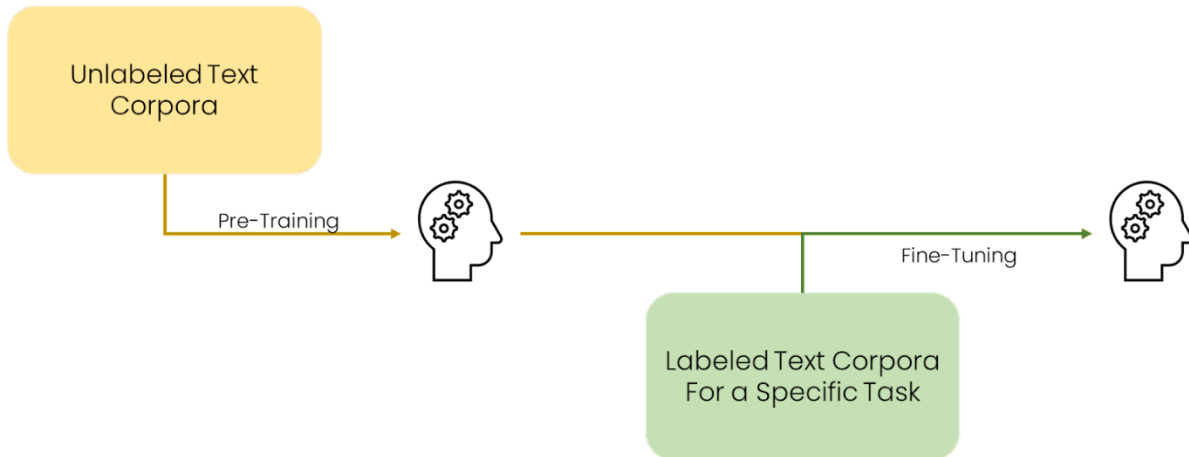
2.1.2 GPT(Generative Pre-Training)

GPT는 자연어 처리의 특정 task에서만 사용할 수 있는 모델이 아닌 모든 task에서 사용할 수 있는 언어 모델(Language Model)입니다.

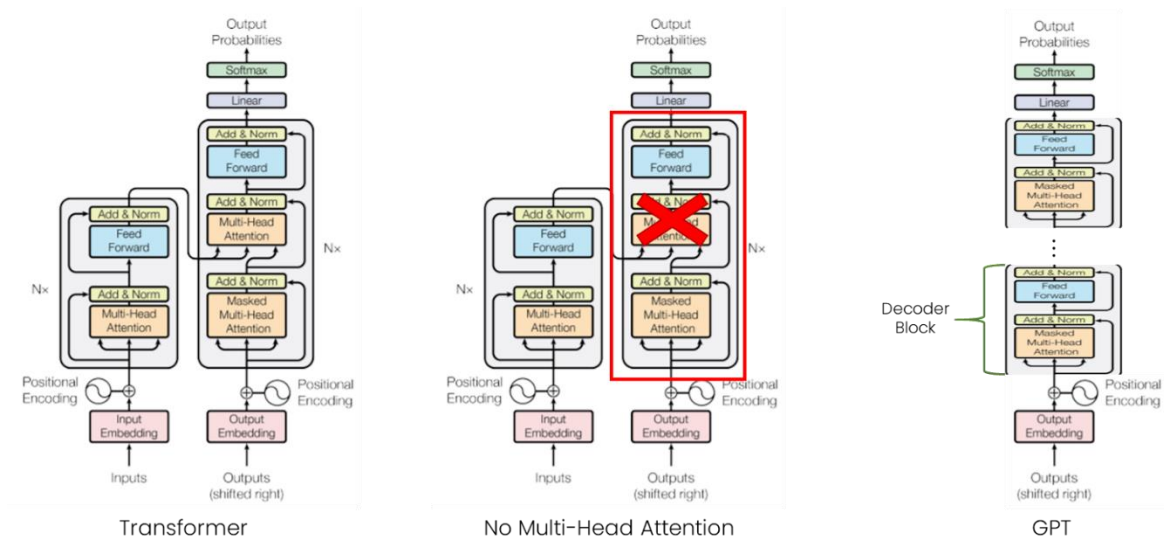
당시 NLP분야에서는 데이터 폭발의 시대에서 unlabeled data를 사용해 task specific하지 않은 모델을 만들고자 하는 시도들이 있었습니다. Unlabeled data는 labeled data에 비해 labeling에 대한 cost가 없고 대규모 수집이 용이합니다. 이 점을 이용해 pre-training을 통해 방대한 unlabeled 자연어 data의 특징을 학습한 언어 모델을 만들고, task에 맞게 fine-tune하는 방법론이 제시되었습니다. 이는 바로 비지도 사전학습(unsupervised pre-training)이며 GPT모델의 핵심 방법론입니다.

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

1. 큰 말뚝치에서 대용량의 언어모델을 사전학습(Un-Supervised Pre-Training)
2. Specific Task에 맞게 Labeled 된 Dataset에 대해서 언어모델을 미세조정(Supervised Fine-Tuning)

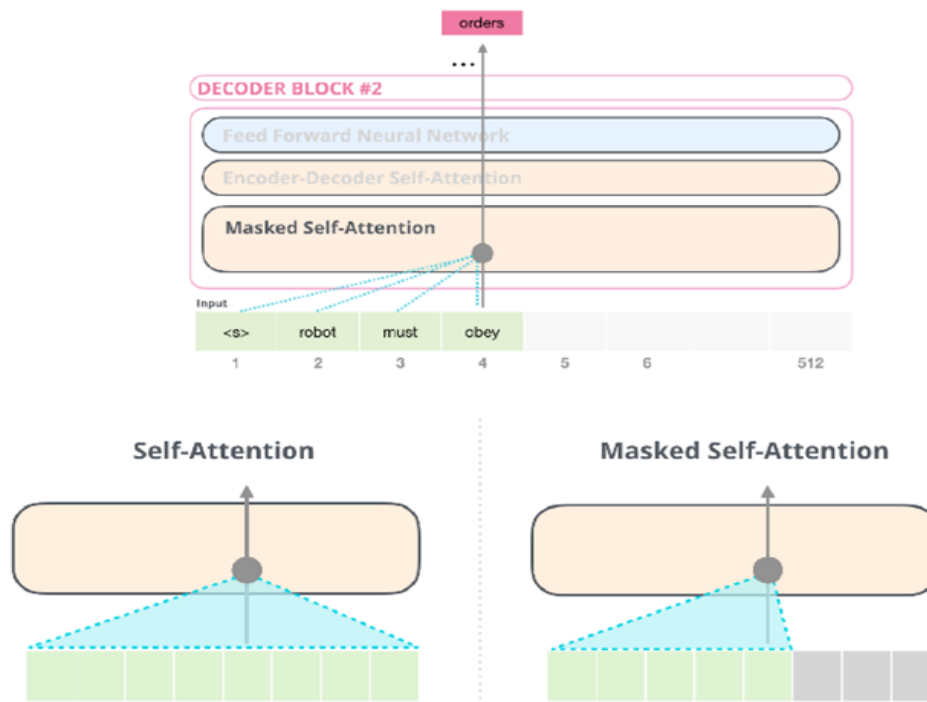


[그림 6] GPT 학습 개요



[그림 7] GPT 구조

GPT 모델은 기본적으로 Transformer의 Decoder 단의 구조를 가집니다. 단, Decoder단의 Multi-head attention을 제외하고 Masked Multi-head attention과 Feed Forward Network의 조합으로 구성됩니다.



[그림 8] self-attention과 masked self-attention 비교


Masked Multi-head attention은 기존 Encoder-Decoder Multi-head attention과 달리 현재 추론하고 있는 Token까지만 self-attention의 score를 적용합니다. 그 외의 score에는 $-\infty$ 으로 변경합니다. 즉, 현재까지 단어들의 정보를 가지고 이후에 나타날 단어를 예측하도록 설계되는 것입니다.

1. GPT input

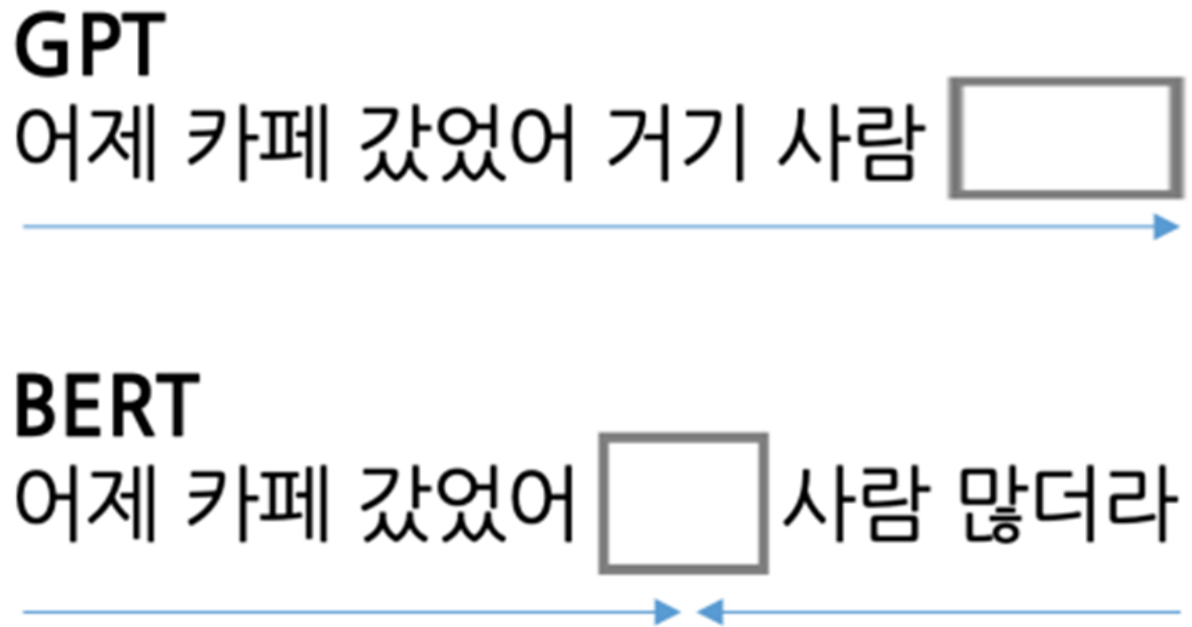
GPT의 입력은 자연어인 문장이 입력으로 들어갑니다. 물로 GPT는 인간의 언어를 이해할 수 없기 때문에 vector의 형태로 표현해야 합니다. 우선 Byte Pair Encoding(BPE)를 거쳐 표현한 후 등록된 vocab을 통해 token화 합니다. 해당 token은 token embedding과 positional encoding을 거쳐 input vector로 변환됩니다.

2. GPT's Pre-training

GPT의 사전학습 방법은 후술할 BERT와 다르게 순차적으로 진행되면서 다음 단어를 예측하도록 학습됩니다. 순차적으로 다음 단어에 대한 prediction을 하기 때문에, next token prediction 능력이 중점적으로 학습됩니다. 해당 학습 방법이 GPT에서 유리하게 작용하여

 <div> <p>국민대학교</p> <p>소프트웨어학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

문장 생성 등의 task에서 강한 성능을 보입니다.



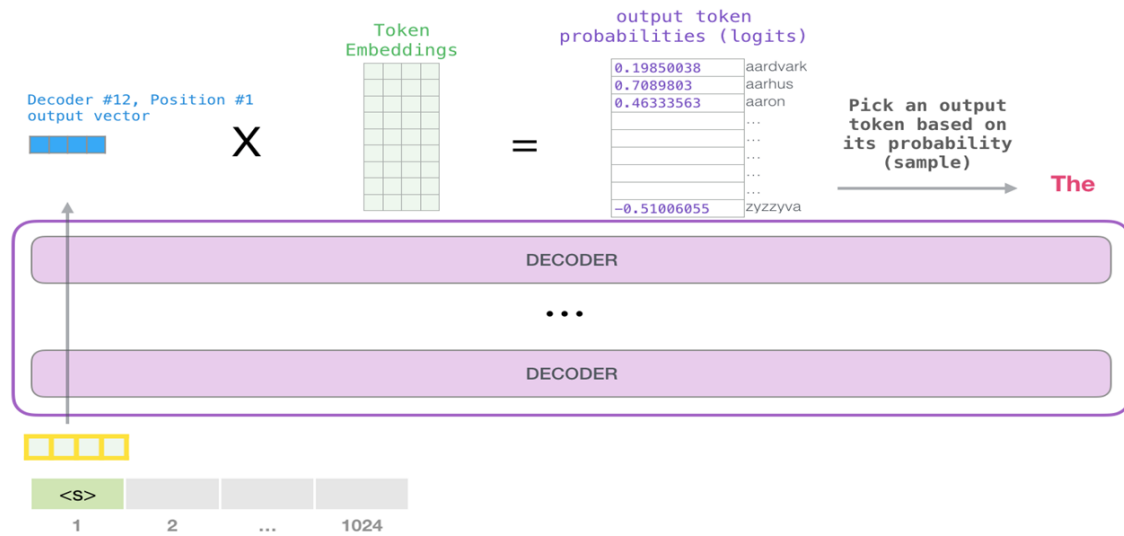
[그림 9] GPT, BERT의 학습 방법 비교

3. GPT’s Fine Tuning

Unlabeled data를 통해 사전학습이 완료되면 specific task에 맞춰 가중치를 미세조정 합니다. 단순히 문장의 긍정/부정을 구분하는 분류(classification) task라면, GPT의 최상 위 layer의 출력 값을 선형 변환하여 SoftMax를 취해 이진 분류하는 형식으로 설계할 수 있습니다.

4. GPT’s 문장 생성

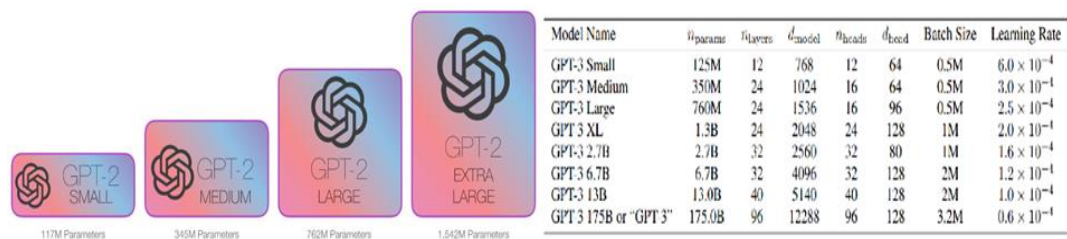
입력 token을 여러 Decoder block을 거쳐 최상위 block를 통해 Embedding된 표현벡 터가 나오면, 해당 벡터를 token embedding matrix와 내적 하여 vocab에 존재하는 모 든 단어들에 대한 확률 값으로 표현(logits)합니다. 해당 logits를 SoftMax를 취하여 가 장 높은 확률을 가진 token을 출력하여 다음 입력으로 대입합니다. 해당 과정을 종료 token이 출력될 때까지 반복하는 방식으로 문장 생성을 구현할 수 있습니다.




[그림 10] GPT의 문장 생성 방식

5. GPT, GPT-2, GPT-3

GPT, GPT-2, GPT-3는 버전 업이 됨에 따라 architecture의 개선이 이루어졌지만, Transformer decoder block을 사용하는 구조에는 큰 차이 없이 동일합니다. 버전마다 다른 것은 학습 데이터의 규모와 Decoder block의 개수입니다. GPT-3는 사칙연산, 번역, 웹 코딩 등등 여러 task에 적용이 가능합니다. GPT논문의 저자 Open AI에서는 현재 GPT-2까지만 코드를 open하였고, GPT-3에 대해서는 잠재적인 위험과 악용을 이유로 유료로 api를 제공하고 있습니다.



[그림 11] GPT 모델 파라미터

	국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서	
		프로젝트 명	AID(AI Doctor)
		팀 명	캡스톤 디자인 1 12팀
		Confidential Restricted	Version 1.0 2022-MAR-31

6. SKT-Brain KoGPT-2

앞선 GPT, GPT-2, GPT-3의 논문은 영어에 대해 사전학습을 한 모델이었습니다. 이는 한국어에 대해서는 성능이 떨어지기 때문에, 한국어 데이터셋에 대해 사전학습이 된 모델을 사용해야 합니다. 저희는 GPT-2를 한국어로 사전학습한 SKT-Brain의 KoGPT-2를 Hugging Face를 통해 받아와 Fine-tune하여 사용했습니다.

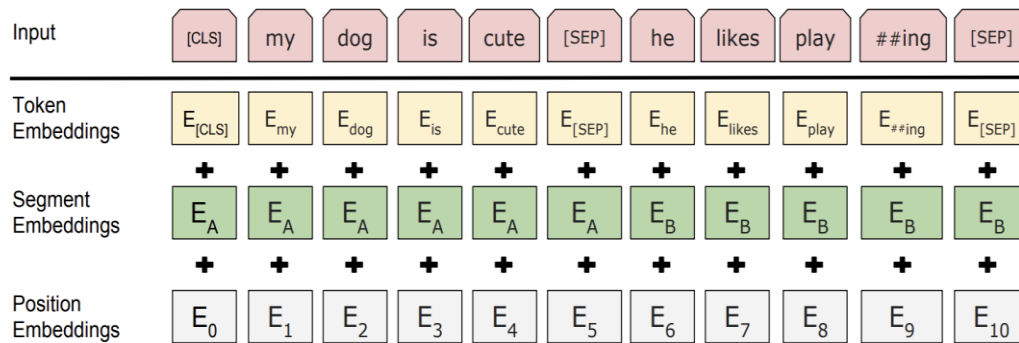
SKT-KoGPT-2는 40GB이상의 한국어 말뭉치(한국어 위키 백과, 뉴스, 모두의 말뭉치 v1.0, 청와대 국민청원)을 통해 사전학습 되었습니다. 사전의 크기는 512000이며, 대화에 자주 쓰이는 이모티콘 이모지를 추가하여 토큰의 인식력을 높였습니다.

2.1.3 BERT(Bidirectional Encoder Representations from Transformers)

Bert는 unlabeled text를 이용해서 단어를 embedding하는 모델입니다. Bert는 일부 단어를 masking해서 이를 예측하도록 만드는 방식으로 학습합니다. 이를 통해서 bidirectional하게 학습을 진행해 좋은 성능을 이끌어 냈던 모델입니다. 아래 배경지식에서는 Bert 원본 논문을 기준으로 소개하고, 한글 pre-train 모델인 KLUE는 아래 섹션에서 소개하겠습니다.

1. Input Embedding

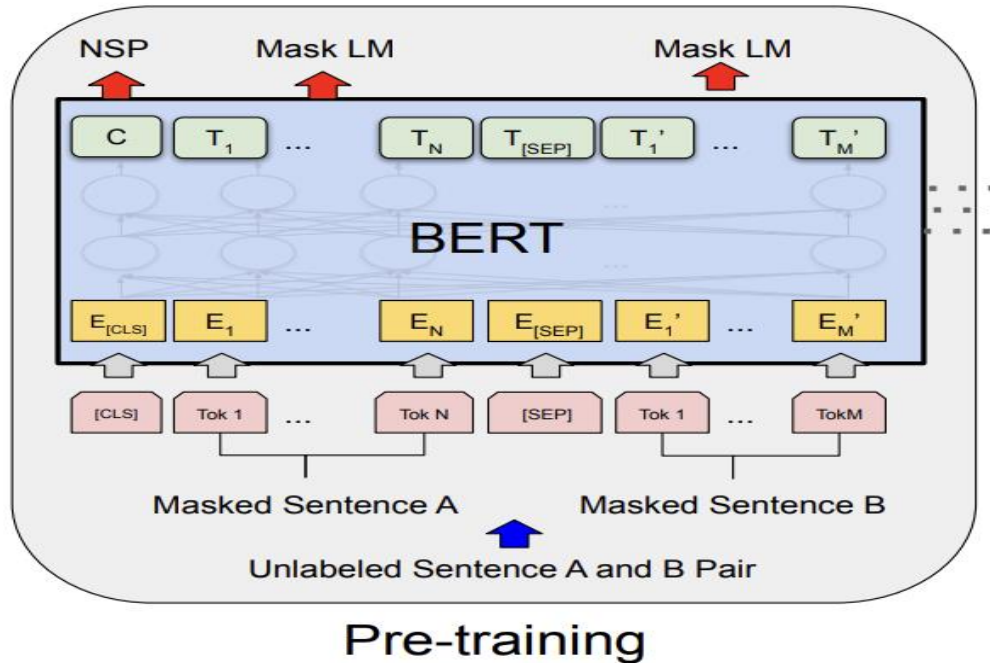
Bert를 학습시키기 위해서는 언어를 기계가 이해할 수 있는 vector형태로 표현해야 합니다. 이를 위해서 word piece embedding 방식이 사용되었습니다. Word piece embedding 방식은 의미를 표현할 수 있는 최소 단위로 단어를 분리합니다. Bert의 input embedding에서는 하나의 문장 혹은 두개의 문장이 합쳐진 것을 sequence라고 정의하고, sequence의 맨 앞에는 [CLS] token을 sequence에 있는 두개의 문장을 분리하기 위해서 [SEP] token을 사용했습니다.



[그림 12] Bert의 input Embedding

[그림 12]는 Bert의 input Embedding을 나타내는 그림입니다. 총 3가지의 Embedding이 사용되었습니다. 먼저 position Embedding은 token의 순서를 나타내기 위해서 사용한 Embedding 방식입니다. 이는 Transformer 원본 논문과 동일한 absolute positional encoding방식이 사용되었습니다. Segment embedding은 어떤 문장에 속해 있는지를 나타내는 Embedding입니다. 마지막으로 token embedding은 word piece embedding을 통해서 각 최소 의미 단위를 vector의 형태로 나타내는 것을 의미합니다. Bert에서는 [MASK]를 통한 예측과 다음 문장 예측을 동시에 진행하면서 학습이 되기 때문에 기존 Transformer에서 segment embedding이 추가된 형태로 embedding을 합니다.

2. 모델 구조




[그림 13] BERT 구조

BERT의 구조는 [그림 13]와 같습니다. 구체적인 모델 구조는 Transformer 원본 논문에서 사용된 구조에서 Decoder 부분을 제거한 형태와 동일합니다.

3. BERT pretraining 방법

BERT의 pre-train은 두가지 task에 대해서 진행합니다. 하나는 Masked LM이고, 다른 하나는 Next Sentence Prediction입니다.

Masked LM은 input embedding의 일부 단어를 masking하여 이 단어를 예측하도록 학습하는 방식입니다. 이는 word2vec의 학습 방법과 유사하다고 볼 수 있습니다. 이를 통해서 모델이 문맥을 이해할 수 있도록 만듭니다. 하지만, BERT에서는 fine-tune시에 input embedding에 [MASK] token이 등장하지 않기 때문에 pre-train시에 (1) 80% 단어는 [MASK]로 변경하고 (2) 10%의 단어는 임의의 token으로 변경하고 (3) 10%의 단어는 바꾸지 않은 상태로 학습을 진행합니다. 구체적으로는 [그림 16]과 같이 masking이 진행되어 학습됩니다.

	중간보고서		
	국민대학교	AID(AI Doctor)	
	소프트웨어학부	캡스톤 디자인 1 12팀	
	캡스톤 디자인 I	Confidential Restricted	Version 1.0 2022-MAR-31


- 80% of the time: Replace the word with the [MASK] token, e.g., my dog is hairy → my dog is [MASK]
- 10% of the time: Replace the word with a random word, e.g., my dog is hairy → my dog is apple
- 10% of the time: Keep the word unchanged, e.g., my dog is hairy → my dog is hairy. The purpose of this is to bias the representation towards the actual observed word.

[그림 14] BERT pre-train시 masking 전략

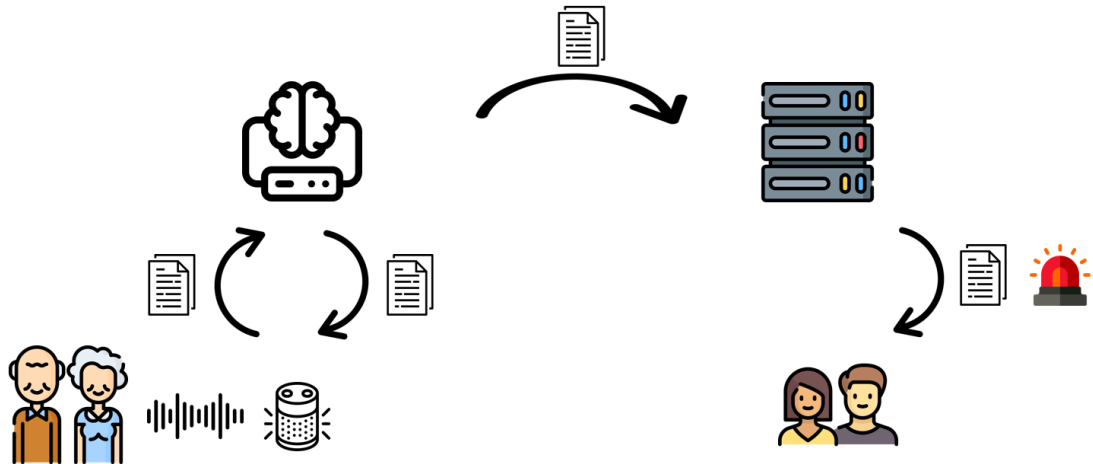
Next Sentence Prediction은 [CLS] token의 BERT 모델을 통과한 이후의 Embedding을 통해서 Is Next인지 Not Next인지를 판단하는 binary classification을 진행하면서 학습합니다.

BERT 원본 논문은 영어를 기반으로 학습된 모델입니다. 따라서 한국어 서비스를 하기에는 부적합합니다. 따라서 많은 연구자들이 BERT를 한국어로 학습시키기 위해서 노력했고, Hugging Face를 통해서 pre-train 모델이 공개되었습니다. 저희 팀은 그 중 하나인 KLUE pre-train 모델을 사용했습니다.

모델을 pre-train 시키기 위해서 사용된 데이터는 5가지(MODU, CC-100-KOR, NAMUWIKI, NEWSCRAWL, PETITION)이고, 총 62.65GB입니다. 수집된 데이터의 윤리적 문제를 피하기 위해서 전화번호, 홈페이지 주소 등 민감한 정보는 fake 데이터로 바꾸어서 학습을 진행했습니다. Vocab는 32k개로 사용했고, Byte Pair Encoding(BPE)방식으로 vocab을 만들었습니다. BPE 이전에 형태소 단위로 분리한 이후, 알고리즘을 진행합니다. BPE 방식은 빈도수를 기반으로 형태소를 합치면서 vocab을 만드는 방식입니다. BPE이외에도 단어 단위로 분리해서 진행할 수도 있습니다. 하지만, 이는 신조어에 대해서 대처하기 어렵기 때문에 BPE를 사용하는 것이 유리합니다.

 <div> <p>국민대학교</p> <p>소프트웨어학부</p> <p>캡스톤 디자인 I</p> </div>	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

2.2 계획서 상의 연구 내용

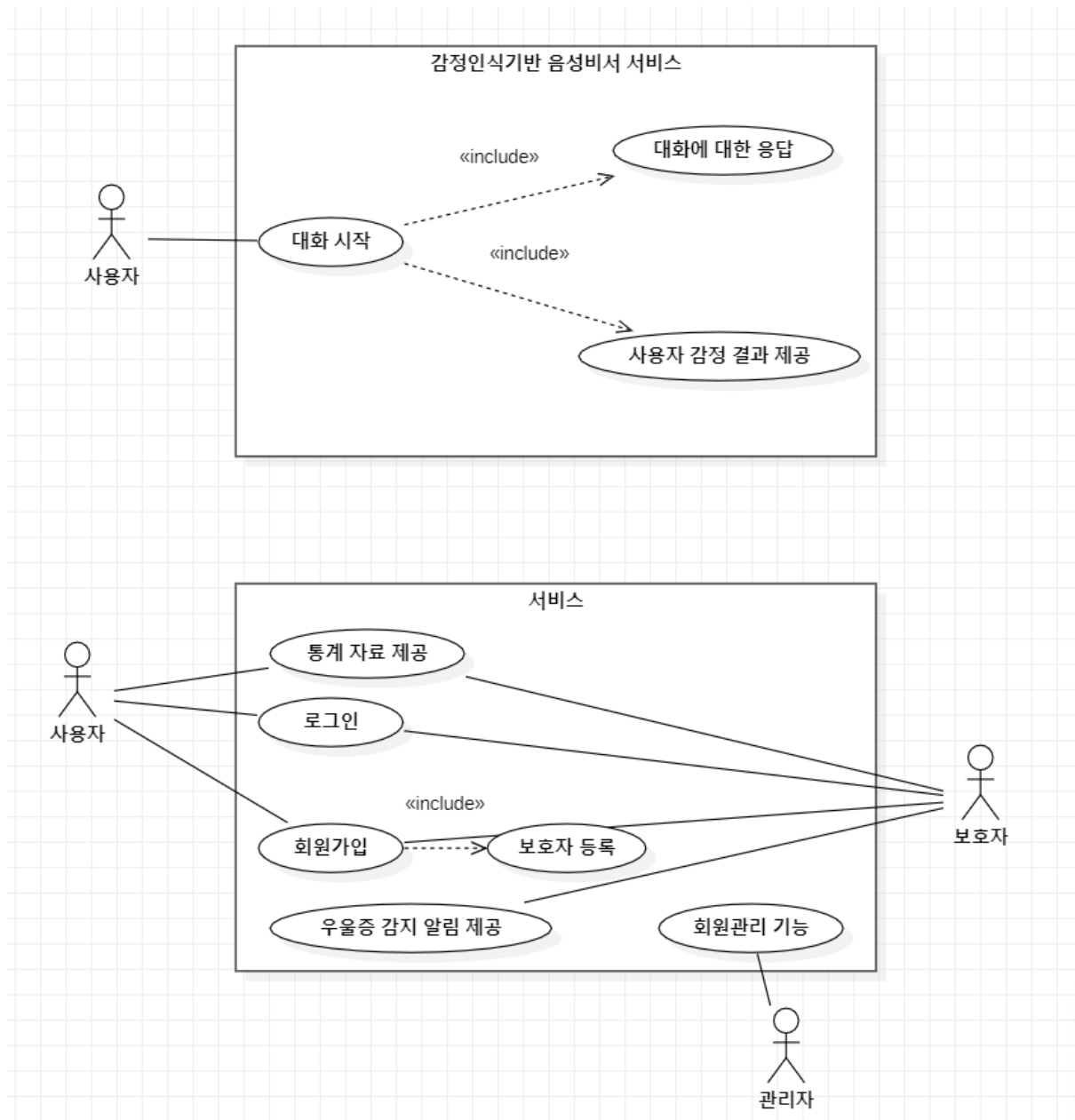


[그림 15] 시스템 구성도

[그림 15]은 저희 팀이 구상한 전체 시스템 구성도입니다. 위의 시스템을 제작하기 위해서는 1. 음성을 텍스트로 변환 2. 텍스트의 감정 분석 3. 입력 텍스트에 알맞은 응답 return 4. 음성 합성 5. 보호자/기관 담당자에게 감정 상태 모니터링이 가능하도록 알림 서비스 5가지 기능을 구현해야 합니다. 이번 section에서 각 기능 구현의 계획을 소개하겠습니다.

1. 음성을 텍스트로 변환: AI Hub에서 제공하는 음성 인식 api를 통해서 음성을 텍스트로 변환합니다.
2. 텍스트의 감정 분석: 문장 전체의 의미를 파악할 수 있는 딥러닝 모델을 사용하여 진행합니다. 대용량의 데이터를 수집해서 처음부터 학습하기는 어렵기 때문에 pre-train 모델을 사용해서 fine-tune을 진행합니다.
3. 입력 텍스트에 알맞은 응답 return: 문장 전체의 의미를 파악해 적절한 응답을 할 수 있는 generation 모델을 사용합니다. 대용량의 데이터를 수집해서 처음부터 학습하기는 어렵기 때문에 pre-train 모델을 사용해서 fine-tune을 진행합니다.
4. 음성 합성: python에서 손쉽게 사용할 수 있는 라이브러리를 사용합니다.

5. 보호자/기관 담당자에게 감정 상태 모니터링이 가능하도록 알림 서비스: node.js의 express, django등과 같은 웹 프레임워크를 사용해서 api server를 생성합니다.



[그림 16] usecase diagram

[그림 16]는 usecase diagram을 그린 것입니다.

2.3 수행내용

현재 1차적으로 1-4번 과정은 완료된 상황입니다. 아래는 1-4번 과정에 대한 자세한 소개입니다.

1. 음성을 텍스트로 변환: 계획대로 AI Hub API를 활용해서 텍스트로 변환했습니다.
2. 텍스트의 감정 분석:

모델	정확도	시간(100개 기준)
Human(30명 대상)	65.17	-
Bert-Base	79.0	0.48
Robert-Base	77.0	0.49
Robert-Small	77.0	0.31
Robert-Small-Attention	78.0	0.36

[표 1] 감정 분석 모델 성능 비교

[표 1]은 실험에서 사용한 감정 분석 모델입니다. 사용한 데이터는 AI Hub에 공개된 감성 대화 말뭉치 데이터입니다. 위의 데이터는 국가 주도하에 수집된 정제된 데이터이기 때문에 저희 팀이 현재 수집할 수 있는 데이터 중 신뢰도가 높다고 생각해서 선택하게 되었습니다.

위의 데이터는 baseline에 대한 정보가 없기 때문에 학습한 모델이 어느정도 추론을 잘하는지 판단이 어렵습니다. 따라서 저희 팀은 baseline 성능을 설정하기 위해서 모델 test시 사용할 100개의 데이터에 대해서 설문조사를 진행했습니다. [그림 17]는 설문조사의 샘플 사진입니다.

11 중 2 섹션

설문지(1/10)

각 문항에서 제시된 문장에서 느껴지는 감정을 보기에서 골라 주세요.

아프다고 약속을 취소한 친구가 사실 거짓말을 한 거였어. 내가 싫은 건가 *

- ☐ 기쁨
- ☐ 불안
- ☐ 슬픔
- ☐ 분노

[그림 17] 설문조사


총 29명을 대상으로 설문조사를 진행했습니다. 그 결과 평균은 65.17/100, 중앙값은 66/100, 최소값은 56, 그리고 최대값은 74로 나타났습니다.

감정 분석 모델은 계획대로 문장 전체의 의미를 파악할 수 있는 모델을 사용하고자 했습니다. 이를 위해서 Transformer 기반의 모델 중 pre-train 모델이 공개 되어있는 모델을 사용했습니다. Pre-train 모델은 hugging face를 통해서 사용할 수 있는 klue의 가중치를 사용했습니다.

감정 분석 모델은 pre-train 모델에서의 [CLS] token의 embedding을 linear classifier를 통해 분류를 진행했습니다. [CLS] token을 사용한 이유는 문장 전체에 대한 표현을 대표할 수 있기 때문이었습니다.

Linear classifier를 추가한 모델에 표현력 향상을 기대하고 상위 4개의 block에서의 embedding에 대한 attention을 추가적으로 적용해 실험을 진행했습니다.

저희 팀은 정확도와 시간을 기준으로 판단했을 때, Bert-Base와 linear classifier를 사용한 모델이 적합하다고 생각했습니다.

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

- 입력 텍스트에 알맞은 응답 return: AI Hub의 데이터가 모델을 학습하기에 충분하지 않기 때문에 감정 분석 모델과 동일하게 pre-train 모델을 활용해서 fine-tune을 했습니다. Pre-train모델은 skt-brain에서 공개한 GPT-2를 활용했습니다. GPT-2기반에 AI Hub 데이터를 이용해서 fine-tune을 진행했습니다.
- 음성 합성: 음성 합성은 python 라이브러리인 gTTS와 pygame을 사용해서 진행했습니다. 음성 합성에서는 2가지 시도를 했습니다. 하나는 file로 응답 음성 파일을 저장하는 방법과 다른 하나는 file로 저장하지 않고 응답을 출력하는 방식이었습니다. 하지만, 두 가지 방법에서 시간 차이가 없어서 file로 응답 음성 파일을 저장하는 방법을 선택했습니다.

3 수정된 연구내용 및 추진 방향


3.1 수정사항

현재까지 계획에서 수정된 사항은 없습니다. 다만 웹개발이 빠르게 끝난다면 시스템을 확장할 생각입니다.

구체적으로 혐오 표현에 대한 filtering기능과 일정 주기로 스피커를 통해 질문을 하는 기능을 추가하고자 합니다.

혐오표현 filtering은 Bert와 같은 큰 모델을 사용하지 않고 간단한 모델을 사용하고자 합니다. 이를 통해서 ai 스피커에서 발생할 수 있는 윤리 문제에 대해 방지하고자 합니다.

일정 주기로 스피커를 통해 질문하는 기능은 특정 시간에 노인에게 질문을 하고 노인의 답변을 듣는 방식으로 구현할 계획입니다. 우울증의 전조 증상 중 하나가 질문에 대한 답변을 대부분 '모른다'로 답하는 경우가 많다고 합니다. 이 점을 활용해서 질문에 대한 답변을 하지 않거나, '모른다'와 유사도가 높은 답변의 비율이 지정한 Threshold보다 높다면 이 역시 알람을 주는 기능을 추가하고자 합니다.

 국민대학교 소프트웨어학부 캡스톤 디자인 I	중간보고서		
	프로젝트 명	AID(AI Doctor)	
	팀 명	캡스톤 디자인 1 12팀	
	Confidential Restricted	Version 1.0	2022-MAR-31

4 향후 추진계획

4.1 향후 계획의 세부 내용

보호자/기관 담당자에게 감정 상태 모니터링이 가능하도록 알림 서비스에 대한 기능을 개발해야 합니다. 현재 backend로는 express를 사용하고 frontend로는 react를 사용하고자 합니다.

AI 서버에서 입력 받은 텍스트와 GPT를 통해서 나온 응답 텍스트에 대해서 혐오 표현 filter을 추가하고자 합니다. 이를 통해서 윤리적으로 문제가 되지 않을 만한 표현만을 출력하도록 만들 예정입니다.

5 고충 및 건의사항

멘토님 배정이 되면 좋겠다는 생각이 들었습니다. 자연어 처리와 인공지능에 대한 공부를 진행했지만, 실용성 관점에서 조언을 얻고 싶다는 생각이 들었습니다.

모델 학습을 위한 GPU 서버 제공을 받고 싶습니다. DLPC 사용에 대해 공지가 있으면 좋겠습니다.

6 참고 문헌

- [1] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33, 1877-1901.
- [2] Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- [3] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.