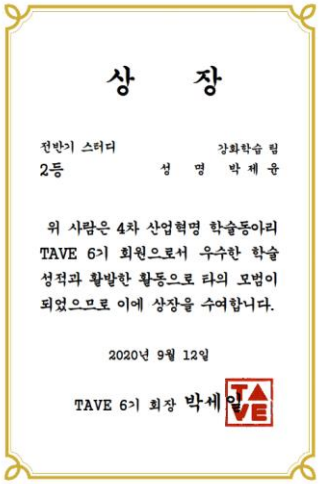
TAVE OB Lecture:
Natural Language Processing & Multimodal

발표자: **박제윤**
TAVE 6기

TAVE TECHNOLOGY WAVE

Who Am I?

- TAVE 6기
- TAVE 6기 전반기 스터디 2등: 강화학습 팀
- TAVE 6기 후반기 프로젝트 1등:
  - (1) 발화 감정 정보 분석을 통한 추천 시스템
  - (2) 다중 도메인 자율성장 AI 기반 대화 시스템
- TAVE 6기 전반기, 후반기 우수회원상
- TAVE 7기 경영처장
- TAVE OB lecture (11/27/2021): [pdf]

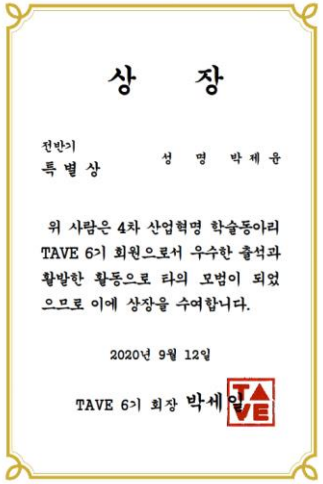

상 장

전반기 스터디 강화학습 팀
2등 성 명 박 채 윤

위 사람은 4차 산업혁명 학습동아리 TAVE 6기 회원으로서 우수한 학습 성적과 활발한 활동으로 타의 모범이 되었으므로 이에 상장을 수여합니다.

2020년 9월 12일

TAVE 6기 회장 박세일

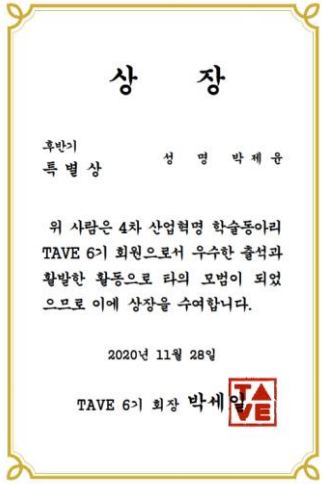

상 장

전반기
특별상 성 명 박 채 윤

위 사람은 4차 산업혁명 학습동아리 TAVE 6기 회원으로서 우수한 출석과 활발한 활동으로 타의 모범이 되었으므로 이에 상장을 수여합니다.

2020년 9월 12일

TAVE 6기 회장 박세일

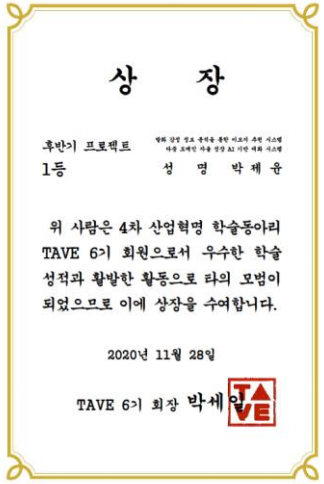

상 장

후반기
특별상 성 명 박 채 윤

위 사람은 4차 산업혁명 학습동아리 TAVE 6기 회원으로서 우수한 출석과 활발한 활동으로 타의 모범이 되었으므로 이에 상장을 수여합니다.

2020년 11월 28일

TAVE 6기 회장 박세일


상 장

후반기 프로젝트
1등 성 명 박 채 윤

위 사람은 4차 산업혁명 학습동아리 TAVE 6기 회원으로서 우수한 학습 성적과 활발한 활동으로 타의 모범이 되었으므로 이에 상장을 수여합니다.

2020년 11월 28일

TAVE 6기 회장 박세일

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

2022.01 - Now

1. 3rd AI SPARK Challenge (2022) [와! 최우수상!🎉 (3/61 = 4.91%)]

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon [본선 진출, 6월 8일 발표 평가😇]

3. Research: Grammatical Error Correction

4. Research: Video Summarization

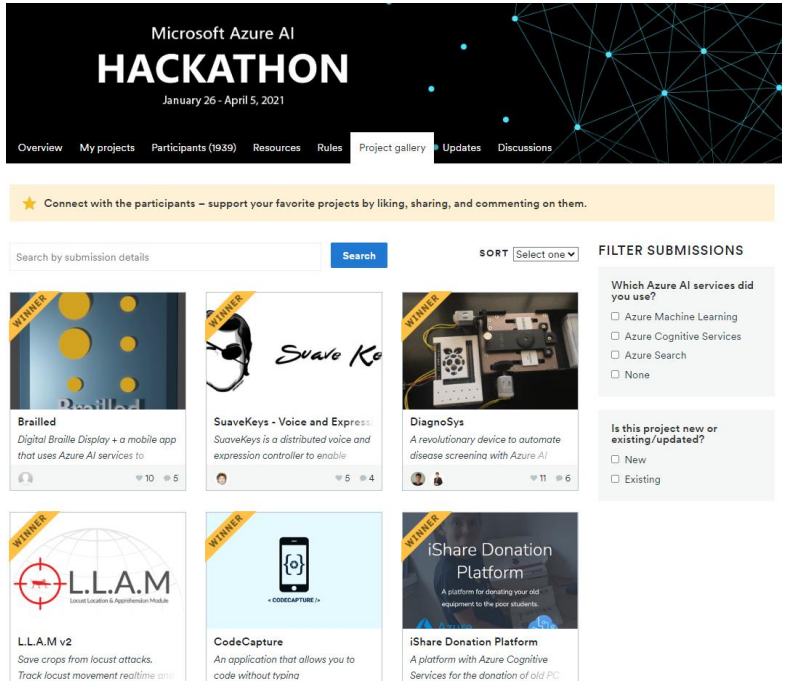
5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

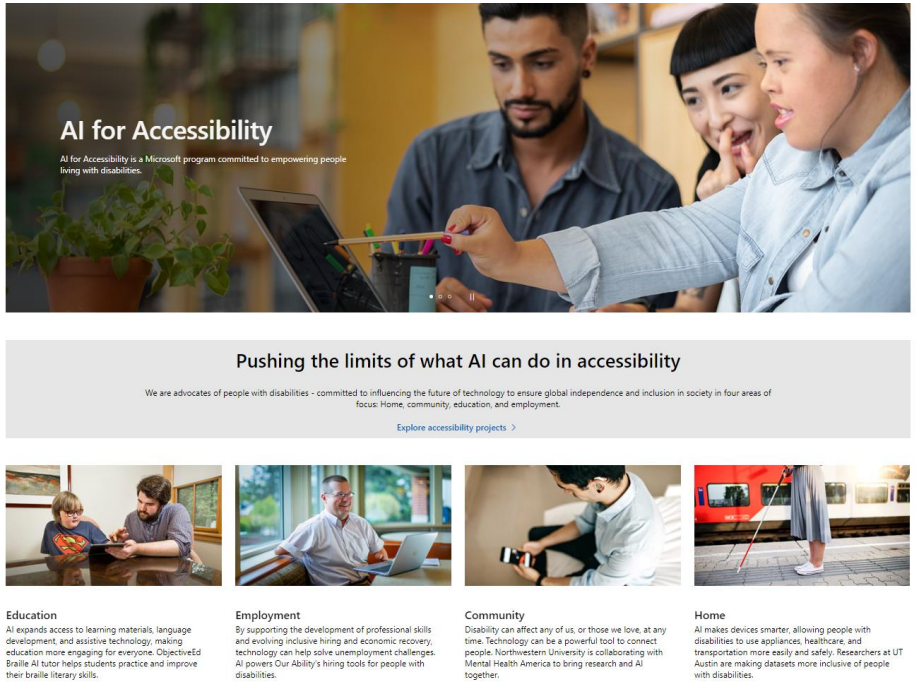
2022 Microsoft Azure Virtual Hackathon

1. Azure Virtual Hackathon 이란?

1) 주최: Microsoft and Github

- Microsoft의 AI solution인 Azure를 활용하여 우리가 직면한 다양한 사회적 문제를 해결하는 대회





2022 Microsoft Azure Virtual Hackathon

1. Azure Virtual Hackathon 이란?

2) 주제: 클라우드와 DevOps / 클라우드와 AI / 클라우드와 Gaming

- Mission: “환경, 인도주의적 문제, 보건 등 여러 분야에서 더 나은 세상을 만들기 위한 아이디어 및 AI 기술을 지원해주세요”

e.g.) 전 세계 기후 변화를 보다 잘 예측하여 대륙 및 1차 산업 종사자에게 도움을 주는 아이디어

e.g.) 새들이 자주 부상을 입는 유형을 파악하여, 공공 카메라로 식별된 새에 대한 자세한 정보를 공유하여 자연 환경에 도움을 줌

2022 Microsoft Azure Virtual Hackathon

2. 핵심 내용

1) **Hi Azure!**: **H**elp**i**ng video creators who are visually-impaired edit videos using **Azure** Cloud AI**!**

2) 3줄 요약

- 기존에는 시각 장애를 가진 영상 크리에이터들은 주로 청각정보에만 의존하여 영상 편집을 해왔음

- 이러한 점은 시각 장애인이 크리에이터가 되는 것과 생산하는 영상 컨텐츠 장르에 큰 장벽이 될 수 있음

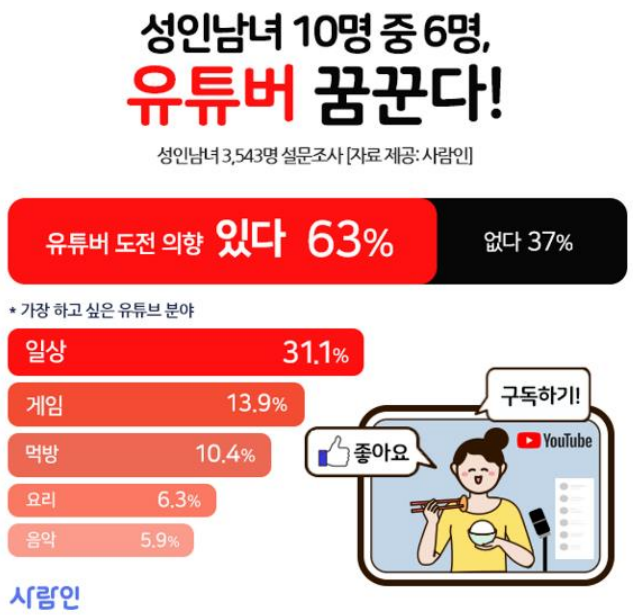
- 우리는 시각 장애 영상 크리에이터들이 컨텐츠 제작에 영상의 시각 정보도 같이 활용할 수 있는 서비스를 제안하였음

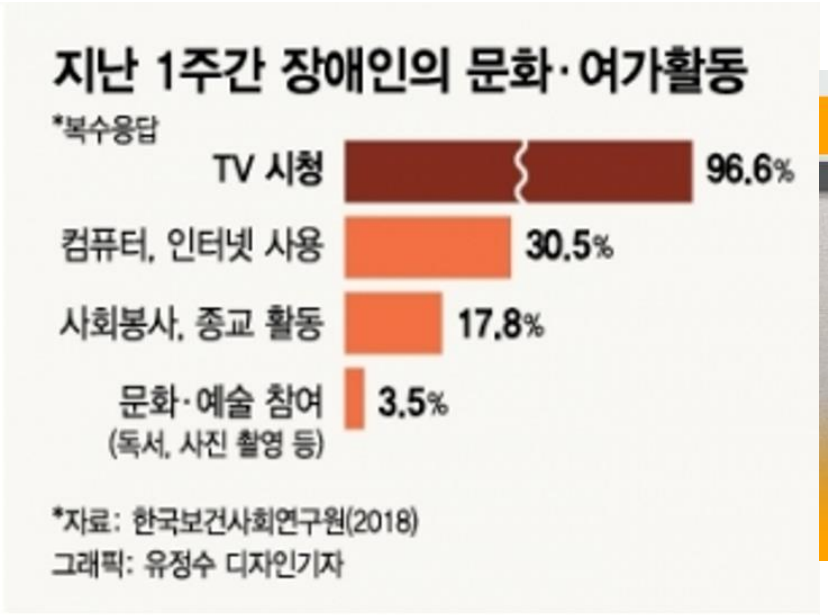
2022 Microsoft Azure Virtual Hackathon

2. 핵심 내용

3) 제안배경 1: 통계 조사

- 시각장애인들은 취미의 절대 다수는 TV 시청등의 정적인 활동이고, 이는 자신이 가진 끼와 재능을 더 다채롭게 펼치기에는 많은 현실적인 제약이 있다고 볼 수 있음





2022 Microsoft Azure Virtual Hackathon

2. 핵심 내용

3) 제안배경 2: 실제 시각장애 크리에이터들의 인터뷰

"편집은 혼자하기 어려워 친구가 도와줍니다. 누군가 편집을 어떻게 하냐고 물어보면 '좋은 친구를 사귀면 된다.'라고 말하죠. 그 친구에게 고마운 점이 많아요."

영상 크리에이터 '원샷한솔' 한국시작장애인복지관 인터뷰 中

"아무래도 시각장애가 있어서 혼자 컴퓨터나 이런 걸로 편집프로그램을 사용하기는 어려움이 있어요. 그래서 저는 도와주는 PD 언니한테 스크립트를 전달해서 영상을 만들고 있습니다."

영상 크리에이터 '우령의 유디오' TBS 인터뷰 中

편집은 어떻게 찍혔는지 기억해가며 진행한다.
이렇다 보니 6 ~ 7분가량의 영상을 만드는 데 3일이 넘는 시간이 걸릴 때도 있다.

영상 크리에이터 '브레드박' 한국일보 기사 中

2022 Microsoft Azure Virtual Hackathon

2. 핵심 내용

4) 제안하는 서비스: Hi Azure!

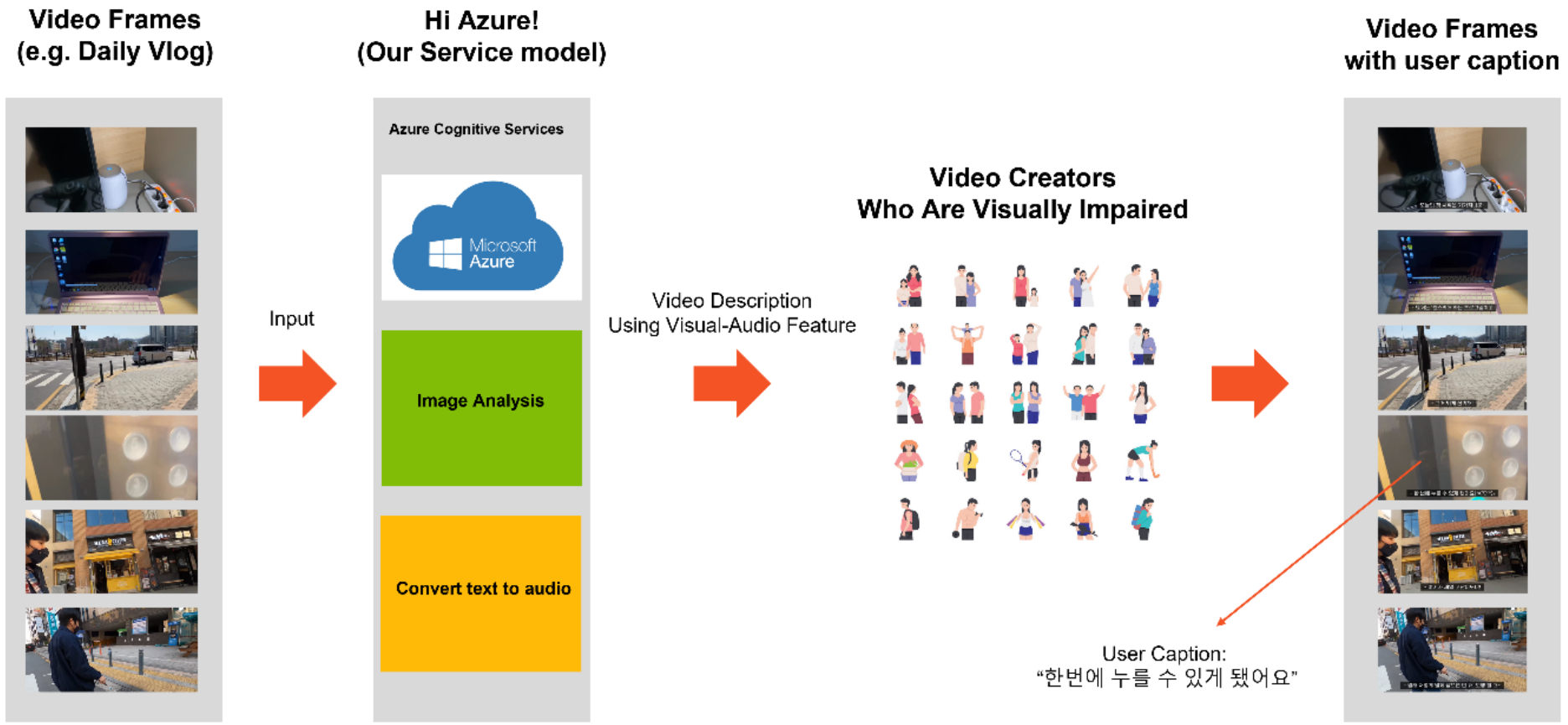
- 기존에는 시각 장애를 가진 영상 크리에이터들은 자신의 관심사나 특기, 취미등을 영상으로 자유롭게 풀어내고 싶어도 진입장벽이 너무 높아서 주변의 도움 없이는 청각정보에만 의존하여 컨텐츠를 제작할 수 밖에 없음

- 따라서, 우리는 이번 프로젝트에서 Azure Cognitive Service를 통해, 영상의 visual feature를 추출한 뒤 음성으로 변환하여 frame-level description으로 제공하여 시각장애 크리에이터 영상 편집 및 컨텐츠 제작에 대한 접근성을 높이려고 함

2022 Microsoft Azure Virtual Hackathon

3. 기술 세부 내용

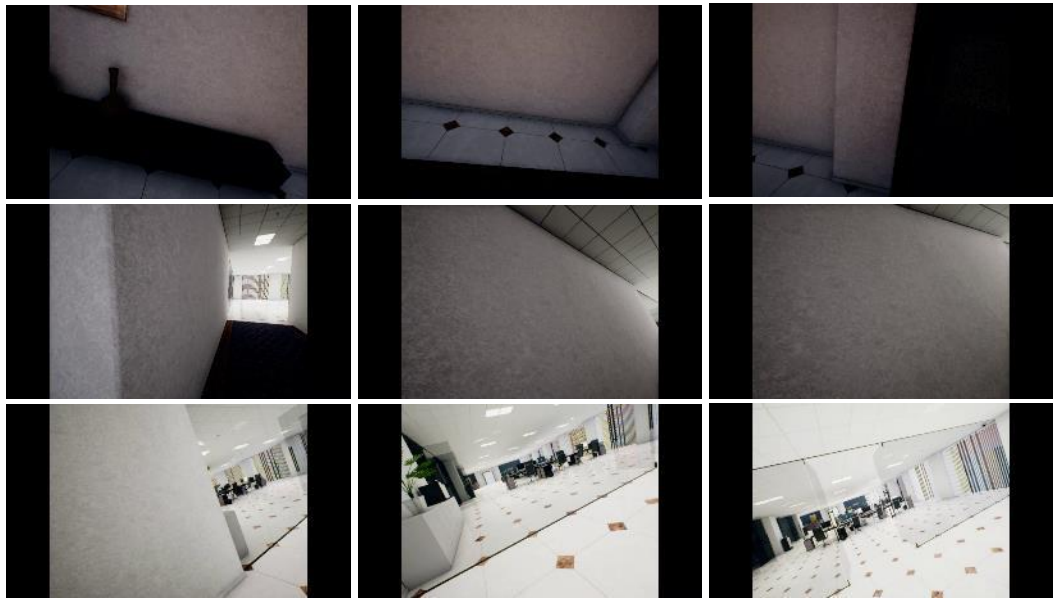
1) A schematic depiction of our service

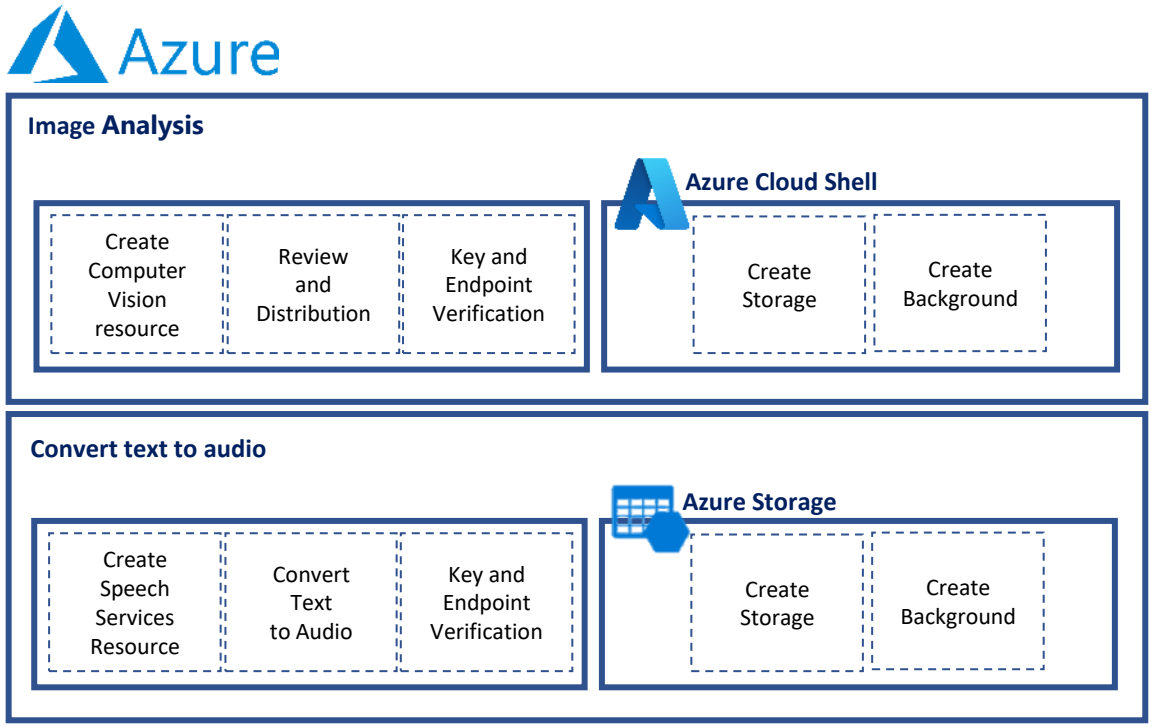


2022 Microsoft Azure Virtual Hackathon

3. 기술 세부 내용

2) Employing Azure to (1) analyzing videos features, (2) generating captions, and (3) converting captions to audio

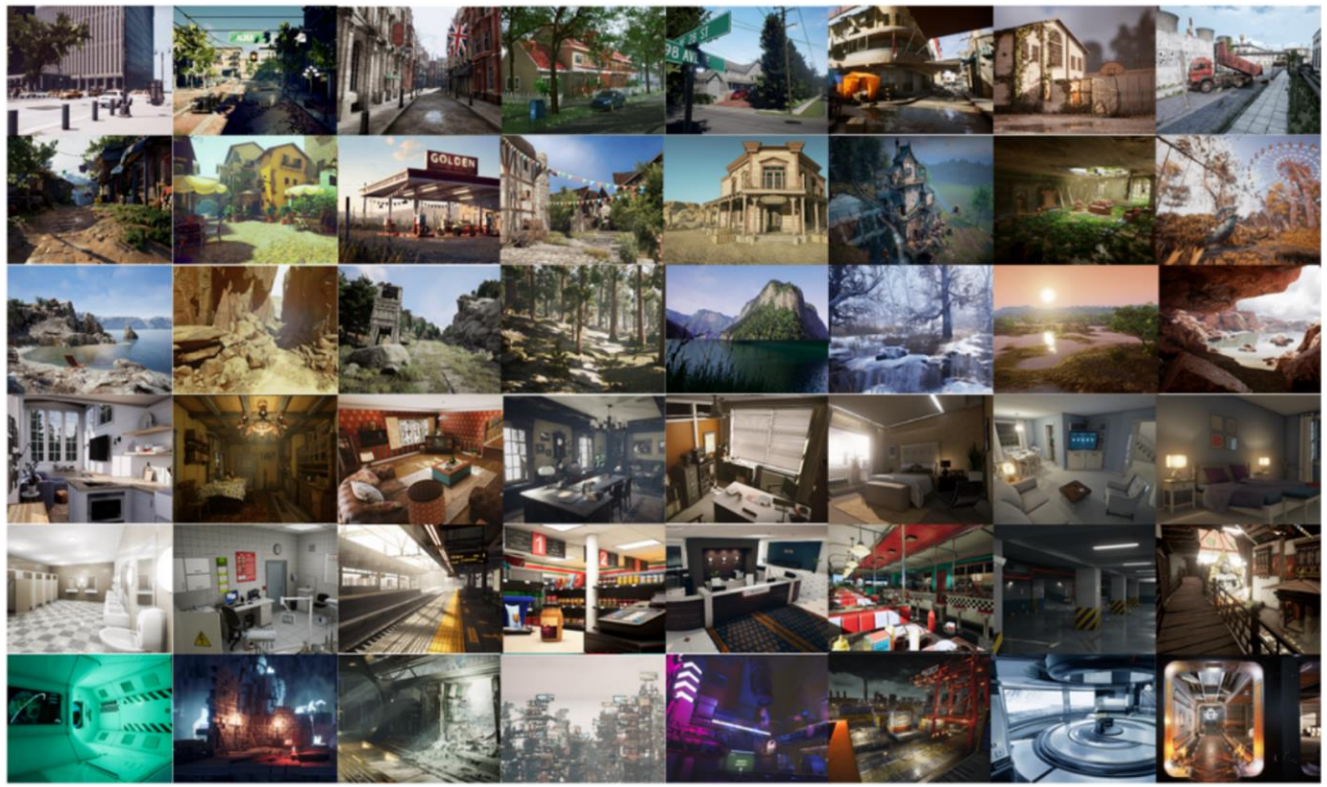




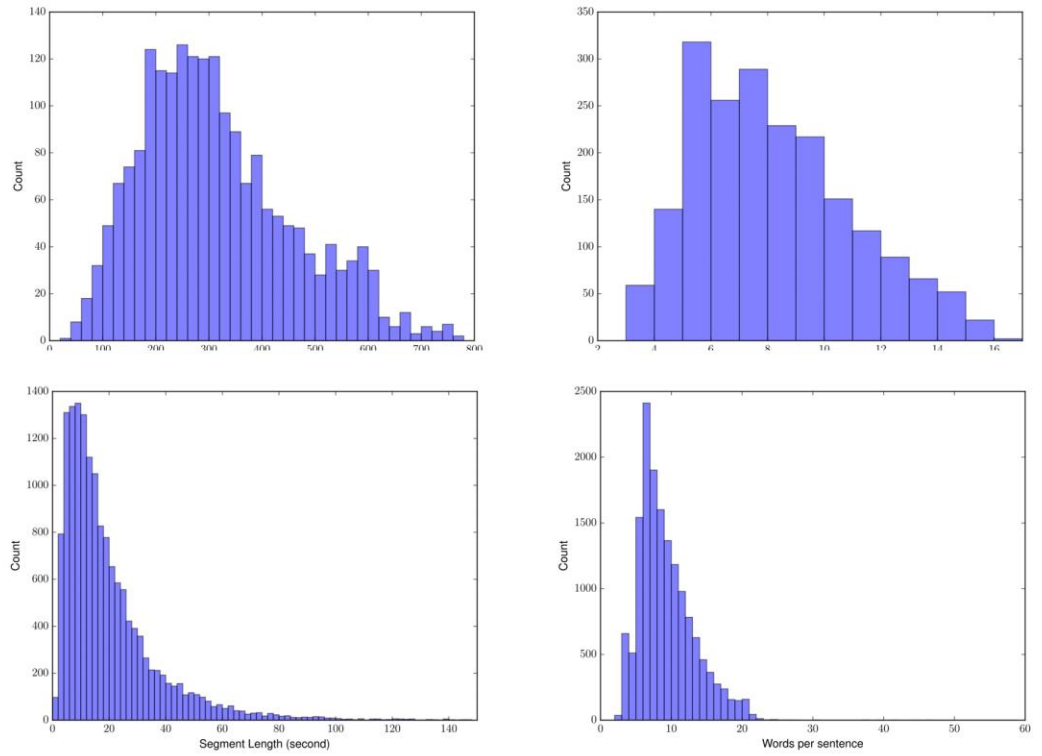
2022 Microsoft Azure Virtual Hackathon

3. 기술 세부 내용

3) Datasets: TartanAir, Youcook2, and some Vlog things from YouTube



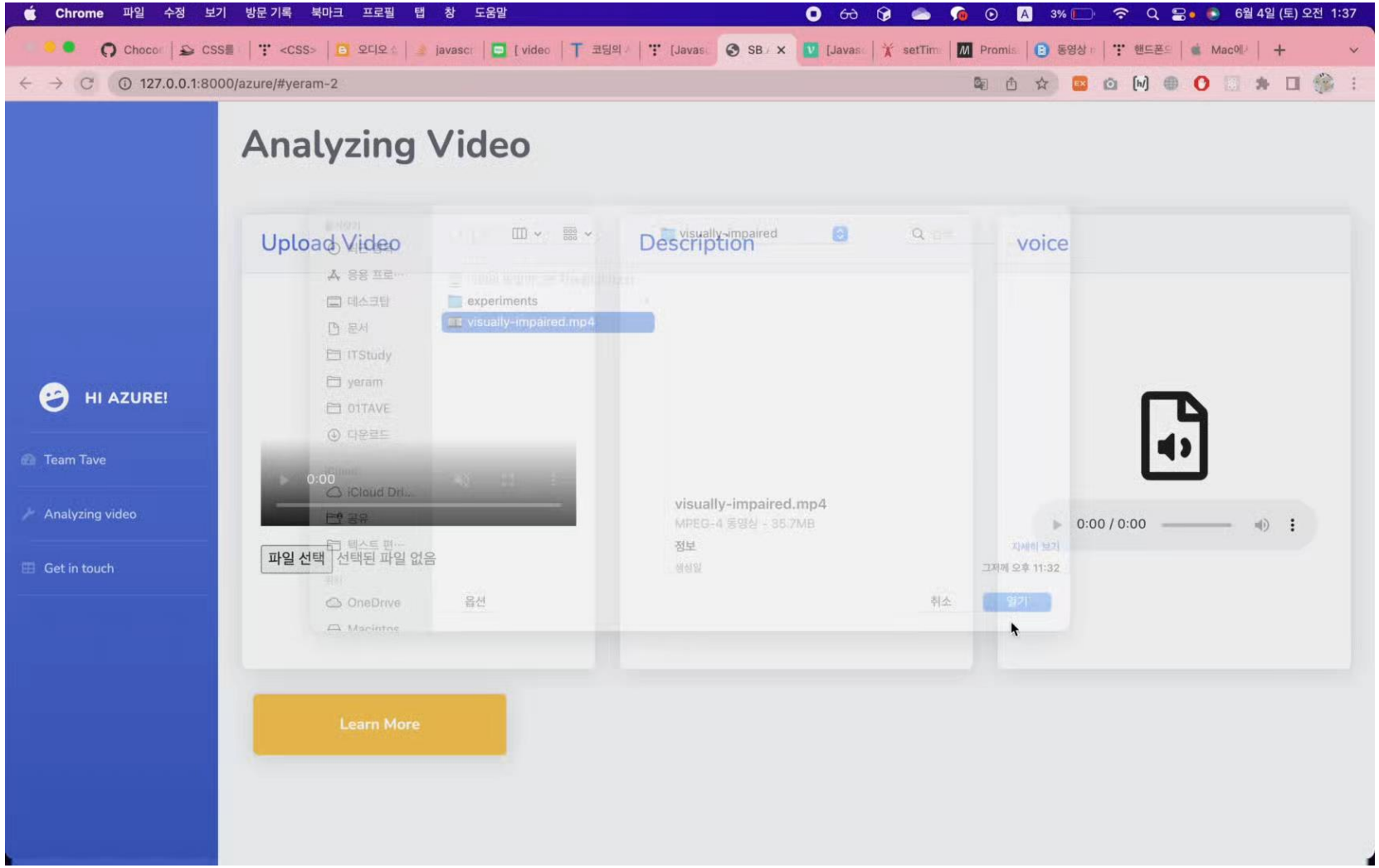
TartanAir dataset (Azure open dataset)



Youcook2 dataset

2022 Microsoft Azure Virtual Hackathon

4. 데모: 제작중 🔧 ^^;



2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

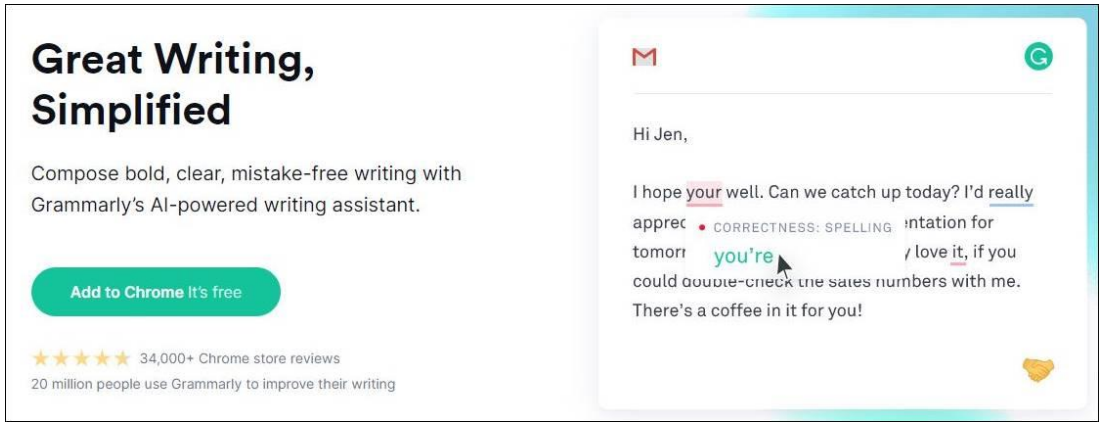
5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

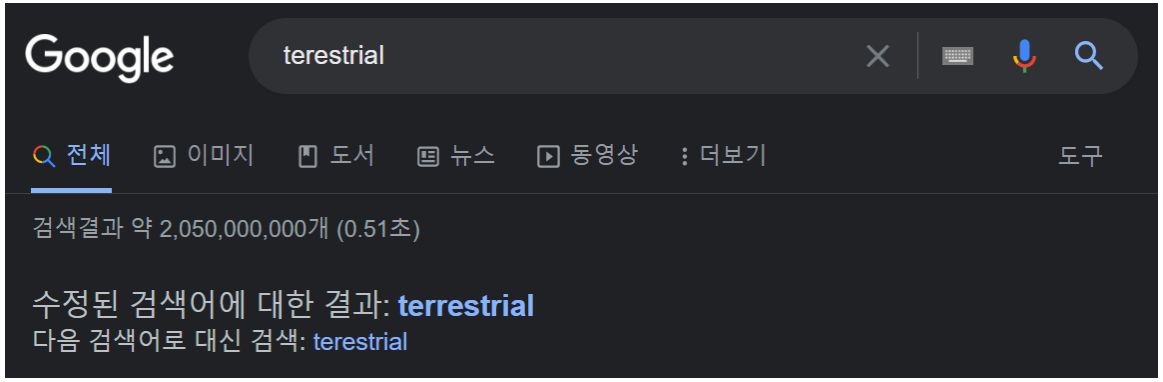
Grammatical Error Correction

1. Grammatical Error Correction (GEC)

24. 다음 글의 밑줄 친 부분 중, 어법상 틀린 것은? [3점]

In some communities, music and performance have successfully transformed whole neighborhoods as ① profoundly as The Guggenheim Museum did in Bilbao. In Salvador, Brazil, musician Carlinhos Brown established several music and culture centers in formerly dangerous neighborhoods. In Candeal, ② where Brown was born, local kids were encouraged to join drum groups, sing, and stage performances. The kids, energized by these activities, ③ began to turn away from dealing drugs. Being a young criminal was no longer their only life option. Being musicians and playing together in a group looked like more fun and was more ④ satisfying. Little by little, the crime rate dropped in those neighborhoods; the hope returned. In another slum area, possibly inspired by Brown's example, a culture center began to encourage the local kids to stage musical events, some of ⑤ them dramatized the tragedy that they were still recovering from.





Grammatical Error Correction

1. Grammatical Error Correction (GEC)

1) GEC is the task of fixing grammatical errors in text, such as typos, tense and article mistakes

2) Training a model for GEC requires a set of labeled *(ungrammatical / grammatical)* sentence pairs, which are expensive to obtain

(1) She like cats.

(2) Nothing is absolute right or wrong. (absolutely)

(3) One option to moving toward both biodiversity and terestrial food supply goals are to produce greater yield from less land

(1) She likes cats.

(2) Nothing is absolutely right or wrong.

(3) One option for moving toward both biodiversity and terrestrial food supply goals is to produce greater yields from less land

Grammatical Error Correction

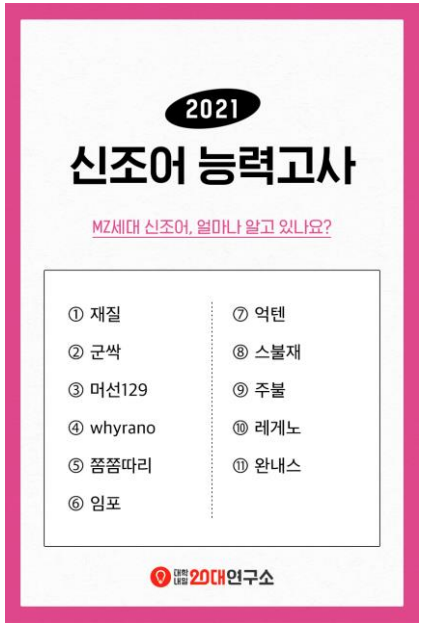
2. Challenges

1) Due to the unrestricted mutability of language, it is hard to design a model that is capable of correcting all possible errors made by non-native learners, especially when error patterns in new text are not observed in training data.

2) Unlike machine translation, a large amount of annotated ungrammatical texts and their corrected counterparts are not available.

3) The artificially generated data cannot precisely capture the error distribution in real erroneous data.

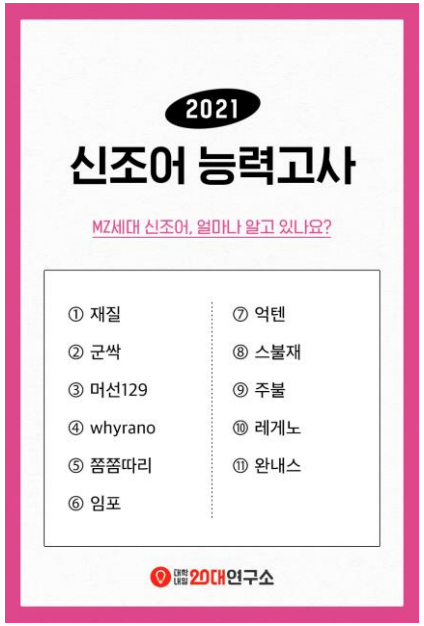
e.g.) We don't use *"a am I boy"* (*"I am a boy"*)



Grammatical Error Correction

2. Challenges

1) Due to the unrestricted mutability of language, it is hard to design a model that is capable of correcting all possible errors made by non-native learners, especially when error patterns in new text are not observed in training data.



✓ 2) Unlike machine translation, a large amount of annotated ungrammatical texts and their corrected counterparts are not available.

✓ 3) The artificially generated data cannot precisely capture the error distribution in real erroneous data.

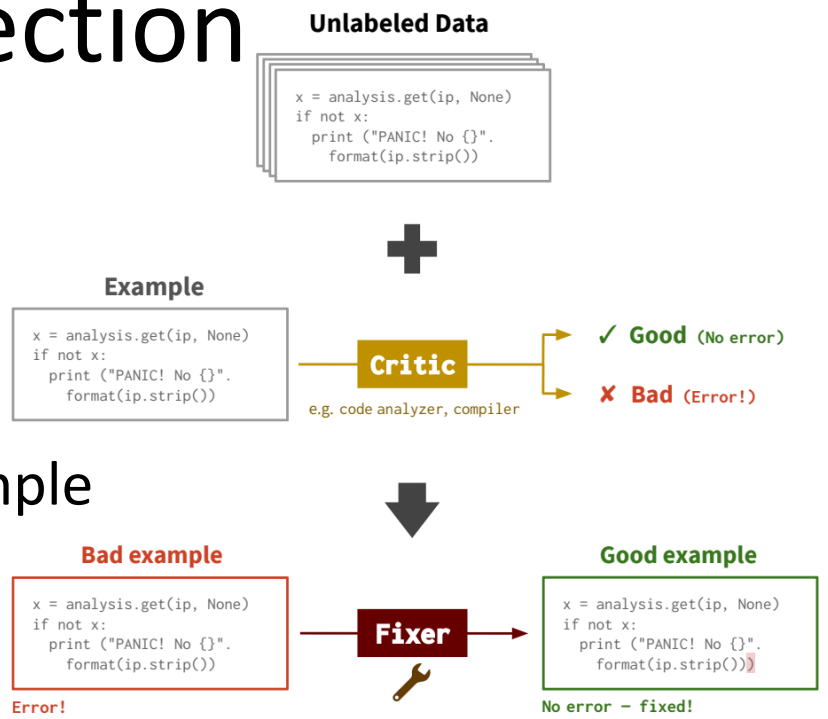
e.g.) We don't use *"a am I boy"* (*"I am a boy"*)

https://arxiv.org/pdf/2106.06600.pdf

Grammatical Error Correction

3. Break-It-Fix-It (BIFI) Framework

- They apply BIFI (Yasunaga et al., ICML 2021) to GEC task
- BIFI consists of (1) Breaker, (2) Fixer, and (3) Critic
- Breaker: Generate realistic bad example from good example
- Fixer: Converts a bad example into a good one
- Critic: Check fixer's output on real bad inputs



4. LM-Critic

- However, a "perfect" critic that returns whether an example is good or bad doesn't exist
- They leverage a pretrained language model (LM) in defining an LM-Critic, which judge a sentence to be grammatical if the LM assigns it a higher probability than its local perturbations

Problem Setup

- Notation

x_{bad}: Ungrammatical sentence
x_{good}: Grammatical version of x_{bad}
f: A GEC model (a.k.a. Fixer)

$D_{pair} = \{(x_{bad}^{(i)}, x_{good}^{(i)})\}$: A paired dataset
$labeled$: the pairs are human-annotated
$unlabeled$: A set of raw sentences $D_{unlabel} = \{x^{(i)}\}$

$$critic\ c\colon c(x) = \begin{cases} 1\ if\ x\ is\ good \\ 0\ if\ x\ is\ bad \end{cases}$$

Problem Setup

- Notation

x_{bad}: Ungrammatical sentence
x_{good}: Grammatical version of x_{bad}
f: A GEC model (a.k.a. Fixer)

$D_{pair} = \{(x_{bad}^{(i)}, x_{good}^{(i)})\}$: A paired dataset
$labeled$: the pairs are human-annotated
$unlabeled$: A set of raw sentences $D_{unlabel} = \{x^{(i)}\}$

$$critic\ c: c(x) = \begin{cases} 1\ if\ x\ is\ good \\ 0\ if\ x\ is\ bad \end{cases}$$

Given $D_{unlabel}$ and LM, which returns a probability distribution $p(x)$ over sentence x, we can define the critic and use that to the fixer

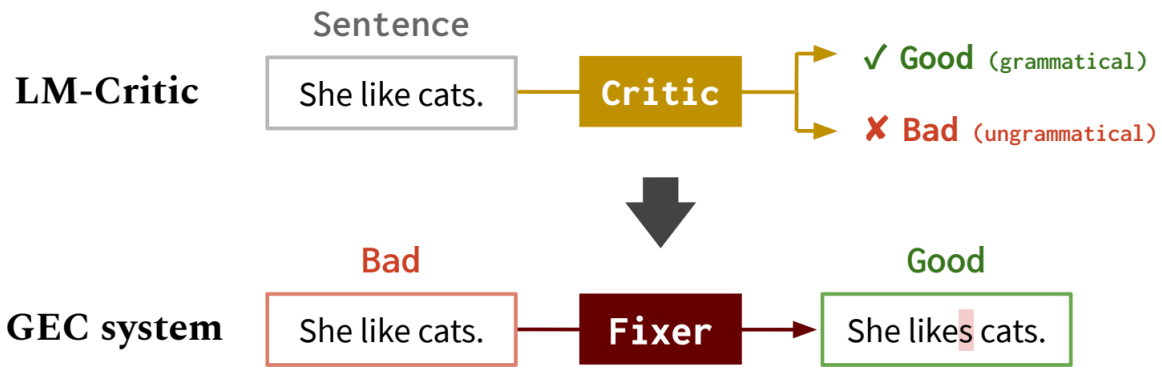
Method: LM-Critic

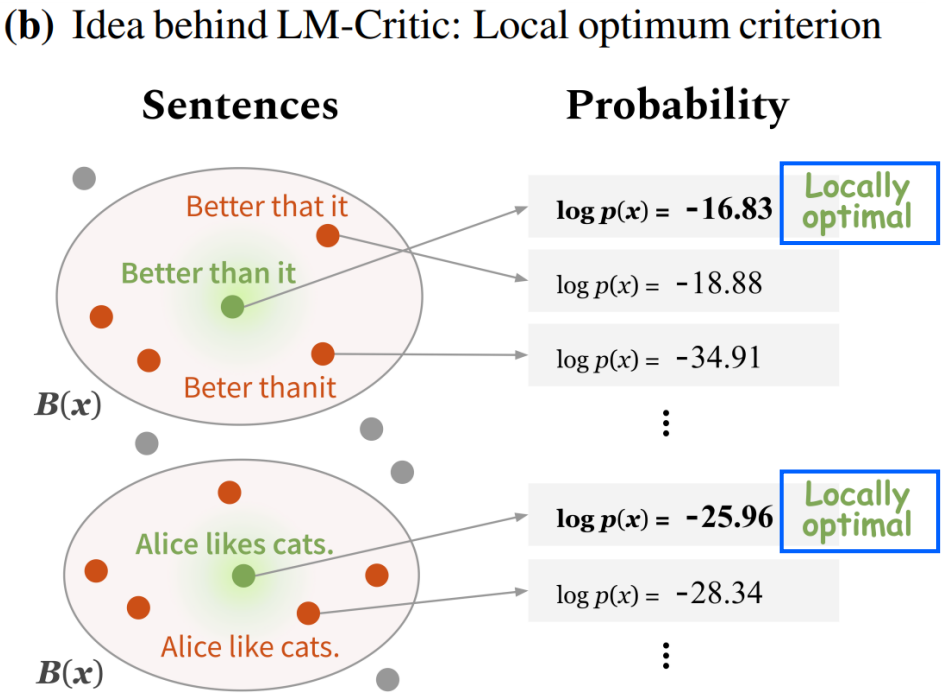
- Criterion

It deems a sentence to be good if it has the highest probability within its local neighborhood (local optimum criterion)

- Implementation

(1) A pretrained LM, and (2) perturbation function





Method: LM-Critic

1. Local Optimum Criterion of Grammaticality

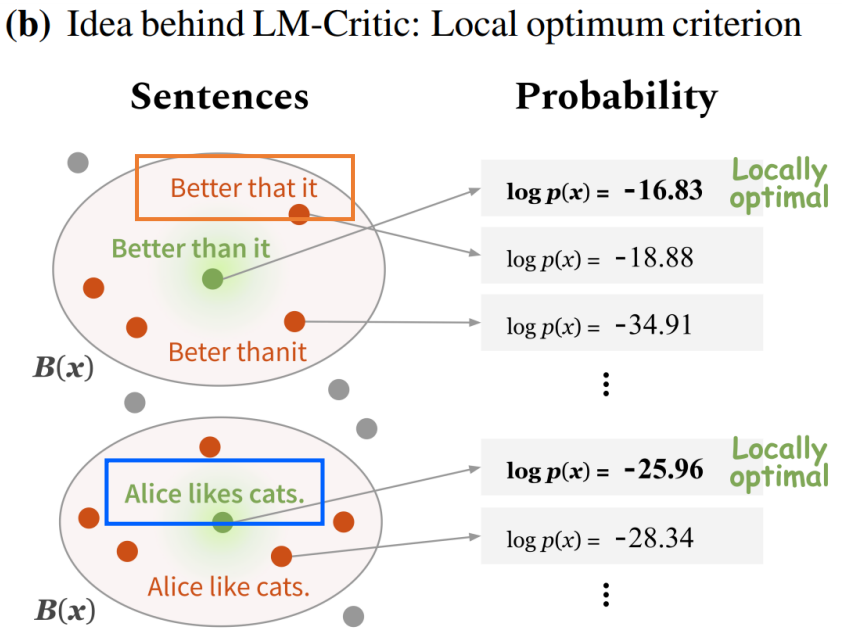
(1) Starting point

- Find a threshold (δ) for the absolute probability and let the critic be:

$$\text{AbsThr-Critic}(x) = \begin{cases} 1 & \text{if } p(x) > \delta \\ 0 & \text{otherwise.} \end{cases}$$

- However, it doesn't work in practice

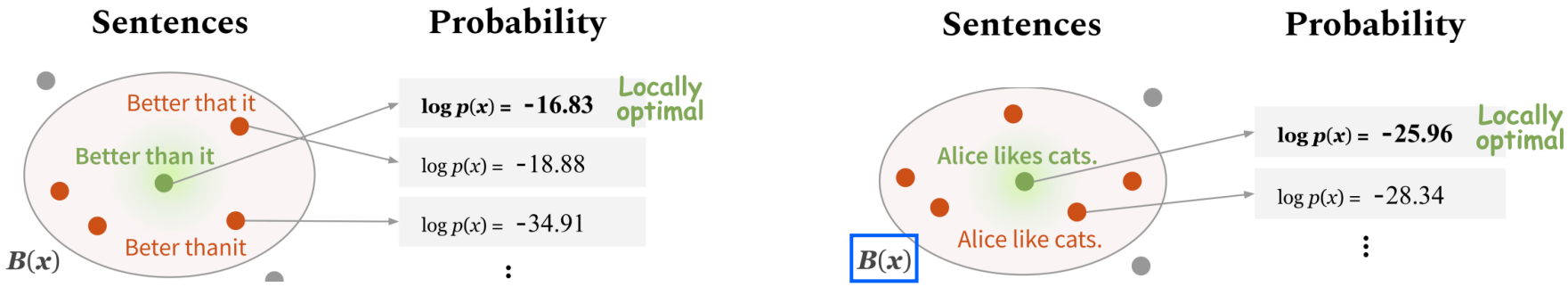
e.g.) $\log p(\text{"Alice likes cats"}) < \log p(\text{"Better that it"})$



Method: LM-Critic

1. Local Optimum Criterion of Grammaticality

(2) LM-Critic compares sentences with the same intended meaning



Local neighborhood (meaning-preserving)

(3) Local optimum criterion of grammaticality:

$$x \text{ is grammatical iff } x = \underset{x' \in B(x)}{\operatorname{argmax}}\, p(x').$$

Method: LM-Critic

2. Implementation of LM-Critic

- Obtaining ground-truth local neighborhood $B(x)$ is difficult → samples $\hat{B}(x)$

$$\text{LM-Critic}(x) = \begin{cases} 1 & \text{if } x = \underset{x' \in \hat{B}(x)}{\text{argmax}}\, p(x') \\ 0 & \text{otherwise.} \end{cases}$$

- There are three decisions for implementing LM-Critic:

(1) Choice of a pretrained LM
(2) Perturbation function b
(3) Sampling method of perturbations

Method: LM-Critic

2. Implementation of LM-Critic

(1) Choice of a pretrained LM

- GPT2 (#: 117M)
- GPT2-medium (#: 345M)
- GPT2-large (#: 774M)
- GPT2-xl (#: 1.6B)

- The LMs were trained on a large set of web text (40GB)

Method: LM-Critic

2. Implementation of LM-Critic

(2) Perturbation function b

- **ED1**: Edit-distance one perturbation in the character space (e.g. insert, delete, replace, and swap two adjacent characters)

- **ED1 + Word-level heuristics (all)**: ED1 can't fully cover word-level errors. It includes heuristics for word-level perturbations based on its dictionary

- **ED1 + Word-level heuristics**: It removes some heuristics that alter the meaning of the original sentence

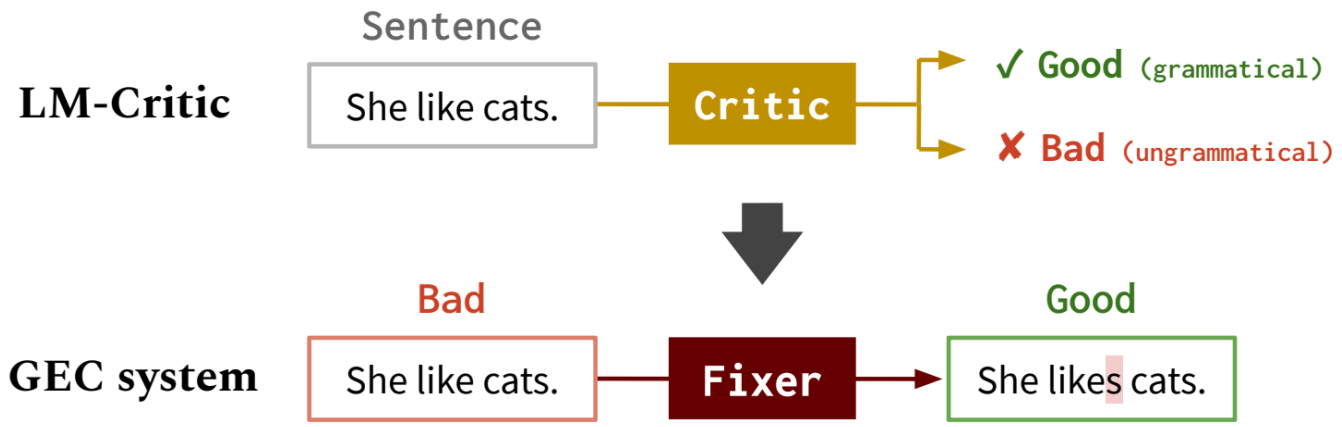
  e.g.) deleting / inserting "not"

Method: LM-Critic

2. Implementation of LM-Critic

(3) Sampling method of perturbations

- Random sampling with sizes of 100, 200, and 400
- They obtain samples from $b(x)$ to be $\hat{B}(x)$



Method: LM-Critic

3. Empirical Analysis

- Simple check: To make sure that LM's probability score correlates with grammaticality

(1) Evaluation data

- They prepare a simple evaluation data consisting of (x_{bad}, x_{good})
- They combine the dev sets of multiple GEC benchmarks, including *GMEG-wiki, GMEG-yahoo, BEA-2019*
- #: about 600

Method: LM-Critic

3. Empirical Analysis

(2) Analysis of LM probability

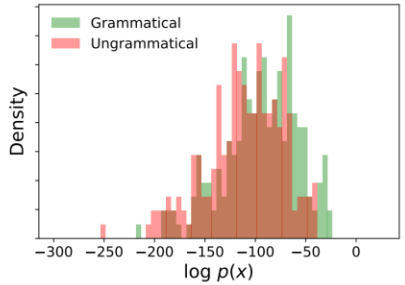


Figure 2: Probability of grammatical (green) and ungrammatical (red) sentences, computed by a pretrained LM (GPT2).

Pretrained LM	How often $p(x_{\text{bad}}) < p(x_{\text{good}})$?
GPT2	94.7%
GPT2-medium	95.0%
GPT2-large	95.9%
GPT2-xl	96.0%

Table 1: How well sentence probability returned by pretrained LMs correlates with grammaticality empirically.

- Remaining pairs (5.3%)

Examples of $p(x_{\text{bad}}) > p(x_{\text{good}})$
(Comma)
x_{bad}: The video was filmed on January 22 and is set to premiere on February 22.
x_{good}: The video was filmed on January 22, and is set to premiere on February 22.
(Quotation)
x_{bad}: Uprising is a 1980 roots reggae album by Bob Marley & The Wailers.
x_{good}: "Uprising" is a 1980 roots reggae album by Bob Marley & The Wailers.
(British spelling)
x_{bad}: The blast could be heard across the whole city centre.
x_{good}: The blast could be heard across the whole city center.
Examples of $p(x') > p(x_{\text{good}}), x' \in \hat{B}(x_{\text{good}})$
(Singular/plural)
x': They are affiliated to either the state boards or to national education boards.
x_{good}: They are affiliated to either the state board or to national education boards.
(Tense)
x': As well as touring Europe, they tour with such acts as Green Day.
x_{good}: As well as touring Europe, they toured with such acts as Green Day.

Table 3: Failure cases of LM-Critic. (Top) GPT2 assigns a higher probability to bad sentences. (Bottom) our neighborhood function ("ED1 + word") includes sentences with a higher LM probability than the original good sentence.

Method: LM-Critic

3. Empirical Analysis

(3) Performance of LM-Critic (using evaluation set)

Perturbation	Recognize "Good"			Recognize "Bad"		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
ED1	58.7	90.1	63.1	78.8	36.8	64.2
ED1 + word(all)	69.7	10.2	32.2	51.5	95.5	56.7
ED1 + word	68.4	75.5	**69.7**	72.7	65.1	**71.1**

Sample size	Recognize "Good"			Recognize "Bad"		
	P	R	$F_{0.5}$	P	R	$F_{0.5}$
100	68.4	75.5	69.7	72.7	65.1	71.1
200	71.3	71.5	71.4	71.4	71.3	**71.4**
400	72.6	68.7	**71.8**	70.3	74.0	71.0

Pretrained LM	Recognize "Good"	Recognize "Bad"
	$F_{0.5}$	$F_{0.5}$
GPT2	69.7	71.1
GPT2-medium	69.9	71.0
GPT2-large	70.3	71.3
GPT2-xl	69.9	71.0

Table 2: **Performance of LM-Critic**, when using different choices of a perturbation function, sample size, and pretrained LM described in §3.2. **(Top)** We set the LM to be GPT2 and the perturbation sample size to be 100, and vary the perturbation function *b*. "ED1 + word" achieves the best $F_{0.5}$. Henceforth, we use this perturbation function. **(Middle)** We set the LM to be GPT2 and vary the perturbation sample size. Increasing the sampling size improves the performance slightly. **(Bottom)** We vary the LM. Increasing the LM size makes slight or no improvement in $F_{0.5}$ on the dataset we used.

Method: LM-Critic

4. Learning GEC with LM-Critic

- Initial fixer f_0 is trained on synthetic data (unsupervised setting) or labeled data (supervised setting)

(1) Apply the fixer f to the bad example D_{bad} (by human)
(2) They train a $breaker\ b$ on resulting paired data
(3) They apply the breaker to the good example D_{good}
(4) They finally train the fixer on the newly-generated paired data in (1) and (3)

- This cycle can be iterated to improve both fixer and breaker

Analysis

- Inference (Jeiyoon)

[original] Parliament House
[Round 1] Parliament House .
[HT_6.7] House Parliament House

[original] Eat food .
[Round 1] Eat food .
[HT_6.7] Eating food.

[original] Yours Sincerely .
[Round 1] Yours Sincerely .
[HT_6.7] Yours is Sincerely.

[original] And this is true .
[Round 1] And this is true .
[HT_6.7] And this is correct.

[original] Everything was dark .
[Round 1] Everything was dark .
[HT_6.7] Everything is dark.

[original] Hello friend ,
[Round 1] Hello friend .
[HT_6.7] Hello friends.

[original] Whilst , we recyling inorganic rubbish too .
[Round 1] Whilst we 're recyling inorganic rubbish too .
[HT_6.7] Whilst, we reclining inorganic rubbish too.

[original] It is small but comfortable and famous for his big burguers .
[Round 1] It is small but comfortable and famous for its big burguers .
[HT_6.7] It is small but comfortable and famous for it's big burgers.

[original] Patras is a very beautiful city and for her culture , and for her monuments .
[Round 1] Patras is a very beautiful city for its culture , and for her monuments .
[HT_6.7] Patras is a very beautiful city and for it's culture, and for it's monuments.

Analysis

- Any drawbacks?: Assumptions

(1) They excludes an ungrammatical sentence which may have no correction

e.g.) “asdfghgfdsa”

(2) They also didn’t consider multiple corrections

e.g.) “The cat sleep” → “The cat sleeps”? or “The cat slept”?

(3) LM-Critic

```
Loaded gpt2
Enter a sentence: I am a boy
I am a boy
Good! Your sentence log(p) = -16.902
```

```
Enter a sentence: I am an boy
I am an boy
Bad! Your sentence log(p) = -23.470
```

```
Enter a sentence: I a am boy
I a am boy
Bad! Your sentence log(p) = -33.839
```

```
Enter a sentence: boy am a I
boy am a I
Good! Your sentence log(p) = -33.335
```

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization [컨퍼런스 논문으로 작성중]

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

Video Summarization

1. Video Summarization

Video Summarization

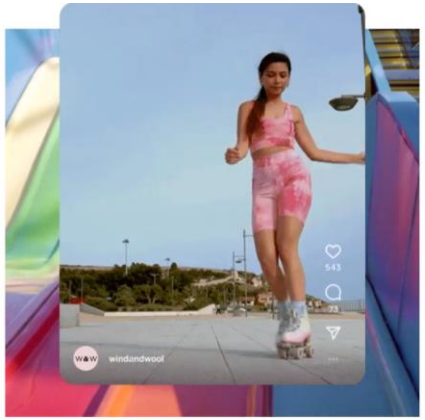
2. Increased Preference for Short-Form Videos

Shorts

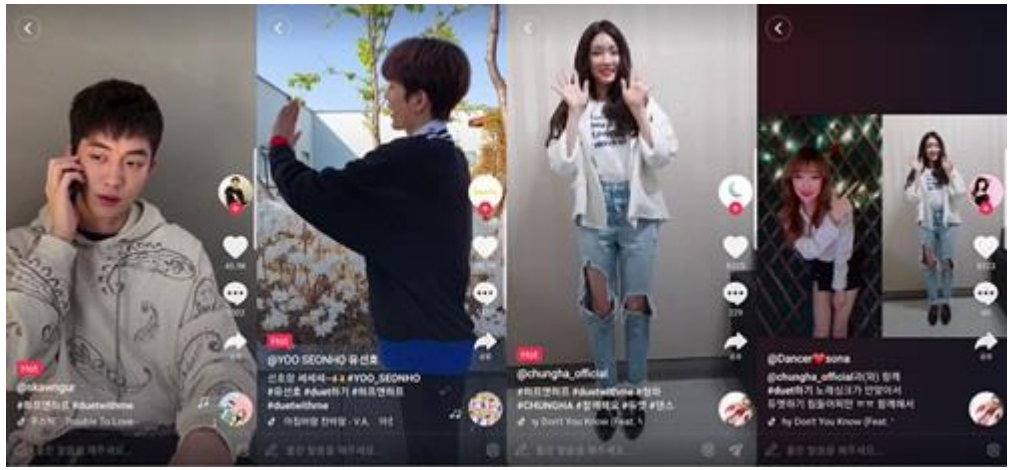
Reels

릴스

Instagram의 모든 곳에서 쉽게 발견할 수 있는 짧고 재미있는 동영상으로 사람들의 관심을 사로잡으세요.



TikTok

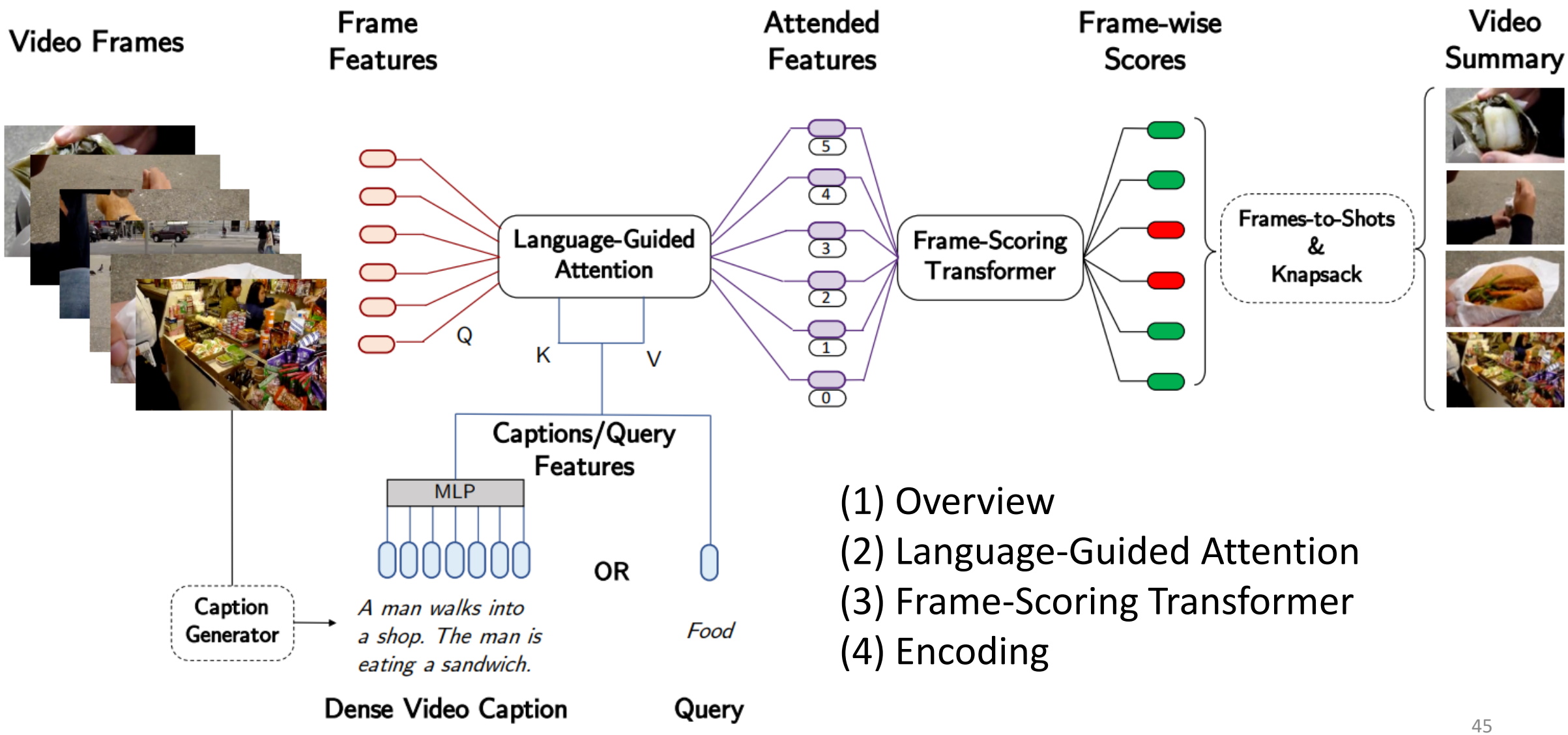


Video Summarization

3. Language-Guided Video Summarization

- The importance of scenes in a video is subjective
- Users should have the option of customizing the summary to specify what is important to them

Language-Guided Video Summarization

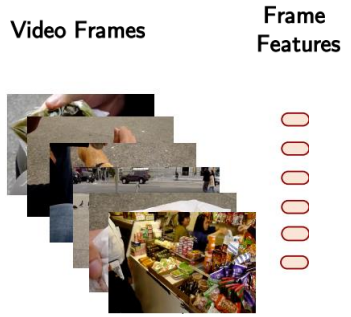


(1) Overview
(2) Language-Guided Attention
(3) Frame-Scoring Transformer
(4) Encoding

Language-Guided Video Summarization

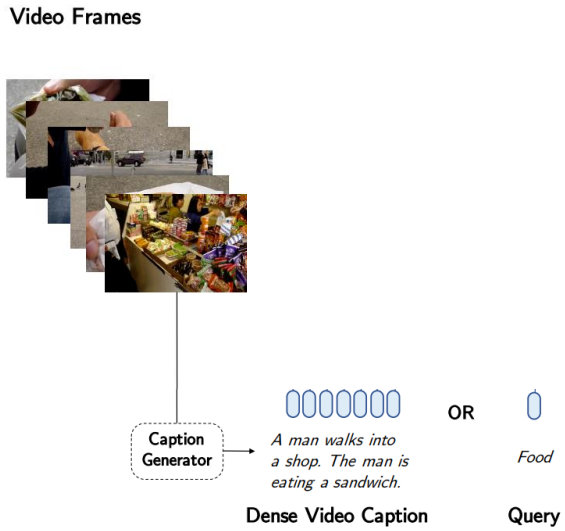
1. Overview

(1) $\boldsymbol{F_i}$ is frames, $i \in [1, \dots, N]$



(2) A pretrained network $\boldsymbol{f_{img}}$ embeds the frames

(3) A pretrained network $\boldsymbol{f_{txt}}$ embeds the query or dense video caption $C_j, j \in [1, \dots, M]$, where M is a sentence

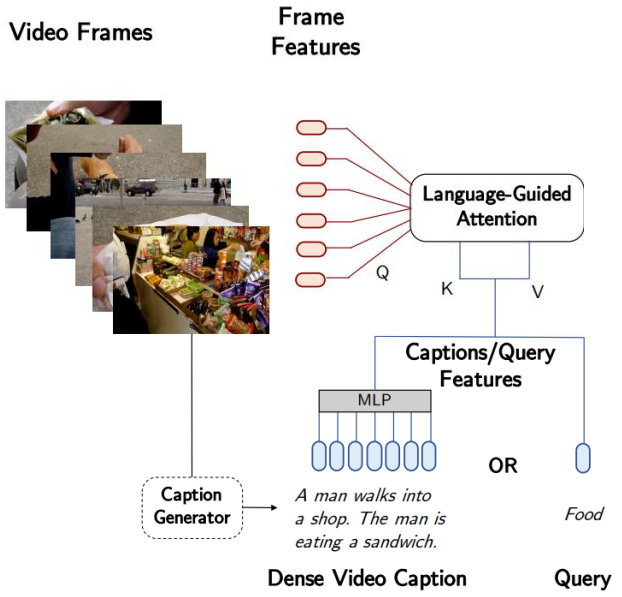


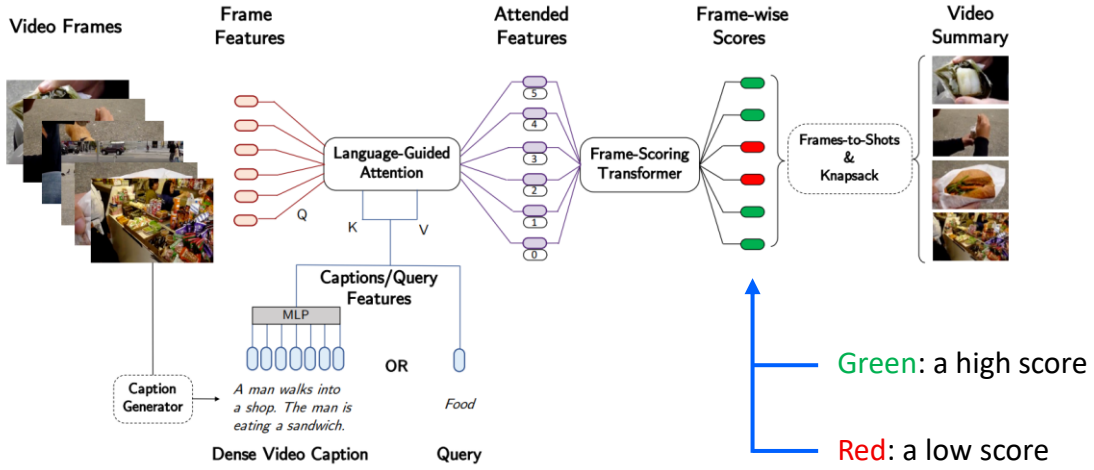
Language-Guided Video Summarization

1. Overview

(4) We compute language attended image embeddings using learned Language-Guided Multi-head Attention: $f^{*}_{img\_txt}$

(5) Finally, we train a Frame-Scoring Transformer which assigns scores to each frame in the video





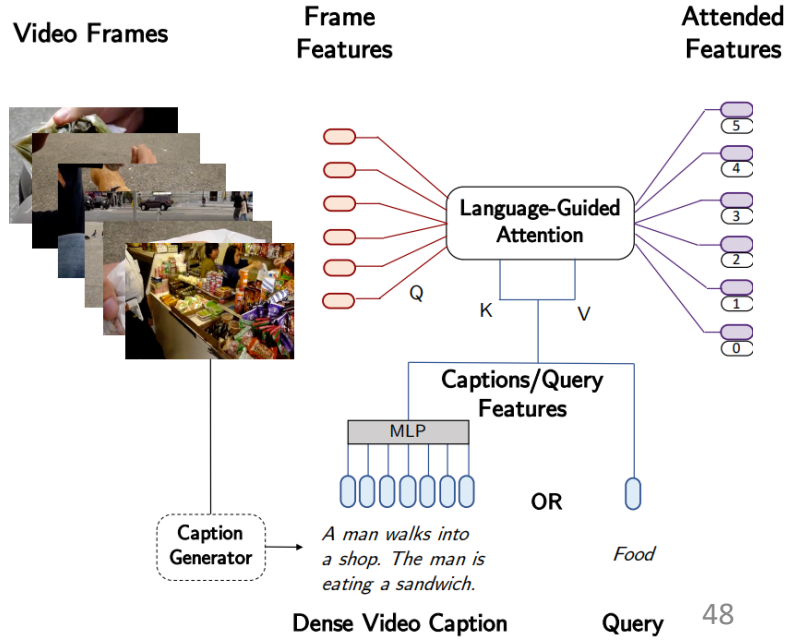
Language-Guided Video Summarization

2. Language-Guided Attention

- Using a single attention head does not suffice as the goal is to allow all captions to attend to all frames in the video

- We set Query Q, Key K, and Value V as follows:

$$
\begin{aligned}
Q &= f_{img}(F_i), \text{where } i \in [1, \dots N], \\
K, V &= f_{txt}(C_j), \text{where } j \in [1, \dots M], \\
\text{Language - Guided Attn.}(Q, K, V) &= \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O, \\
\text{where head}_\text{i} &= \text{Attention}(QW_i^Q, KW_i^K, VW_i^V) \\
\text{and Attention}(Q, K, V) &= \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V
\end{aligned}
$$



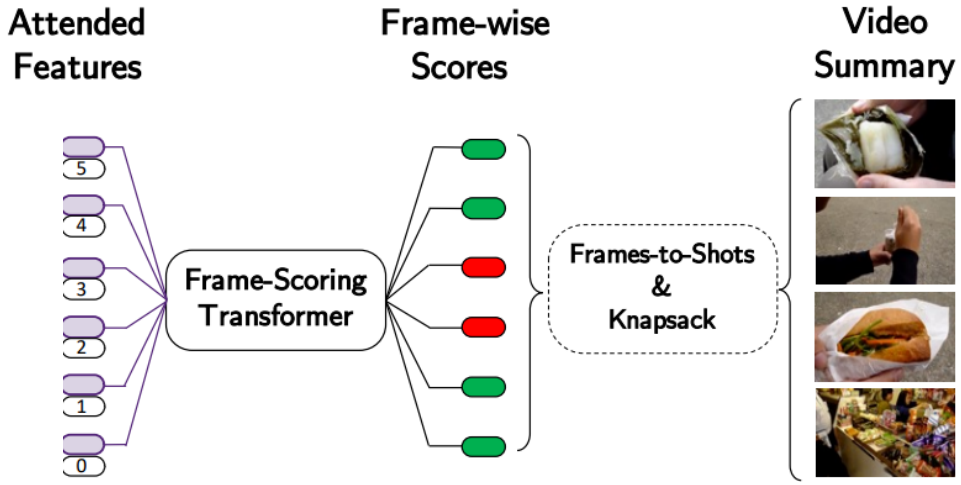
Language-Guided Video Summarization

3. Frame-Scoring Transformer

- It doesn't include redundant information, e.g., several key shots from the same event

- Frame-Scoring Transformer takes image-text representation as input and outputs one score per frame

- It uses positional encoding to insert information about the relative positions of the tokens in the sequence



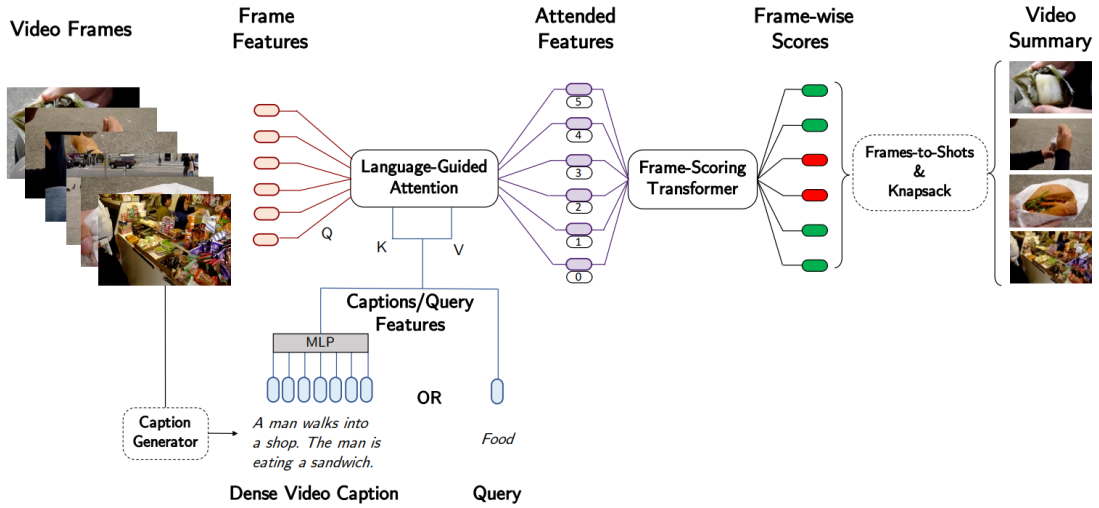
Language-Guided Video Summarization

4. Encoding

(1) Image Encoding (f_{img}): *GoogleNet*, *ResNet*, and *CLIP model*

(2) Text Encoding (f_{txt}): CLIP (ViT and RN101) model

- First, embeds each sentence of the caption using the text encoder f_{txt}
- And then, concatenates and fuses using a multi-layer perceptron (MLP)



Learning

1. Supervised setting

- Classification loss
- Reconstruction loss
- Diversity loss

2. Unsupervised setting

- Reconstruction loss
- Diversity loss

Learning

1. Classification Loss

- Weighted binary cross entropy loss ($\mathcal{L}_c$) for classifying each frame:

$$\mathcal{L}_c = -\frac{1}{N}\sum_{i=1}^{N} w^*[x_i^* \log(x_i)] + (1 - w^*)[(1 - x_i^*)\log(1 - x_i)],$$

where x_i^* is the ground-truth label of the i-th frame

N is the total number of frames in the video

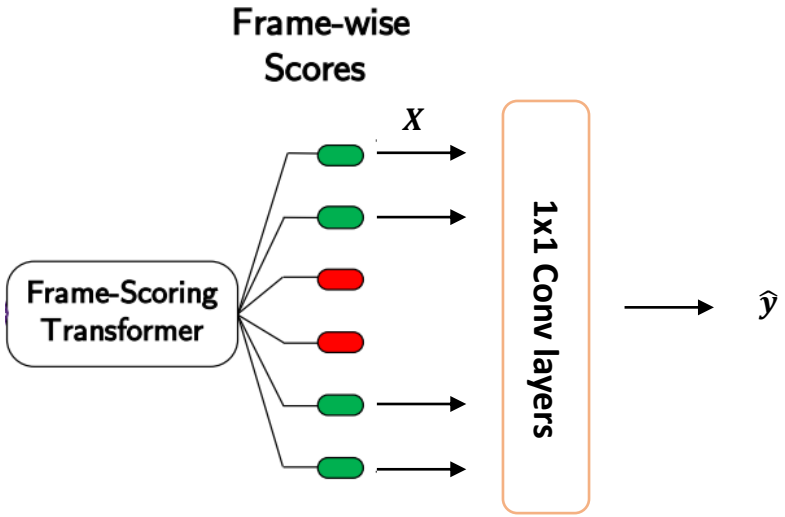
w^* is the weight assigned to the class x_i^*, which is set to $\frac{\#keyframes}{N}$ if x_i^* is a keyframe and $1 - \frac{\#keyframes}{N}$ if x_i^* is a background frame

Learning

2. Reconstruction Loss

- $\mathcal{L}_r$ is defined as the mean squared error between the reconstructed features and the original features corresponding to the selected keyframes, such that:

$$\mathcal{L}_r = \frac{1}{X} \sum_{i \in X} ||\mathbf{x}_i - \hat{\mathbf{y}}_i||_2, \text{ where } \hat{\mathbf{y}} \text{ denotes the reconstructed features.}$$



Learning

3. Diversity Loss

- To enforce diversity among selected keyframes.

$$\mathcal{L}_d = \frac{1}{X(X-1)} \sum_{i \in X} \sum_{j \in X, j \neq i} \frac{\hat{\mathbf{y}}_i \cdot \hat{\mathbf{y}}_j}{||\hat{\mathbf{y}}_i||_2 \cdot ||\hat{\mathbf{y}}_j||_2},$$

where $\hat{\mathbf{y}}_i$ and $\hat{\mathbf{y}}_j$ denote the reconstructed feature vectors of the i-th and j-th node.

$$\mathcal{L}_{sup} = \alpha \cdot \mathcal{L}_c + \beta \cdot \mathcal{L}_d + \lambda \cdot \mathcal{L}_r$$

$$\mathcal{L}_{unsup} = \beta \cdot \mathcal{L}_d + \lambda \cdot \mathcal{L}_r$$

Experiments: Generic Video Summarization

- Results

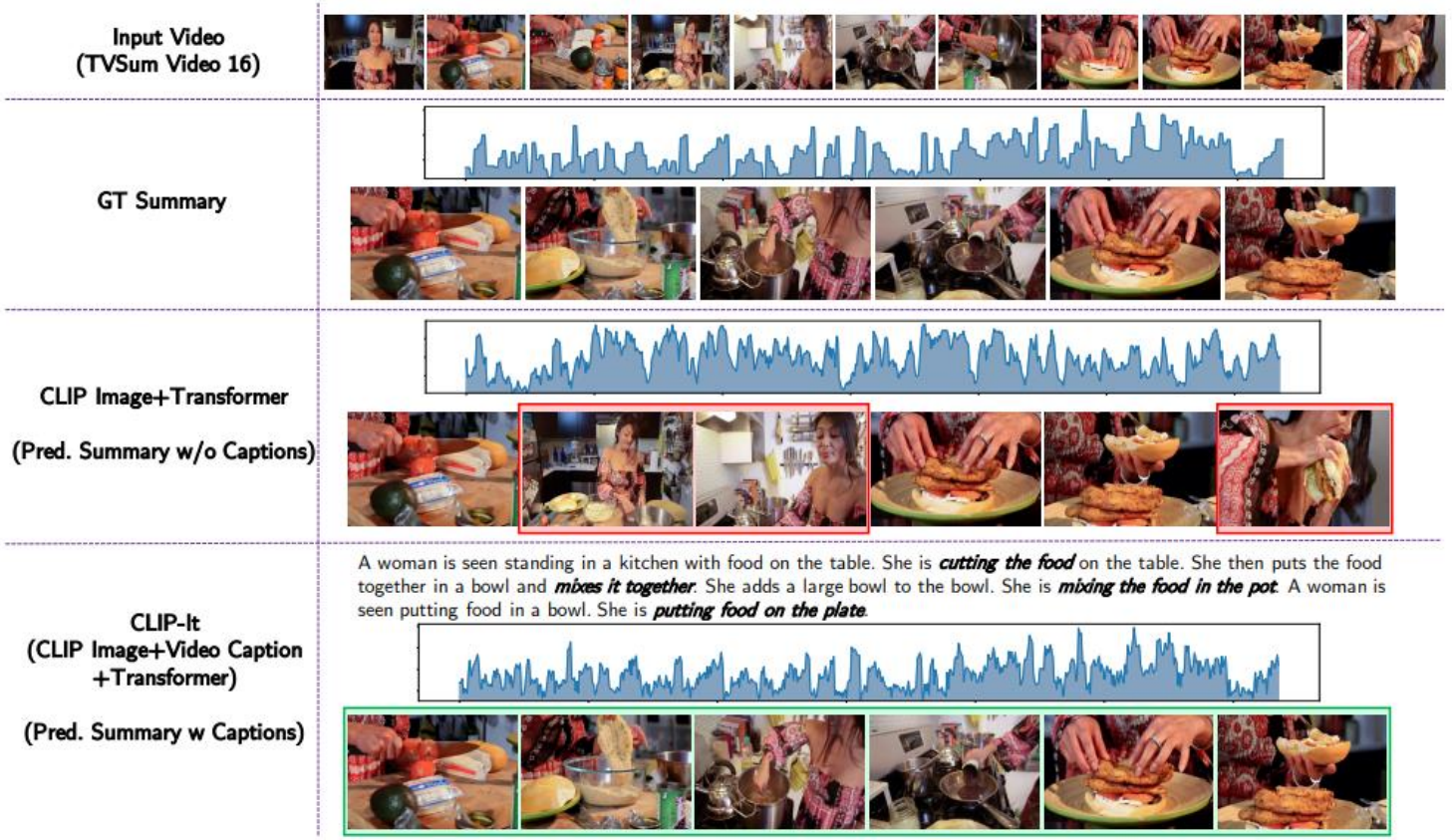


Figure 3: Comparison of ground-truth summary to results from CLIP-Image+Transformer and the full CLIP-It model (CLIP-Image+Video Caption+Transformer). The input is a recipe video. Without captions, the model assigns high scores to certain irrelevant frames such as scenes of the woman talking or eating which hurts the precision. With captions, the cross-attention mechanism ensures that frames with important actions and objects are assigned high scores.

Experiments: Generic Video Summarization

- Results

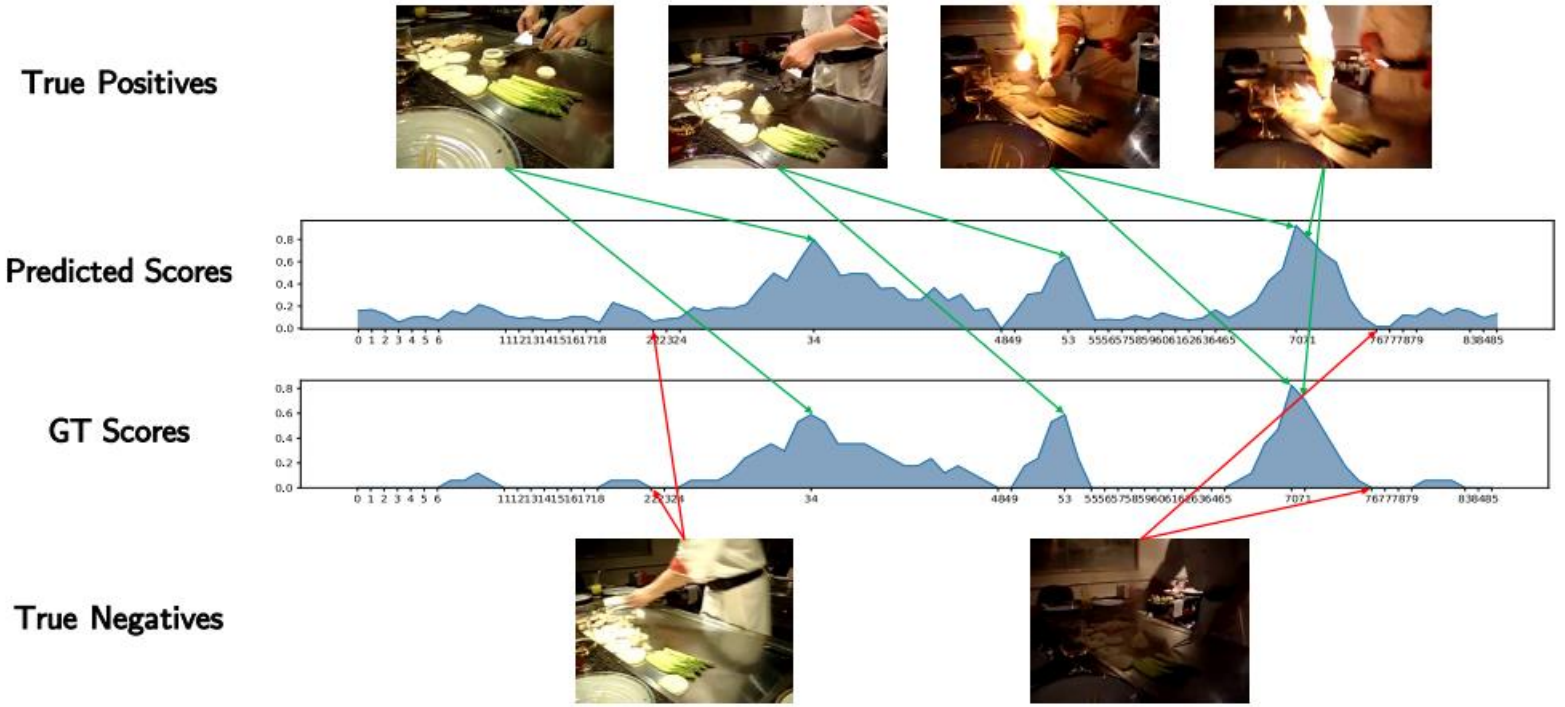


Figure 4: Qualitative result comparing the generic summary from CLIP-It with the ground-truth summary. The plots showing predicted and ground-truth frame-level scores are similar, indicating that frames that were given a high score in ground-truth were also assigned high scores by our model.

https://youtu.be/tmZuPvAtDdw

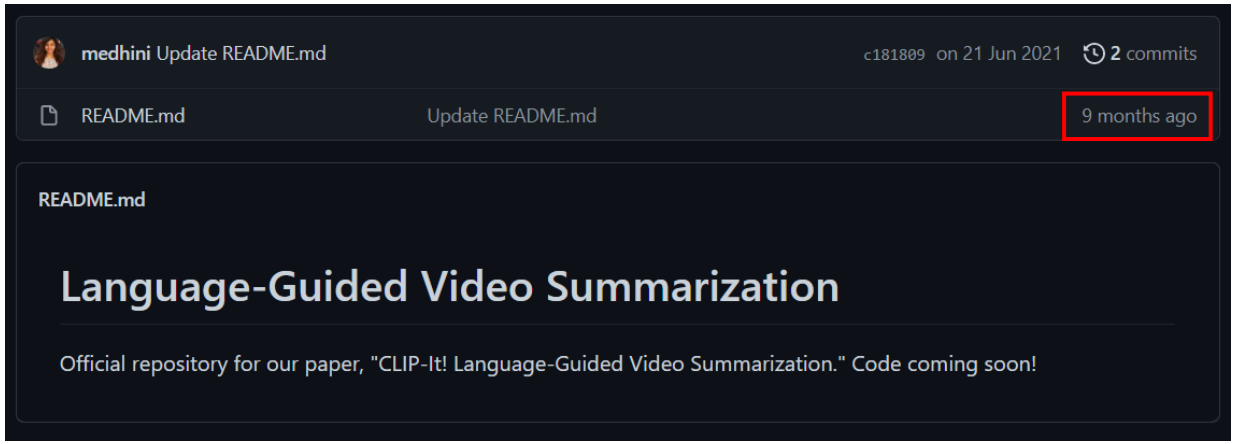
Experiments: Generic Video Summarization

- Results: “Infinite Challenge”

Analysis

1. Any drawbacks?

(1) No code is available



(2) Language-guided summarization?

(3) Modality?

드리고 싶은 말씀

https://youtu.be/IMI_kTupBJA

드리고 싶은 말씀

1. 항상 감사한 마음 갖기

2022.01 - Now

1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

https://youtu.be/IMI_kTupBJA

드리고 싶은 말씀

1. 항상 감사한 마음 갖기

2022.01 - Now

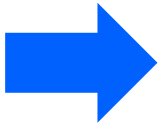
1. 3rd AI SPARK Challenge (2022)

2. 2022 Microsoft Azure Virtual Hackathon

3. Research: Grammatical Error Correction

4. Research: Video Summarization

5. NVIDIA DLI Ambassador:
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)



2022.01 - Now

1. 3rd AI SPARK Challenge (2022) [박제윤, 권기호, 이문기, 허주희]

2. 2022 Microsoft Azure Virtual Hackathon [박제윤, 임예람, 이문기, 허주희]

3. Research: Grammatical Error Correction [박제윤, 홍창표, 오영대]

4. Research: Video Summarization [박제윤, 권기호, 이찬희, 임희석]

5. NVIDIA DLI Ambassador: [박제윤, Mr.TBDs at NVIDIA]
   - Building Transformer-Based NLP Applications
   - (Teaching) Building Real-Time Video AI Applications (현대자동차 남양연구소)
   - (Teaching) Building Real-Time Video AI Applications (글로벌창업사관학교)

드리고 싶은 말씀

Paper: https://arxiv.org/pdf/2103.00020.pdf
Code: https://github.com/openai/CLIP

2. Multimodal

- Learning Transferable Visual Models From Natural Language Supervision (Radford et al., arxiv 2021)

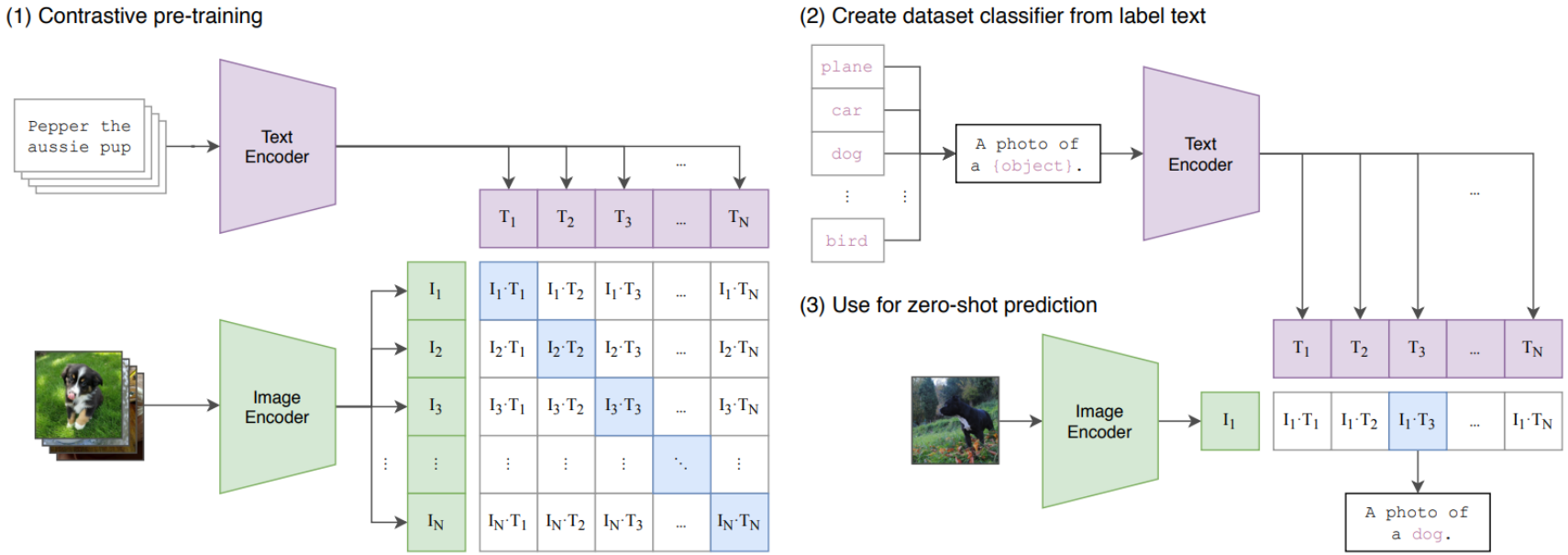


Figure 1. Summary of our approach. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.

https://youtu.be/IMI_kTupBJA

드리고 싶은 말씀

3. TAVE OB 초청강연 2022

Thank you

https://jeiyoon.github.io/

@cloudwantsasnack