Cross-media Structured Common Space for Multimedia Event Extraction

Manling Li*, Alireza Zareian*, Qi Zeng, Spencer Whitehead, Di Lu,
Heng Ji, Shih-Fu Chang

ACL 2020

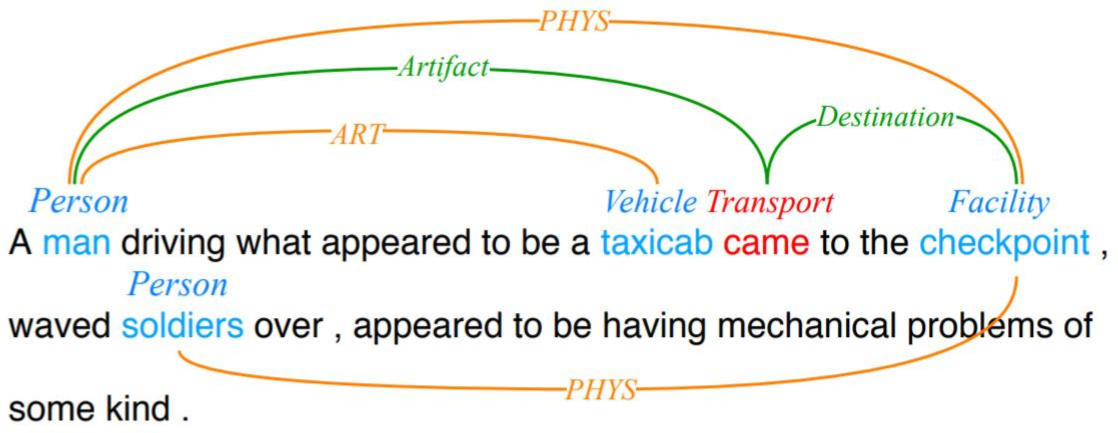
Introduction

- 33% of images in 100 randomly selected VOA articles contain visual objects that serve as event arguments and are not mentioned in the text.

Figure 1: An example of Multimedia Event Extraction. An event mention and some event arguments (*Agent* and *Person*) are extracted from text, while the vehicle arguments can only be extracted from the image.

Introduction

- Event extraction is independently studied in Computer Vision (CV) and Natural Language Processing (NLP), significantly different in terms of task definition, data domain, methodology, and terminology.
- ACE dataset:



Introduction

- Event extraction is independently studied in Computer Vision (CV) and Natural Language Processing (NLP), significantly different in terms of task definition, data domain, methodology, and terminology.
- ImSitu dataset:

Situations

Click image

feeding				
agent	food	source	eater	place
man	fish	hand	dolphin	pool

imSitu Dataset

verbs	504
images	126,102
situations per image	3
total annotations	1,481,851
unique entity types (>3)	11,538 (6,794)
unique roles (role types)	1,788 (190)
images per verb (range)	250.2 (200 - 400)
unique situations (>3)	205,095 (21,505)

Introduction

- Event extraction is independently studied in Computer Vision (CV) and Natural Language Processing (NLP), significantly different in terms of task definition, data domain, methodology, and terminology.

=> They propose a new task: **MultiMedia Event Extraction** (M^2E^2):

+ An evaluation dataset: 245 fully annotated news articles.

+ A new method for the task that learns a structured multimedia embedding space: Weakly Aligned Structured Embedding (WASE).

M^2E^2 dataset

- 245 documents selected from 108,693 multimedia VOA articles.
- Contains 8 ACE types (i.e., 24% of all ACE types), mapped to 98 imSitu types (i.e., 20% of all imSitu types), encompassing 52% ACE events.
- Annotators: 8 with an Inter-Annotator Agreement score of 81.2%.
- This dataset is for **Evaluation Only**.

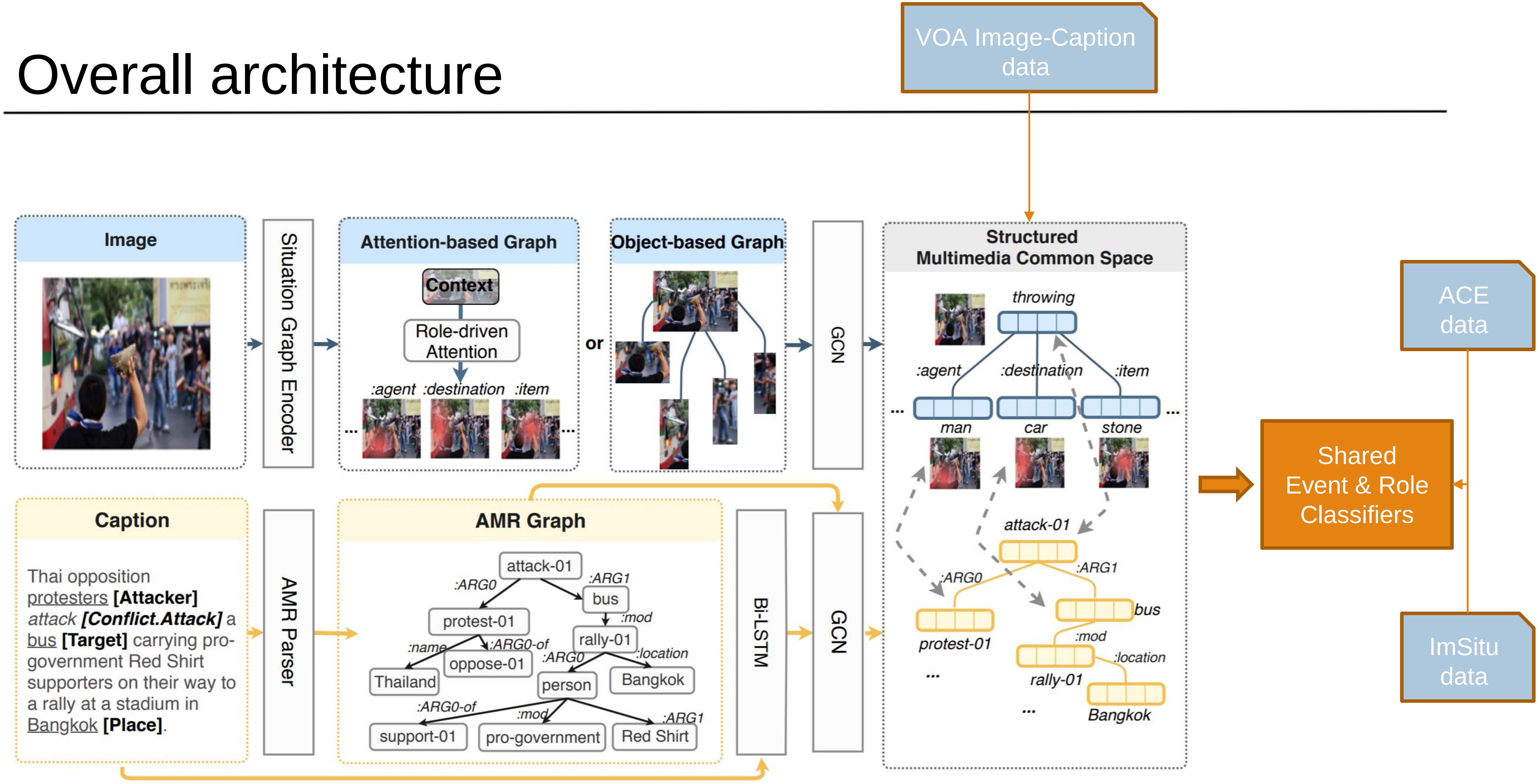
Event Type	Argument Role
Movement.Transport (223|53)	Agent (46|64), Artifact (179|103), Vehicle (24|51), Destination (120|0), Origin (66|0)
Conflict.Attack (326|27)	Attacker (192|12), Target (207|19), Instrument (37|15), Place (121|0)
Conflict.Demonstrate (151|69)	Entity (102|184), **Police** (3|26), **Instrument** (0|118), Place (86|25)
Justice.ArrestJail (160|56)	Agent (64|119), Person (147|99), **Instrument** (0|11), Place (43|0)
Contact.PhoneWrite (33|37)	Entity (33|46), **Instrument** (0|43), Place (8|0)
Contact.Meet (127|79)	Participant (119|321), Place (68|0)
Life.Die (244|64)	Agent (39|0), Instrument (4|2), Victim (165|155), Place (54|0)
Transaction. TransferMoney (33|6)	Giver (19|3), Recipient (19|5), Money (0|8)

Table 1: Event types and argument roles in M^2E^2, with expanded ones in bold. Numbers in parentheses represent the counts of textual and visual events/arguments.

Source		Event Mention		Argument Role	
sentence	image	textual	visual	textual	visual
6,167	1,014	1,297	391	1,965	1,429

Table 2: M^2E^2 data statistics.

Overall architecture



Text Event Extraction

- Use the CAMR parser *(Wang et al., 2015b,a, 2016)* to obtain an AMR graph for each sentence.
- Word representation = Glove + POS + NER + Position
- GCN: $$\boldsymbol{w}_i^{(k+1)} = f\Big(\sum_{j \in \mathcal{N}(i)} g_{ij}^{(k)} (\boldsymbol{W}_{E(i,j)} \boldsymbol{w}_j^{(k)} + \boldsymbol{b}_{E(i,j)}^{(k)})\Big), \quad (1)$$
- Prediction: $$P(y_e|w) = \frac{\exp\left(\boldsymbol{W}_e \boldsymbol{w}^{\mathbb{C}} + \boldsymbol{b}_e\right)}{\sum_{e'} \exp\left(\boldsymbol{W}_{e'} \boldsymbol{w}^{\mathbb{C}} + \boldsymbol{b}_{e'}\right)},$$ $$P(y_a|t) = \frac{\exp(\boldsymbol{W}_a [\boldsymbol{t}^{\mathbb{C}}; \boldsymbol{w}^{\mathbb{C}}] + \boldsymbol{b}_a)}{\sum_{a'} \exp(\boldsymbol{W}_{a'} [\boldsymbol{t}^{\mathbb{C}}; \boldsymbol{w}^{\mathbb{C}}] + \boldsymbol{b}_{a'})}. \quad (2)$$

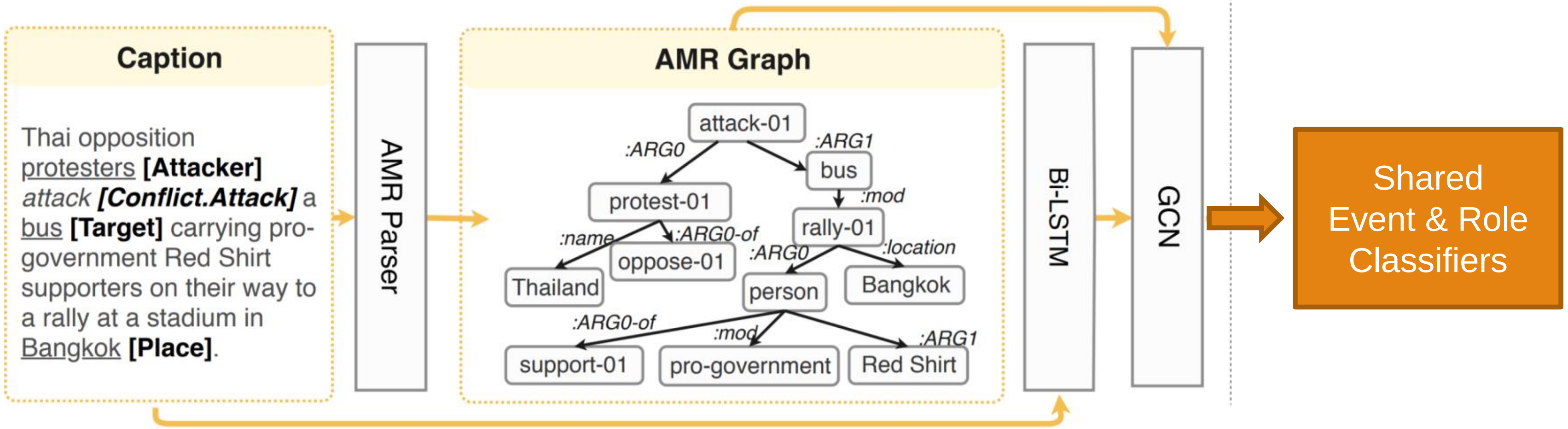


Image Event Extraction

- Produce a situation graph (similar to ARM) for each image:
  > central node is labeled as a verb.
  > neighbor nodes are arguments labeled as (noun, role).
- Propose two ways to construct a situation graphs:
  > Object-based (predefined-type object detection).
  > Attention-based (role-driven object detection).

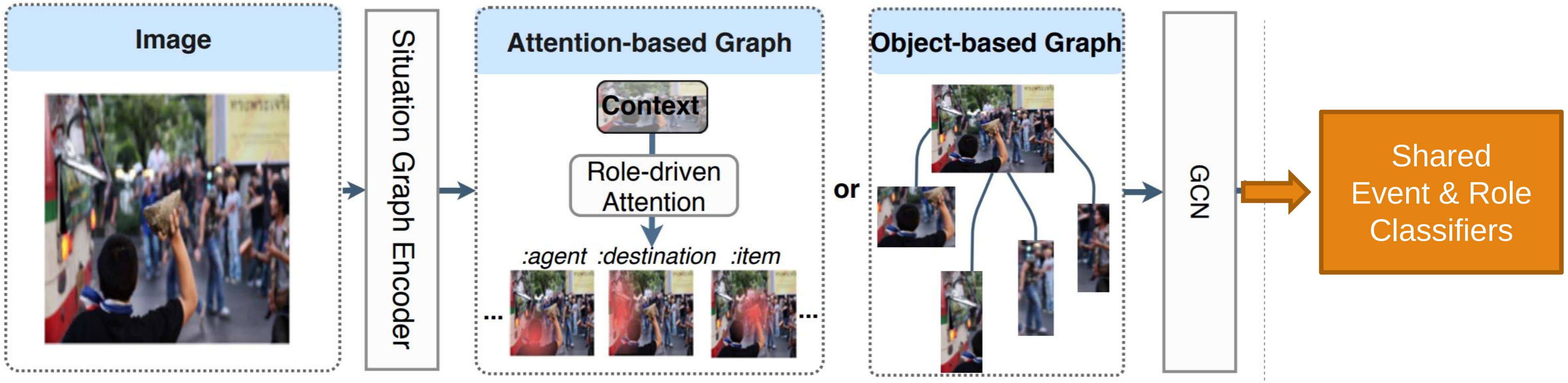


Image Event Extraction: Object-based

- Object detection: use a Faster R-CNN *(Ren et al., 2015)* trained on Open Images with 600 object types.
- Use VGG-16 CNN *(Simonyan and Zisserman, 2014)* to obtain image/object representation.
- Image representation: $\boldsymbol{m}$; object representations: $\boldsymbol{o}_i$
- Embedding layer: $\hat{\boldsymbol{m}} = \mathrm{MLP}_{\mathrm{m}}(\boldsymbol{m})$, $\hat{\boldsymbol{o}}_i = \mathrm{MLP}_{\mathrm{o}}(\boldsymbol{o}_i)$
- Verb and noun prediction: $$P(v|m) = \frac{\exp(\hat{\boldsymbol{m}}\boldsymbol{v})}{\sum_{v'} \exp(\hat{\boldsymbol{m}}\boldsymbol{v}')},$$ $$P(n|o_i) = \frac{\exp(\hat{\boldsymbol{o}}_i\boldsymbol{n})}{\sum_{n'} \exp(\hat{\boldsymbol{o}}_i\boldsymbol{n}')},$$

$$\mathcal{L}_v = -\log P(v^*|m),$$
$$\mathcal{L}_r = -\log(P(r_i^*|o_i) + P(n_i^*|o_i))$$

- Argument role labeling: $P(r_i|o_i) = \sigma\big(\mathrm{MLP}_{\mathrm{r}}(\hat{\boldsymbol{o}}_i)\big)$

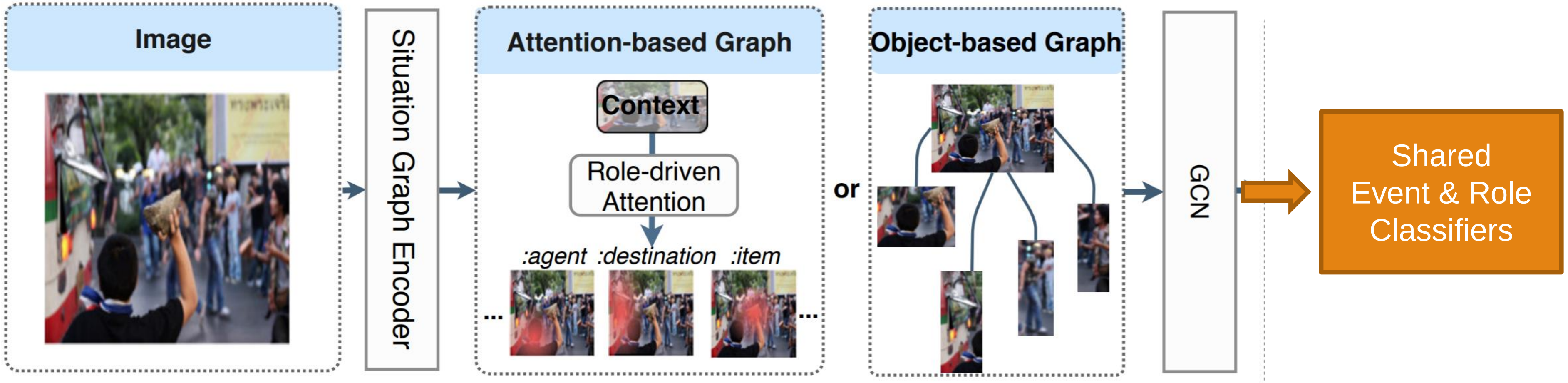
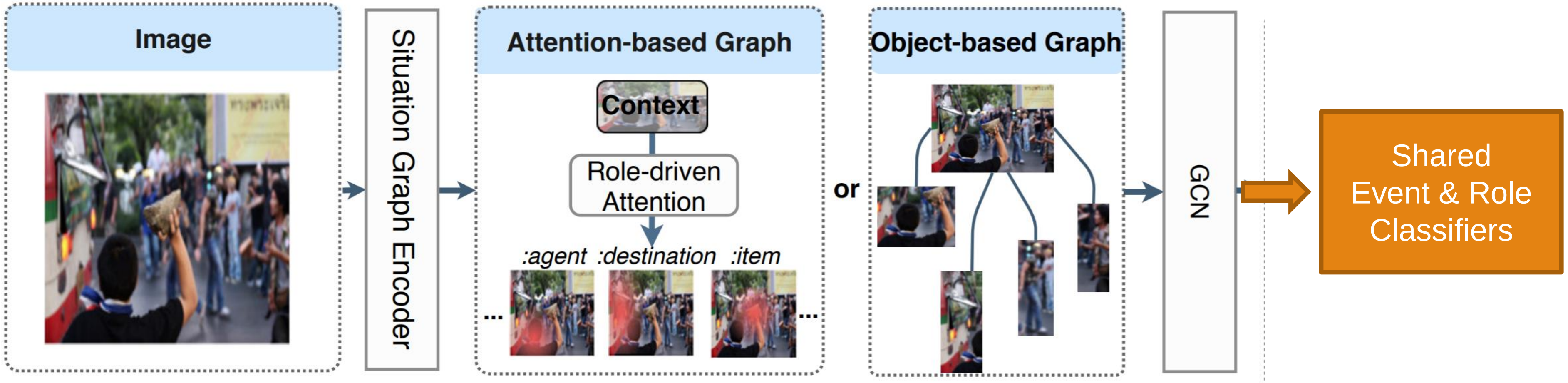


Image Event Extraction: Attention-based

- Many object types are not covered by the pretrained R-CNN model.
- Use VGG-16 CNN to obtain key vectors $\boldsymbol{k}_i$ for 7x7 local regions of the input image.
- For each possible role of the event type (i.e., verb), form the query vector: $\boldsymbol{q}_r = \boldsymbol{W}_q[\boldsymbol{r};\boldsymbol{m}] + \boldsymbol{b}_q$
- Attention is then done over the 7x7 regions to obtain object representations:

$$h_i = \frac{\exp(\boldsymbol{q}_r \boldsymbol{k}_i)}{\sum_{j \in 7\times 7} \exp(\boldsymbol{q}_r \boldsymbol{k}_j)} \qquad \boldsymbol{o}_r = \sum_i h_i \boldsymbol{m}_i$$

- Verb and noun predictions are done similar as in object-based method.



Cross-media joint training

- Form the structured common space by learning to map captions <-> images via VOA image-caption pair data.
- Soft alignment from each words to image objects and vice versa:

$$\alpha_{ij} = \frac{\exp\left(\boldsymbol{w}_{\boldsymbol{i}}^{\mathbb{C}} \boldsymbol{o}_{\boldsymbol{j}}^{\mathbb{C}}\right)}{\sum_{j'} \exp\left(\boldsymbol{w}_{\boldsymbol{i}}^{\mathbb{C}} \boldsymbol{o}_{\boldsymbol{j'}}^{\mathbb{C}}\right)}, \beta_{ji} = \frac{\exp\left(\boldsymbol{w}_{\boldsymbol{i}}^{\mathbb{C}} \boldsymbol{o}_{\boldsymbol{j}}^{\mathbb{C}}\right)}{\sum_{i'} \exp\left(\boldsymbol{w}_{\boldsymbol{i'}}^{\mathbb{C}} \boldsymbol{o}_{\boldsymbol{j}}^{\mathbb{C}}\right)}$$

- Representations aligned to the common space:

$$\boldsymbol{w}_i' = \sum_j \alpha_{ij} \boldsymbol{o}_j^{\mathbb{C}} \ , \ \boldsymbol{o}_j' = \sum_i \beta_{ji} \boldsymbol{w}_i^{\mathbb{C}}$$

- Alignment cost:

$$\langle s, m \rangle = \sum_i ||\boldsymbol{w}_i - \boldsymbol{w}_i'||_2^2 + \sum_j ||\boldsymbol{o}_j - \boldsymbol{o}_j'||_2^2$$

- Training objective:

$$\mathcal{L}_c = \max(0, 1 + \langle s, m \rangle - \langle s, m^- \rangle)$$

Training

- Training objective for verb and noun prediction:

$$\mathcal{L}_v = -\log P(v^*|m),$$
$$\mathcal{L}_r = -\log(P(r_i^*|o_i) + P(n_i^*|o_i))$$

- Training objectives for shared classifiers:

$$\mathcal{L}_e = -\sum_w \log P(y_e|w) - \sum_m \log P(y_e|m),$$
$$\mathcal{L}_a = -\sum_t \log P(y_a|t) - \sum_o \log P(y_a|o),$$

- Training objective for shared common space:

$$\mathcal{L}_c = \max(0, 1 + \langle s, m \rangle - \langle s, m^- \rangle)$$

- Overall training objective:

$$\mathcal{L} = \mathcal{L}_v + \mathcal{L}_r + \mathcal{L}_e + \mathcal{L}_a + \mathcal{L}_c$$

Inference

- Given a multimedia document with:
  > a set of sentences: $S = \{s_1, s_2, \dots\}$
  > a set of images: $M = \{m_1, m_2, \dots,\}$

- First, compute pair-wise similarities $\langle s, m \rangle$
  > select the closest image for each sentence.
  > select the closest sentence for each image.

- Compute the aligned representations for words (and objects similarly):

$$\gamma = \exp(-\langle s, m \rangle) \qquad \boldsymbol{w}_i'' = (1-\gamma)\boldsymbol{w}_i + \gamma \boldsymbol{w}_i'$$

- Predictions are then done with the aligned representations.

Results

Training	Model	Text-Only Evaluation						Image-Only Evaluation						Multimedia Evaluation					
		Event Mention			Argument Role			Event Mention			Argument Role			Event Mention			Argument Role		
		P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1	P	R	F_1
Text	JMEE	42.5	58.2	48.7	22.9	28.3	25.3	-	-	-	-	-	-	42.1	34.6	38.1	21.1	12.6	15.8
	GAIL	43.4	53.5	47.9	23.6	29.2	26.1	-	-	-	-	-	-	44.0	32.4	37.3	22.7	12.8	16.4
	$\text{WASE}^{\mathbb{T}}$	42.3	58.4	48.2	21.4	30.1	24.9	-	-	-	-	-	-	41.2	33.1	36.7	20.1	13.0	15.7
Image	$\text{WASE}^{\mathbb{I}}{}_{\text{att}}$	-	-	-	-	-	-	29.7	61.9	40.1	9.1	10.2	9.6	28.3	23.0	25.4	2.9	6.1	3.8
	$\text{WASE}^{\mathbb{I}}{}_{\text{obj}}$	-	-	-	-	-	-	28.6	59.2	38.7	13.3	9.8	11.2	26.1	22.4	24.1	4.7	5.0	4.9
Multimedia	VSE-C	33.5	47.8	39.4	16.6	24.7	19.8	30.3	48.9	26.4	5.6	6.1	5.7	33.3	48.2	39.3	11.1	14.9	12.8
	Flat_{att}	34.2	63.2	44.4	20.1	27.1	23.1	27.1	57.3	36.7	4.3	8.9	5.8	33.9	59.8	42.2	12.9	17.6	14.9
	Flat_{obj}	38.3	57.9	46.1	21.8	26.6	24.0	26.4	55.8	35.8	9.1	6.5	7.6	34.1	56.4	42.5	16.3	15.9	16.1
	WASE_{att}	37.6	66.8	48.1	27.5	33.2	**30.1**	32.3	63.4	42.8	9.7	11.1	10.3	38.2	67.1	49.1	18.6	21.6	**19.9**
	WASE_{obj}	42.8	61.9	**50.6**	23.5	30.3	26.4	43.1	59.2	**49.9**	14.5	10.1	**11.9**	43.0	62.1	**50.8**	19.5	18.9	19.2