

중간 발표 1차

For

시각 장애인을 위한 상황 묘사 어플리케이션

Team 4

정준호(201113275) 전민규(201411802) 김도연(201614157)



Contents



프로젝트 소개

최종 산출물 & 1차 구현

System Testing Results

Pass / Fail Criteria

Component Diagram

2nd Iteration 계획

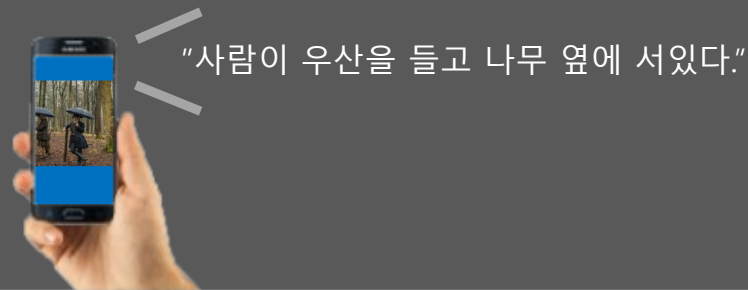
시각 장애인을 위한 상황 묘사 어플리케이션

' 시각 장애인을 위한 상황 묘사 어플리케이션 ' 이란 ?

시각 장애인분들을 위해 접근성이 쉬운 핸드폰 카메라로 찍히는 사진의 상황을 글 및 소리로 묘사해주는 어플리케이션입니다.

' 시각 장애인을 위한 상황 묘사 어플리케이션 ' 이 작동하는 방식은 ?

스마트폰을 들고 있으면 자동적으로 10초마다 한 장씩 사진을 찍어 그에 대한 설명을 음성으로 들려줍니다.



최종 산출물 & 1차 구현



- 전송 받은 사진을 학습된 모델단에 input
- 학습된 모델을 걸쳐 나온 사진에 적합한 문장을 output

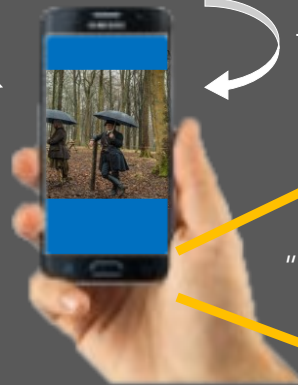
사진 전송

문장 전송

10초마다
사진 찍기

TTS API 를 통해 문장을 음성으로 출력

"사람이 우산을 들고 나무에 기대고 있는 사
진"



System Testing Results

TEST CASE	NAME	DESCRIPTION	Pass / Fail
1-1	전면부 상황 입력	<ul style="list-style-type: none">스마트폰 앱에서 10초마다 사진이 찍히는지 테스트 한다.	Pass
1-2	전면부 상황 입력	<ul style="list-style-type: none">새로 핸드폰 카메라로 '사람이 우산을 들고 나무 옆에 있는 사진'을 찍고 그 사진을 이미 GCP에서 학습된 weight를 갖고 있는 모델의 입력으로 넣고, 모델 입력 단에서 카메라로 찍은 사진 frame이 들어오는지 확인 하는 코드를 추가한다.Input image가 잘 들어오면 'OK', 아니면 'No Image'라는 메시지를 코드단에서 출력하도록 한다.	Fail
2	딥러닝 학습을 통한 모델	<ul style="list-style-type: none">Test data(train에서 사용하지 않은 MSCOCO 8000장)로 모델을 평가하였을 때, 평균적인 BLUE Score-4 기준 어절단위 0.111 의미형태소 단위 0.225 형태소 단위 0.251 이상이면 신뢰도를 갖고 있다고 판단한다.	Fail
3	학습된 모델을 통한 결과 도출	<ul style="list-style-type: none">위에서 정의한 '사람이 우산을 들고 나무 옆에 있는 사진' 과 그에 맞는 문장 '사람이 우산을 들고 나무 옆에 서있음' 을 준비해두고, 준비해둔 사진을 모델의 입력으로 넣었을 때, 미리 준비해둔 그 사진을 설명하는 문장 '사람이 우산을 들고 나무 옆에 서있음'과 유사한 문장이 10초 이내에 출력 되는지 확인한다.	Fail
4	TTS (Text To Speech)	<ul style="list-style-type: none">결과로 나온 문장을 TTS API를 통해 음성으로 바꿔서 출력하는지를 체크한다.	Pass

Pass / Fail Criteria

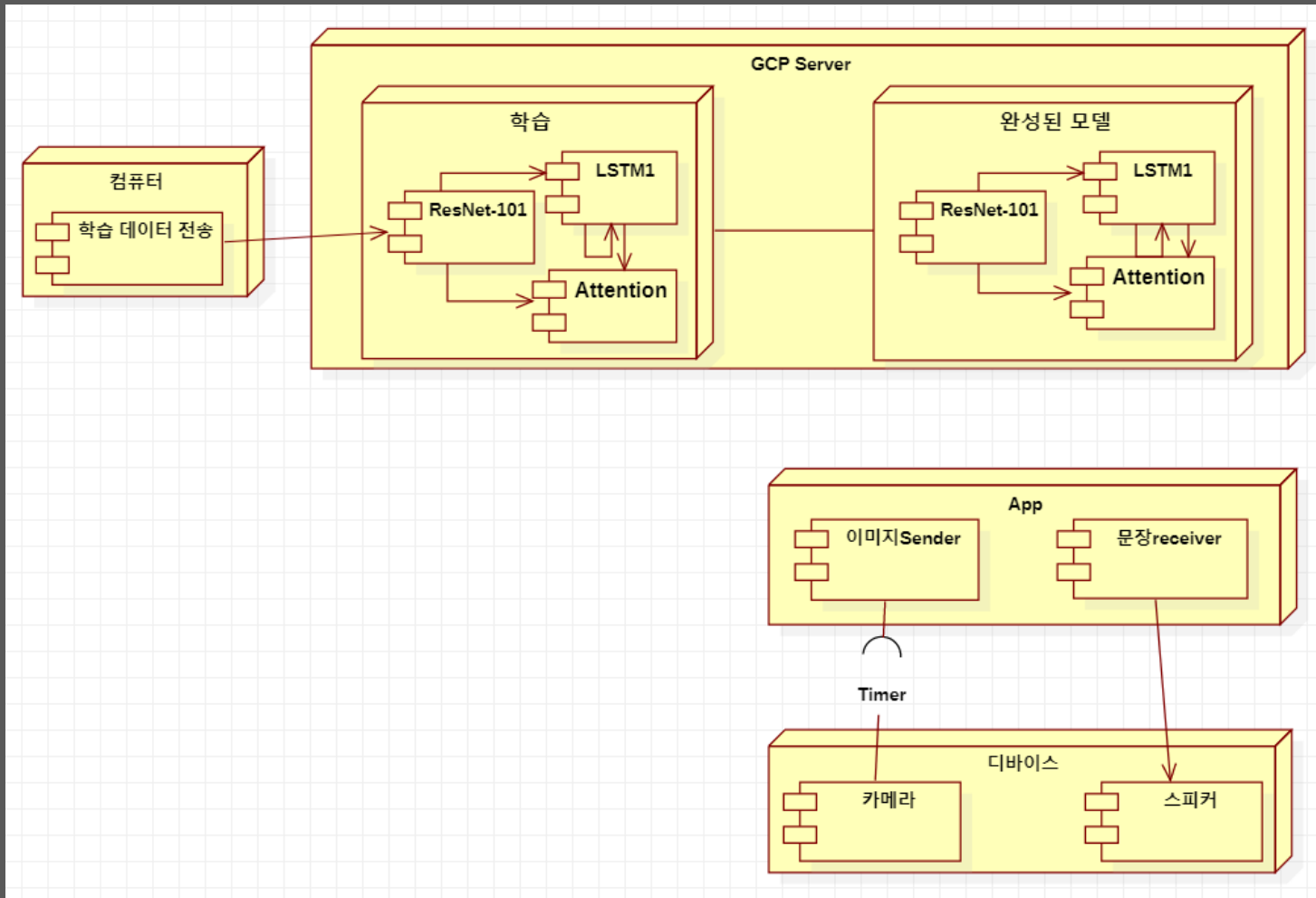
TEST CASE	NAME	Pass / Fail	Reason
1-1	전면부 상황 입력	Pass	→ 10초마다 정상적으로 사진이 찍힌다.
1-2	전면부 상황 입력	Fail	→ 아직 앱과 서버를 연결하지 못해서 앱에서 찍은 사진을 서버에 있는 모델단에 넣어주지 못했다.
2	딥러닝 학습을 통한 모델	Fail	<p>→ 모델은 Design단계에서 말씀드렸던 것과 동일하게 Encoder/Attention/Decoder 다 구현했으나 내부적인 것을 살짝 바꿔 완성시켰다. Ex) Encoder : RCNN -> ResNet-101로 바꾸었고 Decoder : LSTM1 & LSTM2 -> LSTM1 만 구현</p> <p>→ train 시켜본 결과, 정했던 BLUE Score-4 기준에 도달하지 못했다.</p> <p>→ 또한 실제 사진을 input하여 나온 결과를 확인하여 보니 사진의 상황과 부합하지 못하는 문장을 만들어내는 것을 볼 수 있었다. Ex) 데모 - '사람이 우산을 들고 나무 옆에 서있음' 이 아닌 '푸른 하늘을 배경으로 한 흐릿한 사진' 이 라는 적합하지 않은 문장을 출력</p>
3	학습된 모델을 통한 결과 도출	Fail	→ 아직 앱과 서버를 연결하지 못해서 앱에서 사진을 찍는 순간부터 앱에서 음성을 출력하는 것까지의 총 시간을 재지 못했다.
4	TTS (Text To Speech)	Pass	→ 앱에서 그냥 임의의 문장을 출력하게 했을 때, 문장을 정확하게 읽었다.

Traceability Matrix

Requirement NO.	Req Desc	Design (H) (Module)	Design(모듈) NO.	Seq Diagram NO.	Design(H) Desc	Design (L) (Class)	Design(L) Desc	Pass / Fail
1-1	10초 마다 주기적으로 사진을 촬영	App	A.1	1	사진 요청	CameraActivity	takePicture()	Pass
		Device	D.1	2	사진 return	카메라		
1-2	주어진 사진을 모델단에 입력	App	A.2	1	사진 전송	CameraActivity	sendImage(image)	Fail
		GCP Server	G.1	2	사진 전송 체크	RCNN	getImage()	Fail
2	딥러닝 학습을 통한 모델	GCP Server	G.2	1	사진 입력	CameraActivity	sendImage(image)	Fail
		Resnet(Encoder)	R.1	2	feature map (A) 추출	resnet	resnet	Pass
			R.2	6	추출한 A input	resnet	sendFeatureMap()	Pass
			R.3	8	추출한 A input	resnet	sendFeatureMap()	Pass
		LSTMCell1	L1.1	3	이전 hidden_state1, memory_cell1 값 input	LSTMCell	getBeforeHiddenState()	Pass
			L1.2	7	hidden_state1 추출	LSTMCell	getBeforeMemoryCell()	Pass
			L1.3	9	hidden_state1 input	LSTMCell	sendHiddenState(h1_n)	Pass
			L1.4	12	hidden_state1 input	LSTM1	sendHiddenState(h1_n)	Pass
		attention	Att.1	10	attended feature 추출	Attention	attendFeature (input, h1_n,h2_n-1)	Pass
			Att.2	11	attended feature input	Attention	sendAttendFeature (att_n)	Pass
		LSTM2	L2.1	4	이전 hidden_state2 output	LSTM2	sendHiddenState()	Fail
			L2.2	5	이전 hidden_state2, memory_cell2 값 input	LSTM2	getBeforeHiddenState(), getBeforeMemoryCell()	Fail
			L2.3	13	hidden_state2 추출	LSTM2	sendHiddenState()	Fail
			L2.4	14	hidden_state2-word output	LSTM2	w_q	Fail
3	학습된 모델을 통한 결과 도출	App	A.3	2	SendImage	CameraActivity	sendImage()	Fail
		GCP Server	G.3	3	SendText	SaveSentence	getWord(), createSentence(word[n])	Fail
4	TTS (Text To Speech)	GCP Server	G.4	1	SendText	SaveSentence	sendSentence(Sentence)	Fail
		App	A.4	2	TTS	MainActivity	getSentence(), Speech()	Pass
		Device	A.5	3	음성출력	스피커		

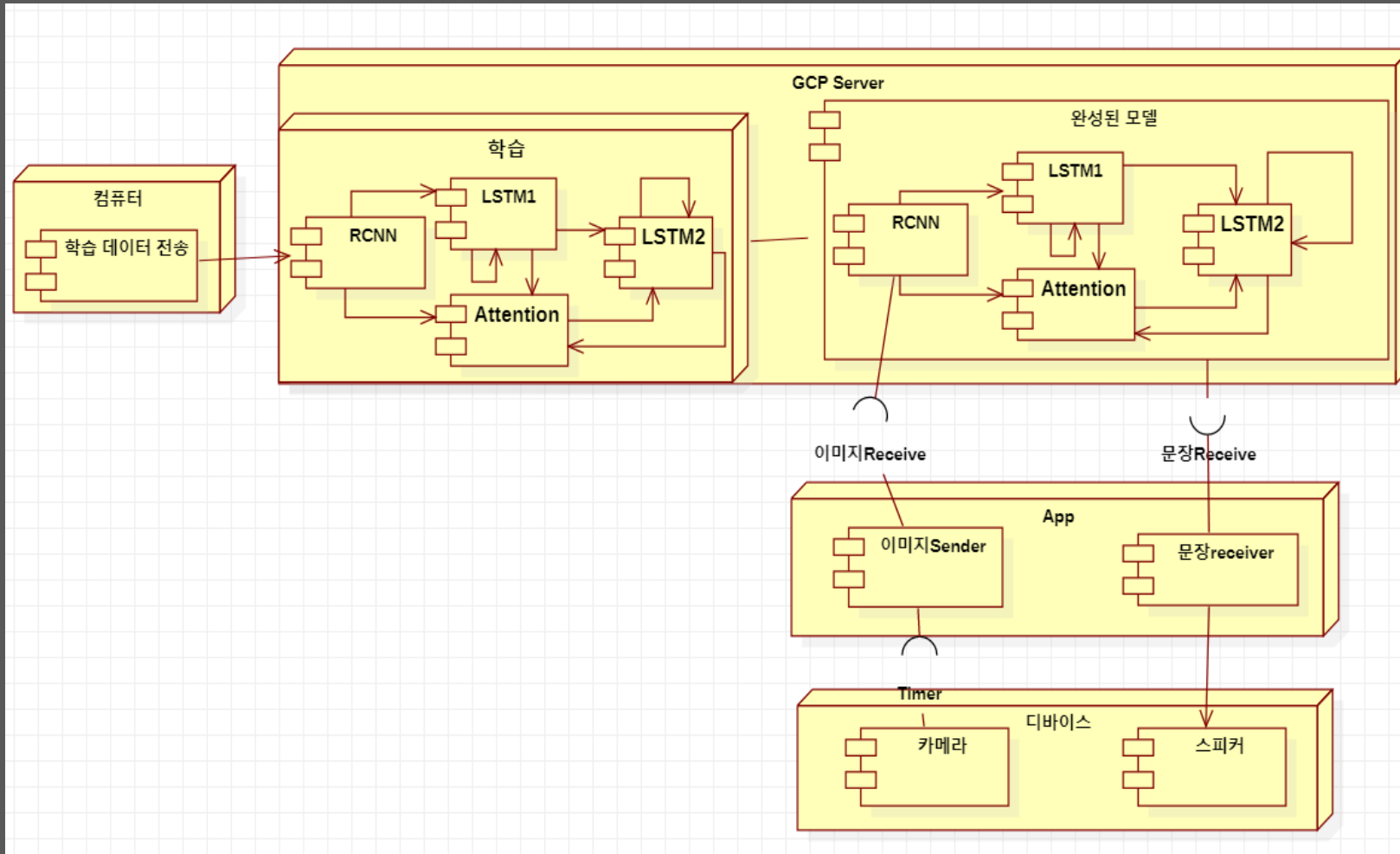
Component Diagram

1st Iteration



Component Diagram

2nd Iteration



2nd Iteration 계획

Date	5월 20일	5월 21일	5월 22일	5월 23일	5월 24일	5월 25일	5월 26일	5월 27일
Do	Change Encoder, Make LSTM2							
		Connect App And GCP Server						
				Training				
					Check the time from beginning to end			
							Performance Check	

TEST CASE	NAME	Pass / Fail	Plan
1-1	전면부 상황 입력	Pass	X
1-2	전면부 상황 입력	Fail	앱과 서버를 연결시켜 모델단에 사진이 들어가는지 확인한다.
2	딥러닝 학습을 통한 모델	Fail	Encoder 부분을 ResNet-101에서 RCNN으로 바꾸고 LSTM2를 추가해서 성능 향상을 한다. 평균적인 BLUE Score-4 기준 어절단위 0.111 의미형태소 단위 0.225 형태소 단위 0.251 이상에 도달하도록 계속 학습을 시킨다.(이건 3rd Iteration까지 계속될 예정)
3	학습된 모델을 통한 결과 도출	Fail	앱과 서버를 연결시켜 앱에서 사진을 찍는 순간부터 앱에서 음성을 출력하는 것까지의 총 시간을 잰다. 50장으로 평균 총 시간을 구해 10초 이내면 Pass이다.
4	TTS (Text To Speech)	Pass	X