Document-level Relation Extraction

최신 논문을 중심으로 한 서베이

집현전 최신반 7조
이창희, 하헌진, 현지웅

2021-07-25

발표자: 현지웅

발표 목차

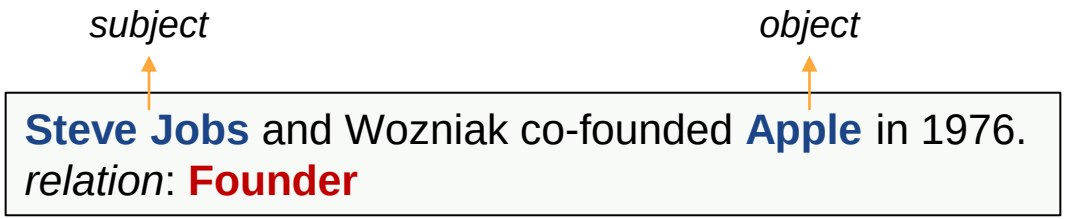
01 Task Description / Dataset

02 Graph 계열 Approach (GAIN, SIRE)

03 Transformer 계열 Approach (ATLOP, SSAN)

Document-level Relation Extraction

기존 Relation Extraction은 한 문장 내의 두 entity 간의 관계를 예측하는 task



Document-level Relation Extraction은 **전체 문서** 내의 두 entity 간의 관계를 예측하는 task

두 entity가 서로 떨어져 있는 경우 논리적인 추론이 필요함

Dataset - DocRED

현재 가장 큰 규모의 document-level relation extraction 데이터셋

Kungliga Hovkapellet

[1] ***Kungliga Hovkapellet*** (The ***Royal Court Orchestra***) is a *Swedish* orchestra, originally part of the *Royal Court* in ***Sweden***'s capital *Stockholm*. [2] The orchestra originally consisted of both musicians and singers. [3] It had only male members until *1727*, when *Sophia Schröder* and *Judith Fischer* were employed as vocalists; in the *1850s*, the harpist *Marie Pauline Åhman* became the first female instrumentalist. [4] From *1731*, public concerts were performed at ***Riddarhuset*** in *Stockholm*. [5] Since *1773*, when the ***Royal Swedish Opera*** was founded by *Gustav III* of ***Sweden***, the ***Kungliga Hovkapellet*** has been part of the opera's company.

Subject: ***Kungliga Hovkapellet; Royal Court Orchestra***
Object: ***Royal Swedish Opera***
Relation: `part_of` **Supporting Evidence:** 5

Subject: ***Riddarhuset***
Object: ***Sweden***
Relation: `country` **Supporting Evidence:** 1, 4

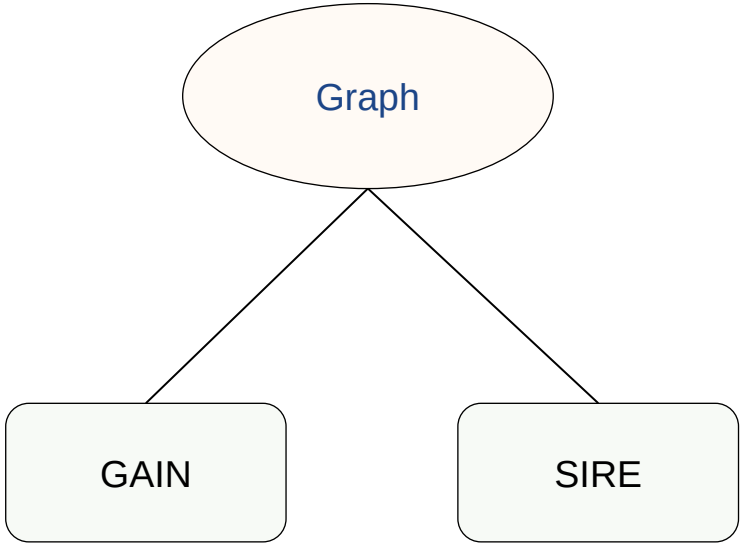
Reasoning Types	%	Examples
Pattern recognition	38.9	[1] ***Me Musical Nephews*** is a 1942 one-reel animated cartoon directed by Seymour Kneitel and animated by Tom Johnson and George Germanetti. [2] Jack Mercer and Jack Ward wrote the script. ... **Relation:** `publication_date` **Supporting Evidence:** 1
Logical reasoning	26.6	[1] "Nisei" is the ninth episode of the third season of the American science fiction television series The X-Files. ... [3] It was directed by David Nutter, and written by Chris Carter, Frank Spotnitz and Howard Gordon. ... [8] The show centers on FBI special agents ***Fox Mulder*** (David Duchovny) and Dana Scully (Gillian Anderson) who work on cases linked to the paranormal, called X-Files. ... **Relation:** `creator` **Supporting Evidence:** 1, 3, 8
Coreference reasoning	17.6	[1] ***Dwight Tillery*** is an American politician of the Democratic Party who is active in local politics of Cincinnati, Ohio. ... [3] He also holds a law degree from the University of Michigan Law School. [4] ***Tillery*** served as mayor of Cincinnati from 1991 to 1993. **Relation:** `educated_at` **Supporting Evidence:** 1, 3
Common-sense reasoning	16.6	[1] ***William Busac*** (1020-1076), son of William I, Count of Eu, and his wife Lesceline. ... [4] ***William*** appealed to King Henry I of France, who gave him in marriage Adelaide, the heiress of the county of Soissons. [5] Adelaide was daughter of Renaud I, Count of Soissons, and Grand Master of the Hotel de France. ... [7] ***William*** and Adelaide had four children: ... **Relation:** `spouse` **Supporting Evidence:** 4, 7

평가 지표: entity 단위 F1, Ign F1

* Ign F1: 학습 데이터와 검증/테스트 데이터 사이의 관계적 사실에 대한 중복을 제외하고 구한 F1

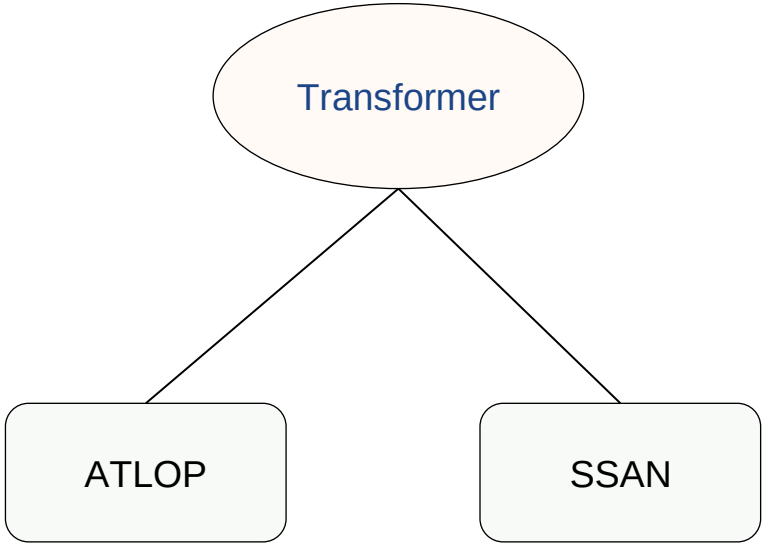
DocRED: A Large-Scale Document-Level Relation Extraction Dataset
https://arxiv.org/abs/1906.06127v3

Approaches in DocRED



bert-base
F1 : 61.24
Ign F1: 59.00

bert-base
F1 : 62.05
Ign F1: 60.18



bert-base
F1 : 61.30
Ign F1: 59.31

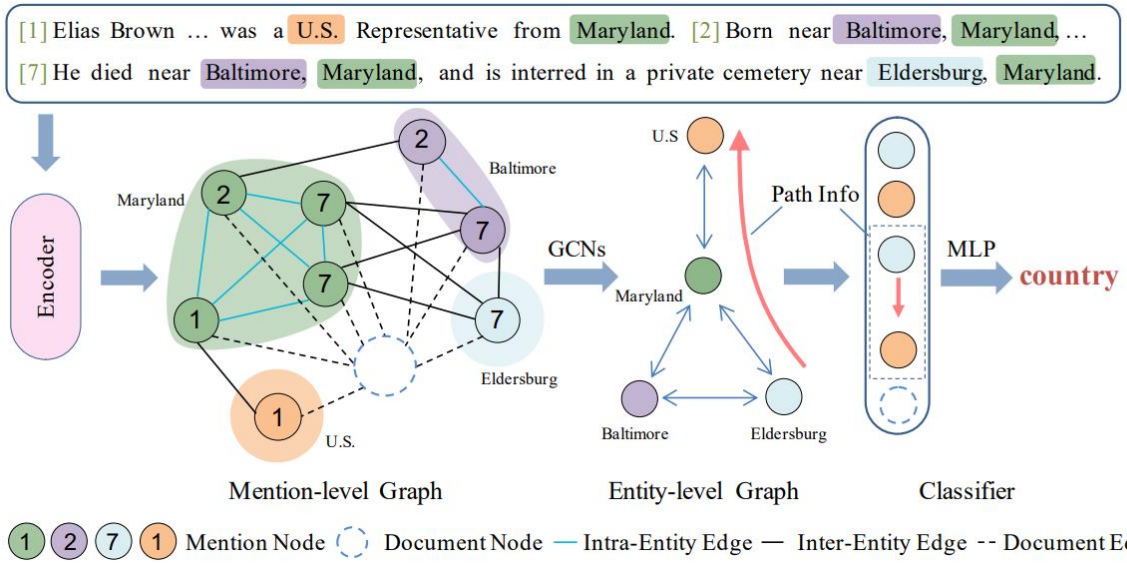
roberta-large
F1 : 63.40
Ign F1: 61.39

roberta-large + adaptation
F1 : **65.92**
Ign F1: **63.78**

roberta-large
F1 : 61.42
Ign F1: 59.47

Graph-based Approach (1): GAIN

한 줄 요약: 2개의 그래프를 모델링하여 subject와 object 간의 path 정보를 기반으로 예측



Double Graph Based Reasoning for Document-level Relation Extraction
https://arxiv.org/pdf/2009.13752.pdf

GAIN: Motivation

sentence-level에서 document-level로 확장되면서 생긴 쟁점

- subject와 object는 서로 다른 문장에 있을 수 있다.
- 같은 entity를 지칭하는 mention이 서로 다른 문장에서 여러 번 언급될 수 있다.
- coreference 및 logical reasoning 능력이 요구된다.

→ 여러 문장 간 reasoning할 수 있는 모델을 구축하자!

GAIN (Graph Aggregation and Inference Network)

GAIN: Encoder

Encoding

전체 문서 내 단어(w_i)들을 시퀀스 벡터(g_i)로 전환하는 단계

{word, entity type, co-reference} embedding을 concat하여 단어 벡터를 구성 (x_i)

단어 representation을 encoder에 넣어 최종 시퀀스 벡터를 구성 (g_i)

encoder: LSTM or BERT

$$x_i = [E_w(w_i); E_t(t_i); E_c(c_i)]$$

$$[g_1, g_2, \ldots, g_n] = Encoder([x_1, x_2, \ldots, x_n])$$

Double Graph Based Reasoning for Document-level Relation Extraction
https://arxiv.org/pdf/2009.13752.pdf

GAIN: Mention-level Graph

Mention-level Graph Aggregation Module

mention node와 document node로 구성

- Intra-Entity Edge: 동일한 entity끼리 연결
- Inter-Entity Edge: 같은 문장 내 서로 다른 entity에 대한 mention끼리 연결
- Document Edge: 문서 내 모든 mention node와 연결

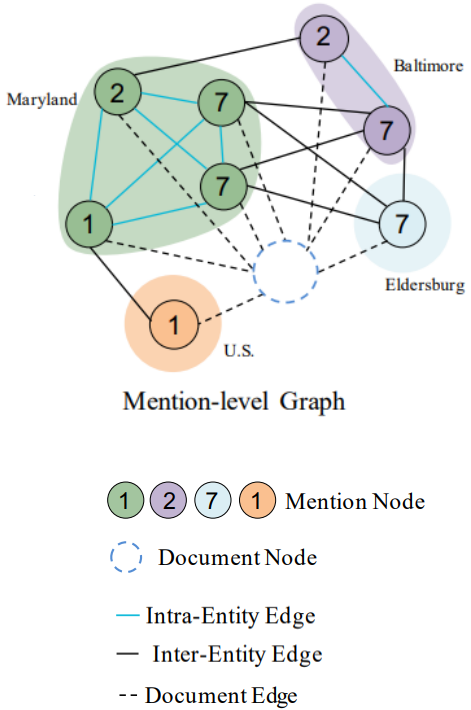
핵심 특징: 서로 다른 mention 끼리 document node를 pivot으로 하여 2단계만에 접근 가능

Mention Representation ($\mathrm{m_u}$)

init: mention 부분에 해당하는 부분의 시퀀스 벡터의 평균 $h_u^{(0)} = \frac{1}{t-s+1}\sum_{j=s}^{t} g_j$

GCN을 적용하여 해당 mention의 이웃으로부터의 feature를 통합

최종: $\mathbf{m}_u = [h_u^{(0)}; h_u^{(1)}; \ldots; h_u^{(N)}]$



Double Graph Based Reasoning for Document-level Relation Extraction
https://arxiv.org/pdf/2009.13752.pdf

GAIN: Entity-level Graph

Entity-level Graph Inference Module

entity, inter-entity edge에 대한 representation

- entity representation (e_i): mention representation의 평균
- inter-entity edge ($e_i \rightarrow e_j$): $\sigma\left(W_q[\mathbf{e}_i; \mathbf{e}_j] + b_q\right)$

head entity와 tail entity 간의 경로는 중간에 거치는 through entity를 포함 (이 논문은 two-hop path까지만 고려)

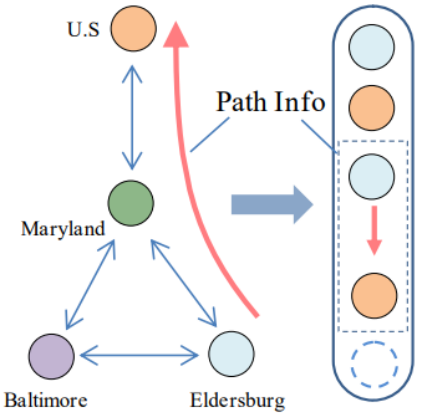
$$\mathbf{p}^i_{h,t} = [\mathbf{e}_{ho}; \mathbf{e}_{ot}; \mathbf{e}_{to}; \mathbf{e}_{oh}]$$

attention을 이용하여, 어느 through entity를 거치는 경로가 더 중요한지 모델링

$$s_i = \sigma([\mathbf{e}_h; \mathbf{e}_t] \cdot W_l \cdot \mathbf{p}^i_{h,t})$$

$$\alpha_i = \frac{e^{s_i}}{\sum_j e^{s_j}}$$

$$\mathbf{p}_{h,t} = \sum_i \alpha_i \mathbf{p}^i_{h,t}$$



Double Graph Based Reasoning for Document-level Relation Extraction
https://arxiv.org/pdf/2009.13752.pdf

GAIN: Classification

Classification Module

head, tail entity와 그에 관련된 정보 + document 정보 + 경로 정보를 종합하여 예측

- head entity (e_h)
- tail entity (e_t)
- head entity와 tail entity의 절대값 차이
- head entity와 tail entity의 element-wise 곱
- mention representation of document node (m_{doc})
- path information ($p_{h,t}$)

$$I_{h,t} = [\mathbf{e}_h; \mathbf{e}_t; |\mathbf{e}_h - \mathbf{e}_t|; \mathbf{e}_h \odot \mathbf{e}_t; \mathbf{m}_{doc}; \mathbf{p}_{h,t}]$$

$$P(r|\mathbf{e}_h, \mathbf{e}_t) = sigmoid\left(W_b\sigma(W_a I_{h,t} + b_a) + b_b\right)$$

loss: binary cross entropy (= task가 multi-label multi-class problem)

GAIN: Results

이전 SOTA 대비 F1 2.85 상승

각 module 제거 성능 비교를 통한 module의 효과성 입증

Model	Dev				Test	
	Ign F1	Ign AUC	F1	AUC	Ign F1	F1
CNN* (Yao et al., 2019)	41.58	36.85	43.45	39.39	40.33	42.26
LSTM* (Yao et al., 2019)	48.44	46.62	50.68	49.48	47.71	50.07
BiLSTM* (Yao et al., 2019)	48.87	47.61	50.94	50.26	48.78	51.06
Context-Aware* (Yao et al., 2019)	48.94	47.22	51.09	50.17	48.40	50.70
HIN-GloVe* (Tang et al., 2020)	51.06	-	52.95	-	51.15	53.30
GAT‡ (Velickovic et al., 2017)	45.17	-	51.44	-	47.36	49.51
GCNN‡ (Sahu et al., 2019)	46.22	-	51.52	-	49.59	51.62
EoG‡ (Christopoulou et al., 2019)	45.94	-	52.15	-	49.48	51.82
AGGCN‡ (Guo et al., 2019)	46.29	-	52.47	-	48.89	51.45
LSR-GloVe* (Nan et al., 2020)	48.82	-	55.17	-	52.15	54.18
GAIN-GloVe	**53.05**	**52.57**	**55.29**	**55.44**	**52.66**	**55.08**
BERT-RE*$_{base}$ (Wang et al., 2019a)	-	-	54.16	-	-	53.20
RoBERTa-RE†$_{base}$	53.85	48.27	56.05	51.35	53.52	55.77
BERT-Two-Step*$_{base}$ (Wang et al., 2019a)	-	-	54.42	-	-	53.92
HIN-BERT*$_{base}$ (Tang et al., 2020)	54.29	-	56.31	-	53.70	55.60
CorefBERT-RE*$_{base}$ (Ye et al., 2020)	55.32	-	57.51	-	54.54	56.96
LSR-BERT*$_{base}$ (Nan et al., 2020)	52.43	-	59.00	-	56.97	59.05
GAIN-BERT$_{base}$	**59.14**	**57.76**	**61.22**	**60.96**	**59.00**	**61.24**
BERT-RE*$_{large}$ (Ye et al., 2020)	56.67	-	58.83	-	56.47	58.69
CorefBERT-RE*$_{large}$ (Ye et al., 2020)	56.73	-	58.88	-	56.48	58.70
RoBERTa-RE*$_{large}$ (Ye et al., 2020)	57.14	-	59.22	-	57.51	59.62
CorefRoBERTa-RE*$_{large}$ (Ye et al., 2020)	57.84	-	59.93	-	57.68	59.91
GAIN-BERT$_{large}$	**60.87**	**61.79**	**63.09**	**64.75**	**60.31**	**62.76**

Model	Dev				Test	
	Ign F1	Ign AUC	F1	AUC	Ign F1	F1
GAIN-GloVe	**53.05**	**52.57**	**55.29**	**55.44**	**52.66**	**55.08**
- *hMG*	50.97	48.84	53.10	51.73	50.76	53.06
- *Inference Module*	50.84	48.68	53.02	51.58	50.32	52.66
- *Document Node*	50.86	48.68	53.01	52.46	50.32	52.67
GAIN-BERT$_{base}$	**59.14**	**57.76**	**61.22**	**60.96**	**59.00**	**61.24**
- *hMG*	57.12	51.54	59.17	54.61	57.31	59.56
- *Inference Module*	56.97	54.29	59.28	57.25	57.01	59.34
- *Document Node*	57.26	52.07	59.62	55.51	57.01	59.63

GAIN: Ablation Study

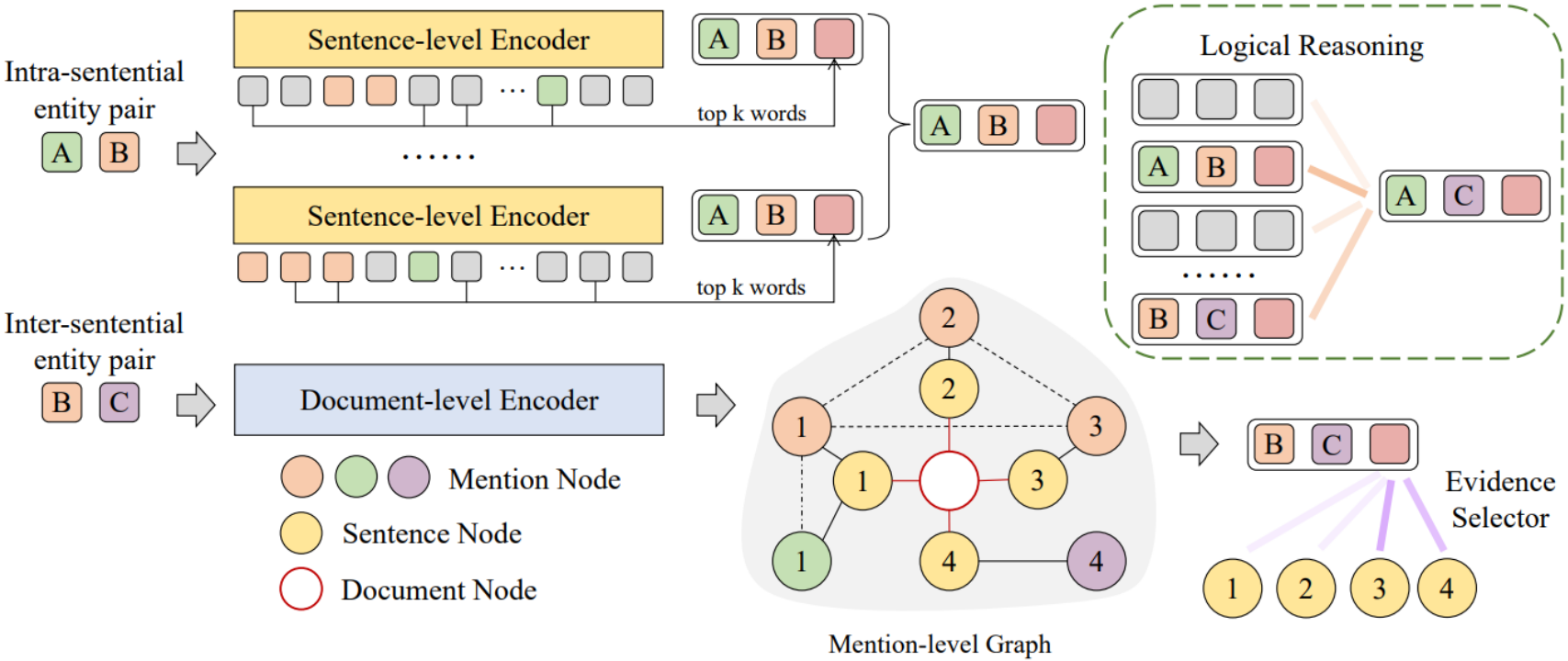
문장 내 관계/문장 외 관계에서의 성능 비교: hMG의 효과 입증
Reasoning을 통해 알 수 있는 성능만 비교: Inference 모듈의 효과 입증

Model	Intra-F1	Inter-F1
CNN*	51.87	37.58
LSTM*	56.57	41.47
BiLSTM*	57.05	43.49
Context-Aware*	56.74	42.26
LSR-GloVe*	60.83	48.35
GAIN-GloVe	**61.67**	**48.77**
- *hMG*	59.72	46.49
BERT-RE*$_{base}$	61.61	47.15
RoBERTa-RE$_{base}$	65.65	50.09
BERT-Two-Step*$_{base}$	61.80	47.28
LSR-BERT*$_{base}$	65.26	52.05
GAIN-BERT$_{base}$	**67.10**	**53.90**
- *hMG*	66.15	51.42

Model	Infer-F1	P	R
CNN	37.11	32.81	42.72
LSTM	39.03	33.16	47.44
BiLSTM	38.73	31.60	50.01
Context-Aware	39.73	**33.97**	47.85
GAIN-GloVe	**40.82**	32.76	**54.14**
- *Inference Module*	39.76	32.26	51.80
BERT-RE$_{base}$	39.62	34.12	47.23
RoBERTa-RE$_{base}$	41.78	37.97	46.45
GAIN-BERT$_{base}$	**46.89**	**38.71**	**59.45**
- *Inference Module*	45.11	36.91	57.99

Graph-based Approach (2): SIRE

한 줄 요약: 두 mention 간의 관계가 같은 문장 내에 존재하는지에 따라 서로 다른 모델링을 하자



SIRE: Separate Intra- and Inter-sentential Reasoning for Document-level Relation Extraction
https://arxiv.org/abs/2106.01709

SIRE: Motivation

Relation types in DocRE
intra-sentential: 두 entity가 같은 문장 안에 등장하는 경우
inter-sentential: 그렇지 않은 경우

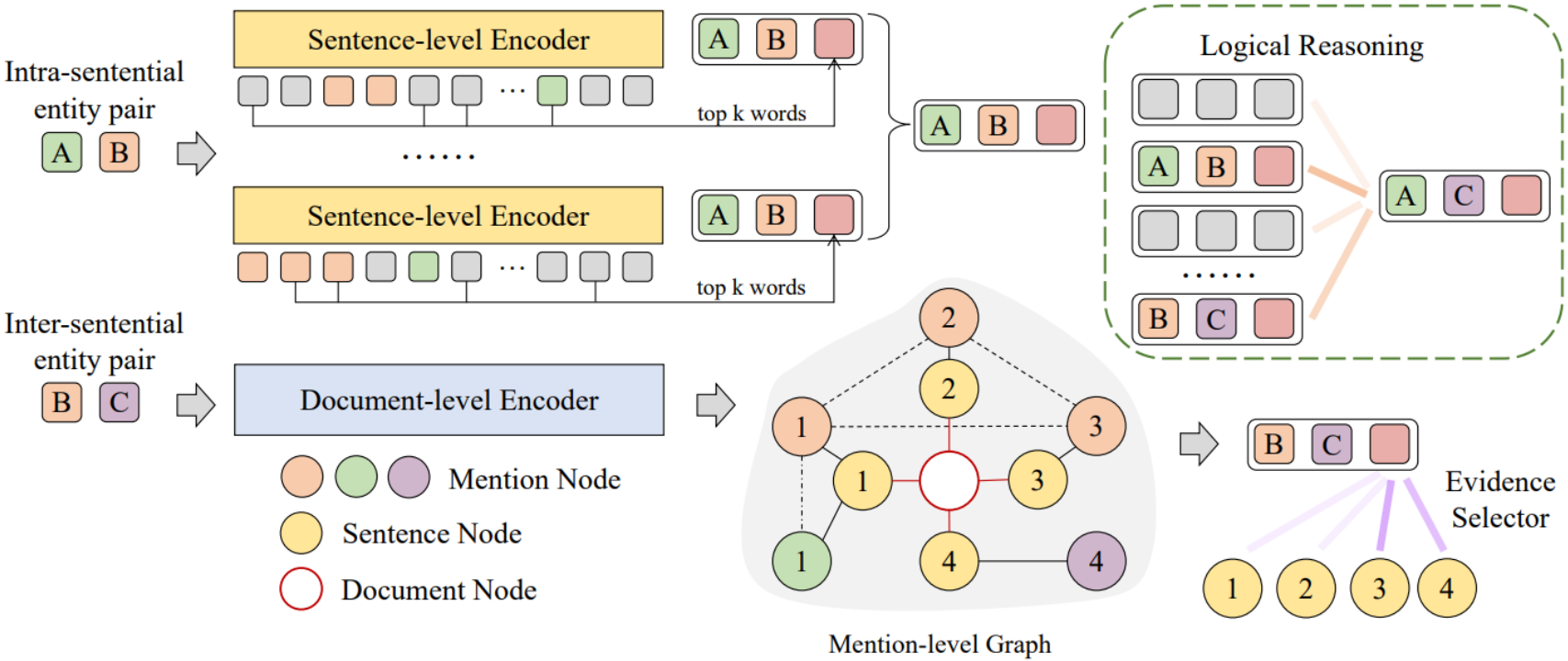
Previous work in DocRE
relation의 종류에 상관없이 모든 entity pair들에 대하여 가능한 relation을 예측하는 방식으로 접근

→ 언어학적 관점에서 intra-sentential과 inter-sentential은 다른 패턴으로 표현되어야 한다.

→ SIRE (**S**eparate **I**ntra- and **I**nter-sentential **Re**asoning)

SIRE

3개의 Module로 구성되어 있음 (Intra- and Inter-sentential Relation Representation Module, Logical Reasoning Module, Classification Module)



SIRE: Separate Intra- and Inter-sentential Reasoning for Document-level Relation Extraction
https://arxiv.org/abs/2106.01709

SIRE: Intra-sentential Relation Representation Module

Encoding

각 문장(S_i)내 단어($w_j^{S_i}$)들을 시퀀스 벡터($g_j^{S_i}$)로 전환하는 단계

{word, entity type, co-reference} embedding을 concat하여
단어 벡터를 구성 (x) $\mathbf{x} = [E_w(w); E_t(t); E_c(c)]$

단어 representation을 sentence-level encoder (f_{enc}^{S})에 넣어
최종 시퀀스 벡터를 구성 ($g_j^{S_i}$) $[\mathbf{g}_1^{S_i}, \ldots, \mathbf{g}_{n_i}^{S_i}] = f_{enc}^{S}([\mathbf{x}_1^{S_i}, \ldots, \mathbf{x}_{n_i}^{S_i}])$

sentence-level encoder: LSTM or BERT

GAIN과 동일

SIRE: Intra-sentential Relation Representation Module

Representing

각 entity pair $(e_{i,h}, e_{i,t})$에 대하여 intra-sentential한 관계를 표현하는 단계 (entity, context)

head entity mention과 tail entity mention이 같은 문장에 등장하는 문장들의 집합 $(S_{co-occur})$ 에 대하여, 각 문장별로 context representation을 구한다 $\{\mathcal{S}_{i_1}, \mathcal{S}_{i_2}, \ldots, \mathcal{S}_{i_C}\}$

* 저자진들은 context representation을 두 mention와 관련 높은 top K개의 단어들로 정의하였음

top-k related word representation

head entity mention과 tail entity mention을 query로 하여 문장 내 모든 단어들에 대한 relatedness(attention) score를 구하여 이를 기반으로 top K related word를 구한다

$$\mathbf{e}_{i,h}^{\mathcal{S}_{i_j}} = \frac{1}{t-s+1} \sum_{k=s}^{t} \mathbf{g}_k^{\mathcal{S}_{i_j}}$$

$$s_{i,k} = \sigma((W_{intra} \cdot [\mathbf{e}_{i,h}^{\mathcal{S}_{i_j}}; \mathbf{e}_{i,t}^{\mathcal{S}_{i_j}}])^T \cdot \mathbf{g}_k^{\mathcal{S}_{i_j}})$$

$$\alpha_{i,k} = Softmax(s_{i,k})$$

SIRE: Intra-sentential Relation Representation Module

context information

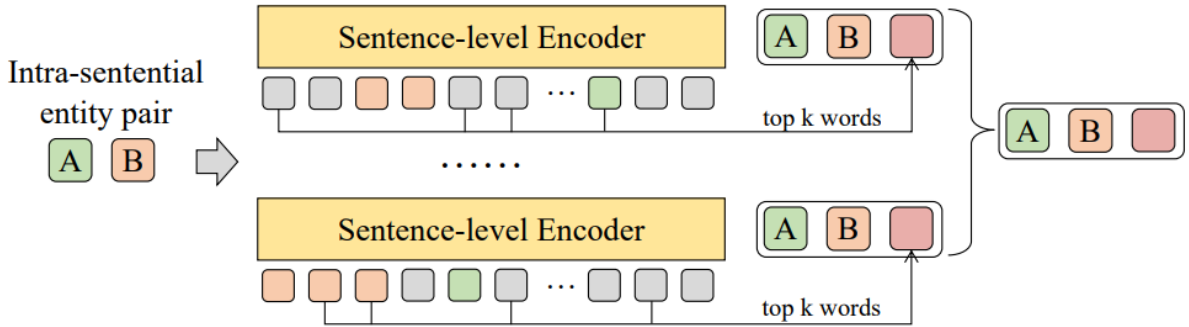
top K related word의 representation과 전체 weighted average representation을 더하여 context information($c_i^{S_{i_j}}$)을 구축한다.

* 이유: relatedness score를 구하기 위해 적용한 W_{intra}에 대한 gradient를 구해야 학습이 가능하기 때문

intra-relation representation (최종)

각 문장의 head, tail, context representation을 average한다.

$$\mathbf{c}_i^{\mathcal{S}_{i_j}} = \beta \cdot \frac{1}{K} \sum_{k \in topK(\alpha_{i,*})} \mathbf{g}_k^{\mathcal{S}_{i_j}} + (1-\beta) \cdot \sum_{t}^{n_{i_j}} \alpha_{i,t} \mathbf{g}_t^{\mathcal{S}_{i_j}} \qquad \mathbf{r}_i = \frac{1}{C} \sum_{\mathcal{S}_{i_j} \in \mathcal{S}_{co-occur}} [\mathbf{e}_{i,h}^{\mathcal{S}_{i_j}}; \mathbf{e}_{i,t}^{\mathcal{S}_{i_j}}; \mathbf{c}_i^{\mathcal{S}_{i_j}}]$$



SIRE: Inter-sentential Relation Representation Module

Encoding

intra-와 동일한 방식으로 단어 representation을 구성

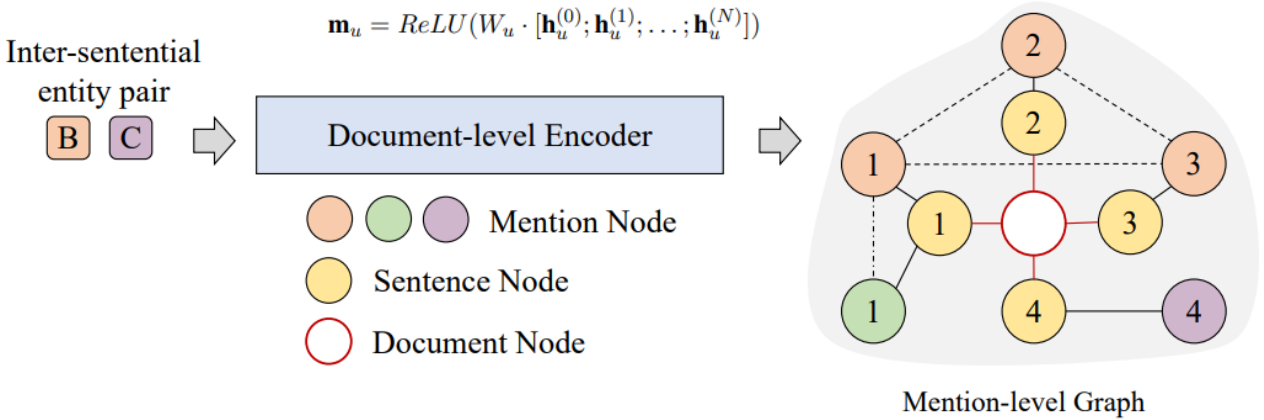
단어 representation을 document-level encoder (f_{enc}^{D})에 넣어 시퀀스 벡터를 구성 (g_j^D)

mention-level graph (MG)를 적용하여 좀 더 문서 안의 상호 관계를 잘 표현할 수 있도록 함

R-GCN 및 feature aggregation을 적용하여 각 node에 대한 representation을 뽑아낸다

$$\mathbf{h}_u^{(l+1)} = ReLU\left(\sum_{t\in\mathcal{T}}\sum_{v\in\mathcal{N}_u^t\cup\{u\}}\frac{1}{c_{u,t}}W_t^{(l)}\mathbf{h}_v^{(l)}\right) \qquad \mathbf{h}_u^{(0)} = \frac{1}{t-s+1}\sum_{j=s}^{t}\mathbf{g}_j^{\mathcal{D}}$$

$$\mathbf{m}_u = ReLU(W_u\cdot[\mathbf{h}_u^{(0)};\mathbf{h}_u^{(1)};\ldots;\mathbf{h}_u^{(N)}])$$



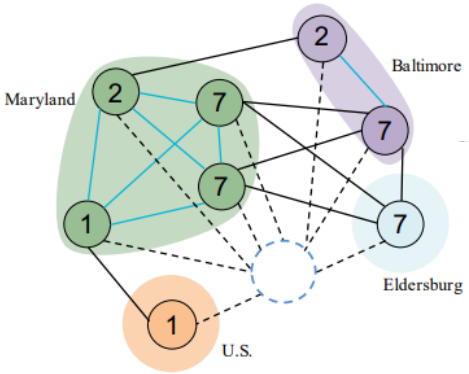
SIRE: Mention-level Graph

Mention-level Graph in GAIN (왼쪽)

mention node와 document node로 구성
서로 다른 mention 끼리 document node를 pivot으로 하여 2단계만에 접근 가능

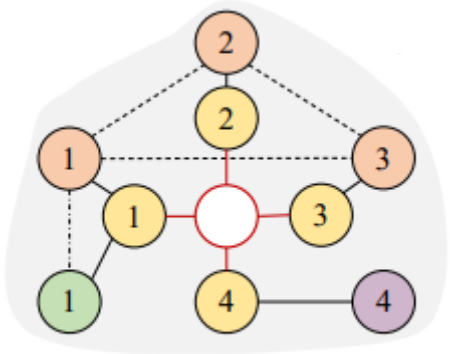
Mention-level Graph in SIRE (오른쪽)

GAIN의 MG는 local context information을 반영하고 있지 못함
따라서, sentence node와 그에 상응되는 edge를 추가하여 MG에 local information을 주입



Mention-level Graph

- **Intra-Entity Edge**: 동일한 entity에 대한 mention끼리 연결
- **Inter-Entity Edge**: 동일 문장 내 서로 다른 entity에 대한 mention끼리 연결
- **Document Edge**: 문서 내 모든 mention과 document node와 연결



Mention-level Graph

- **Sentence-Mention Edge**: 각 문장 내 mention와 문장과 연결
- **Sentence-Document Edge**: 각 문장과 document node와 연결
- 기존 Intra-,Inter-Entity Edge는 동일

SIRE: Inter-sentential Relation Representation Module

Representing

각 entity pair $(e_{i,h}, e_{i,t})$에 대하여 inter-sentential한 관계를 표현하는 단계

entity representation은 MG을 통해 구한 mention representation의 평균 $\mathbf{e}_i = \frac{1}{N} \sum_{j \in M(e_i)} \mathbf{m}_j$

각 sentence node에 대하여 attention을 적용하여 어느 문장이 추론의 evidence가 될 수 있을지 모델링
→ **context representation**

$$P(\mathcal{S}_k | e_{i,h}, e_{i,t}) = \sigma(W_k \cdot [\mathbf{e}_{i,h}; \mathbf{e}_{i,t}; \mathbf{m}_{\mathcal{S}_k}])$$

$$\alpha_{i,k} = \frac{P(\mathcal{S}_k | e_{i,h}, e_{i,t})}{\sum_l P(\mathcal{S}_l | e_{i,h}, e_{i,t})} \qquad \mathbf{c}_i = \sum_k^l \alpha_{i,k} \cdot \mathbf{m}_{\mathcal{S}_k}$$

최종 relation representation은 head, tail, context을 concat

$$\mathbf{r}_i = [\mathbf{e}_{i,h}; \mathbf{e}_{i,t}; \mathbf{c}_i]$$

SIRE: Logical Reasoning Module

Logical Reasoning in Previous Work

각 entity pair 간들의 경로들을 단서로 하여 사용

문제: 모든 entity pair들이 연결되어 있지 않기도 하고 그래프 내에 올바른 추론 경로가 있다는 보장이 없음

Logical Reasoning in SIRE

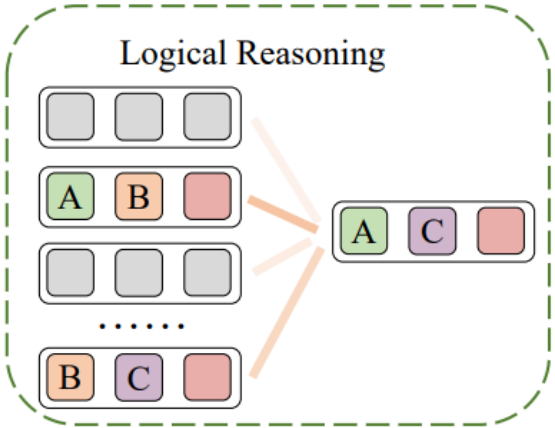
self-attention을 사용하여 logical reasoning을 모델링

한 entity pair (e_h, e_t)에 대하여 two-hop 형식의 logical reasoning chain $\{e_h \rightarrow e_k \rightarrow e_t\}$이 있다고 가정 ($e_k$는 문서 내 존재하는 다른 entity)

$$\mathbf{r}_i^{new} = \sum_{\mathbf{r}_k \in \mathcal{R}_{att} \cup \{\mathbf{r}_i\}} \gamma_k \cdot \mathbf{r}_k$$

$$\gamma_k = Softmax((W_{att} \cdot \mathbf{r}_i)^T \cdot \mathbf{r}_k)$$

*R_{att}는 (e_h, e_k), (e_k, e_t)에 대한 relational representation



SIRE: Classification Module, Results

intra- inter-sentential relation 정보를 종합하여 multi-label classification task로 학습
: with two FC layer with sigmoid

$$P(r|e_{i,h}, e_{i,t}) = sigmoid\left(W_1\sigma(W_2\mathbf{r}_i + b_1) + b_2\right)$$

Model	Dev				Test	
	Ign F1	F1	Intra-F1	Inter-F1	Ign F1	F1
BiLSTM (Yao et al., 2019b)	48.87	50.94	57.05	43.49	48.78	51.06
HIN-GloVe (Tang et al., 2020)	51.06	52.95	-	-	51.15	53.30
LSR-GloVe (Nan et al., 2020)	48.82	55.17	60.83	48.35	52.15	54.18
GAIN-GloVe (Zeng et al., 2020)	53.05	55.29	61.67	48.77	52.66	55.08
SIRE-GloVe	**54.10**	**55.91**	**62.94**	**48.97**	**54.04**	**55.96**
-LR Module	53.73	55.58	62.77	47.87	53.75	55.55
-context	52.57	54.41	61.66	46.92	52.33	54.15
-inter4intra	52.23	54.26	60.81	48.36	51.77	53.30
BERT (Wang et al., 2019a)	-	54.16	61.61	47.15	-	53.20
BERT-Two-Step (Wang et al., 2019a)	-	54.42	61.80	47.28	-	53.92
HIN-BERT (Tang et al., 2020)	54.29	56.31	-	-	53.70	55.60
CorefBERT (Ye et al., 2020)	55.32	57.51	-	-	54.54	56.96
GLRE-BERT (Wang et al., 2020)	-	-	-	-	55.40	57.40
LSR-BERT (Nan et al., 2020)	52.43	59.00	65.26	52.05	56.97	59.05
GAIN-BERT (Zeng et al., 2020)	59.14	61.22	67.10	53.90	59.00	61.24
SIRE-BERT	**59.82**	**61.60**	**68.07**	**54.01**	**60.18**	**62.05**

SIRE: Ablation Study

Ablation Study in SIRE-GloVe

Logical Reasoning Module의 효과를 검증하기 위한 실험

해당 module 제거 후 F1 0.41 하락

context representation의 효과를 검증하기 위한 실험

context 제거 후 F1 1.81 하락

intra-와 inter-의 구분 필요성을 검증하기 위한 실험

모든 entity pairs에 대해서 inter-sentential module만 적용했을 때, F1 2.66, Intra-F1 2.13 하락

Model	Dev				Test	
	Ign F1	F1	Intra-F1	Inter-F1	Ign F1	F1
SIRE-GloVe	**54.10**	**55.91**	**62.94**	**48.97**	**54.04**	**55.96**
-LR Module	53.73	55.58	62.77	47.87	53.75	55.55
-context	52.57	54.41	61.66	46.92	52.33	54.15
-inter4intra	52.23	54.26	60.81	48.36	51.77	53.30

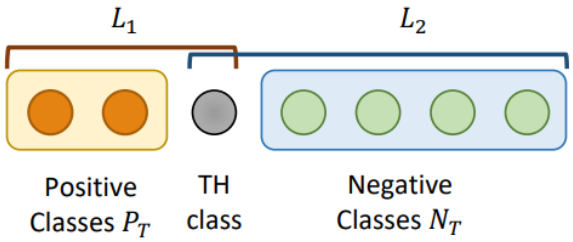
Reasoning Ability

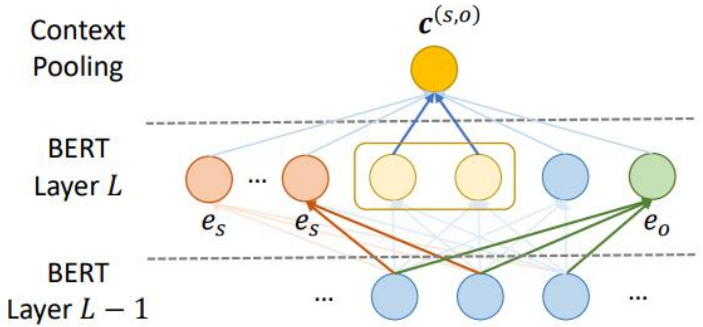
dev set 안에 two-hop을 통해 예측할 수 있는 relation에 대해서만 F1 score를 평가 (Infer-F1)

Model	Infer-F1	P	R
BiLSTM	38.73	31.60	50.01
GAIN-GloVe	40.82	32.76	54.14
SIRE-GloVe	**42.72**	**34.83**	**55.22**
- LR Module	39.18	31.97	50.59

Transformer-based Approach (1): ATLOP

한 줄 요약: class마다 다른 threshold를 줘서 예측하고, 각 entity pair에 attention을 적용





Document-Level Relation Extraction with Adaptive Thresholding and Localized Context Pooling
https://arxiv.org/abs/2010.11304

ATLOP: Motivation

Common practice in multi-label classification

검증 데이터셋의 성능을 최대화하는 threshold를 결정하여 예측하는 방식

→ 각 class마다 그에 상응하는 confidence score가 있을 것이다

Baseline in DocRED

모든 entity pair들의 평균을 최종 context representation으로 나타냄

→ 몇 개의 entity pair는 관계를 예측하는 데 관련이 없을 수도 있다

ATLOP: **A**daptive **T**hresholding + **L**ocalized C**o**ntext **P**ooling

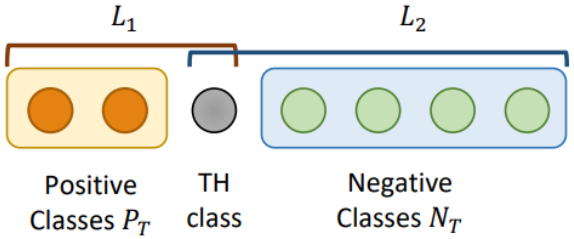
ATLOP: Adaptive Thresholding

positive classes, negative classes, and TH class

entity pair가 해당되는 relation이 positive classes, 그 외를 negative classes
positive classes의 logit은 negative classes의 logit보다 높아야 함
→ TH class를 도입하여 추론 시 TH class 보다 높은 class를 relation으로 예측

special loss for TH class (adaptive thresholding loss)

categorical cross entropy 기반 loss로
(1) positive classes와 TH 사이 (2) negative classes와 TH 사이로 구성



$$\mathcal{L}_1 = -\sum_{r \in \mathcal{P}_T} \log \left(\frac{\exp\left(\text{logit}_r\right)}{\sum_{r' \in \mathcal{P}_T \cup \{\text{TH}\}} \exp\left(\text{logit}_{r'}\right)} \right),$$

$$\mathcal{L}_2 = -\log \left(\frac{\exp\left(\text{logit}_{\text{TH}}\right)}{\sum_{r' \in \mathcal{N}_T \cup \{\text{TH}\}} \exp\left(\text{logit}_{r'}\right)} \right),$$

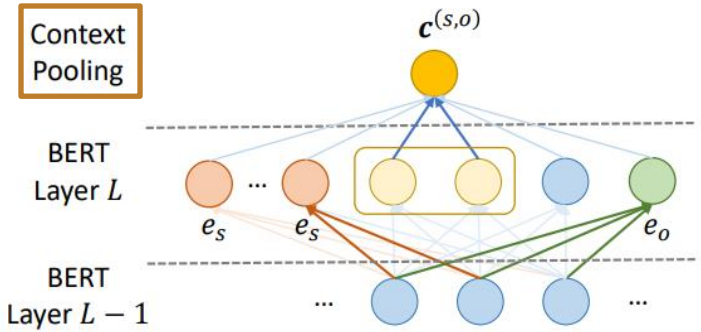
$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2.$$

ATLOP: Localized Context Pooling

subject와 object 간 attention

pre-trained multi-head attention matrix A로부터, mention-level attention 및 entity-level attention $(\mathrm{A}_{\mathrm{s}}^{\mathrm{E}}, \mathrm{A}_{\mathrm{o}}^{E})$ 을 구축

subject와 object의 attention을 곱하여 최종 entity-pair attention 및 localized context embedding 구축



$$A^{(s,o)} = A_s^E \cdot A_o^E,$$

$$q^{(s,o)} = \sum_{i=1}^{H} A_i^{(s,o)},$$

$$a^{(s,o)} = q^{(s,o)} / \mathbf{1}^\intercal q^{(s,o)},$$

$$c^{(s,o)} = H^\intercal a^{(s,o)},$$

ATLOP: Encoder and Classifier

그 외 다른 점

entity hidden states: logsumexppooling 선택, 바로 classification이 아닌 linear에 한번 더

bilinear: group bilinear를 사용하여 모델 파라미터 개수 reduce ($d^2 \rightarrow d^2/k$)

$$\boldsymbol{H} = [\boldsymbol{h}_1, \boldsymbol{h}_2, ..., \boldsymbol{h}_l] = \text{BERT}([x_1, x_2, ..., x_l]).$$

$$\boldsymbol{h}_{e_i} = \log \sum_{j=1}^{N_{e_i}} \exp\left(\boldsymbol{h}_{m_j^i}\right).$$

$$\boldsymbol{z}_s^{(s,o)} = \tanh\left(\boldsymbol{W}_s \boldsymbol{h}_{e_s} + \boldsymbol{W}_{c_1} \boldsymbol{c}^{(s,o)}\right),$$

$$\boldsymbol{z}_o^{(s,o)} = \tanh\left(\boldsymbol{W}_o \boldsymbol{h}_{e_o} + \boldsymbol{W}_{c_2} \boldsymbol{c}^{(s,o)}\right),$$

$$\left[\boldsymbol{z}_s^1; ...; \boldsymbol{z}_s^k\right] = \boldsymbol{z}_s,$$

$$\left[\boldsymbol{z}_o^1; ...; \boldsymbol{z}_o^k\right] = \boldsymbol{z}_o,$$

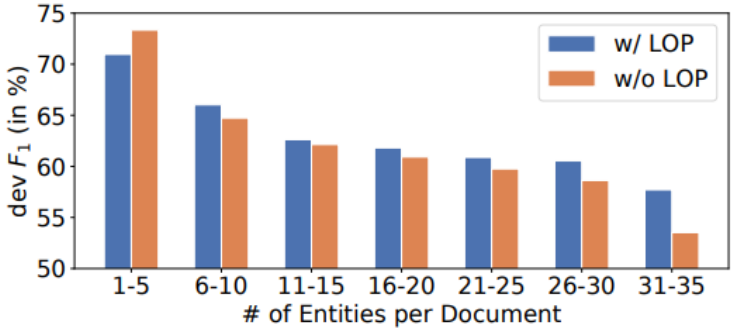
$$\text{P}\left(r | e_s, e_o\right) = \sigma\left(\sum_{i=1}^{k} \boldsymbol{z}_s^{i\intercal} \boldsymbol{W}_r^i \boldsymbol{z}_o^i + b_r\right),$$

ATLOP: Results / Ablation Study

Model	Dev		Test	
	Ign F_1	F_1	Ign F_1	F_1
Sequence-based Models				
CNN (Yao et al. 2019)	41.58	43.45	40.33	42.26
BiLSTM (Yao et al. 2019)	48.87	50.94	48.78	51.06
Graph-based Models				
BiLSTM-AGGCN (Guo, Zhang, and Lu 2019)	46.29	52.47	48.89	51.45
BiLSTM-LSR (Nan et al. 2020)	48.82	55.17	52.15	54.18
BERT-LSR$_{\text{BASE}}$ (Nan et al. 2020)	52.43	59.00	56.97	59.05
Transformer-based Models				
BERT$_{\text{BASE}}$ (Wang et al. 2019a)	-	54.16	-	53.20
BERT-TS$_{\text{BASE}}$ (Wang et al. 2019a)	-	54.42	-	53.92
HIN-BERT$_{\text{BASE}}$ (Tang et al. 2020a)	54.29	56.31	53.70	55.60
CorefBERT$_{\text{BASE}}$ (Ye et al. 2020)	55.32	57.51	54.54	56.96
CorefRoBERTa$_{\text{LARGE}}$ (Ye et al. 2020)	57.35	59.43	57.90	60.25
Our Methods				
BERT$_{\text{BASE}}$ (our implementation)	54.27 ± 0.28	56.39 ± 0.18	-	-
BERT-E$_{\text{BASE}}$	56.51 ± 0.16	58.52 ± 0.19	-	-
BERT-ATLOP$_{\text{BASE}}$	59.22 ± 0.15	61.09 ± 0.16	59.31	61.30
RoBERTa-ATLOP$_{\text{LARGE}}$	**61.32 ± 0.14**	**63.18 ± 0.19**	**61.39**	**63.40**

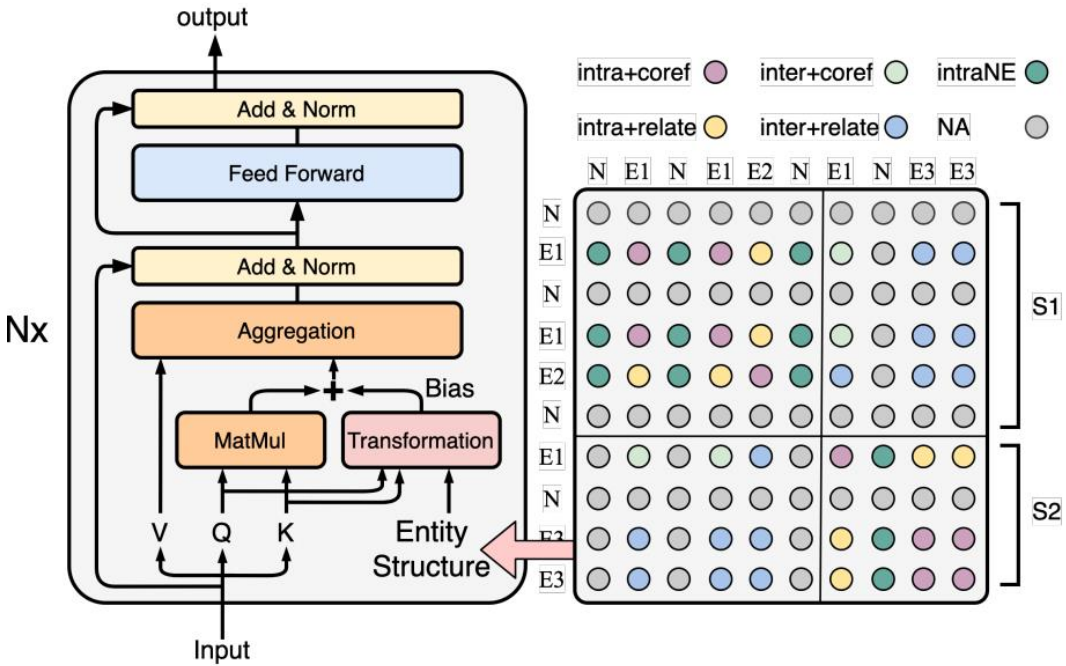
Model	Ign F_1	F_1
BERT-ATLOP$_{\text{BASE}}$	59.22	61.09
– Adaptive Thresholding	58.32	60.20
– Localized Context Pooling	58.19	60.12
– Adaptive-Thresholding Loss	39.52	41.74

Strategy	Dev F_1	Test F_1
Global Thresholding	60.14	60.62
Per-class Thresholding	**61.73**	60.35
Adaptive Thresholding	61.27	**61.30**



Transformer-based Approach (2): SSAN

한 줄 요약: 두 mention 간 관계를 같은 문장 내 존재하는 지와 같은 entity를 나타내는지에 따라 모델링



Entity Structure Within and Throughout: Modeling Mention Dependencies for Document-Level Relation Extraction
https://arxiv.org/abs/2102.10249

SSAN: Motivation

Coreference information

coreference 정보를 인코더 단계에서 embedding layer만으로 사용
같은 mention에 대해서 단순히 average pooling

Entity structure in graph

contextual representation encoder (LSTM) + graph 표현으로 entity 구조 모델링
: 인코더와 그래프 네트워크 간의 이질성…

즉, 문서 내 구조적 dependency를 encoding network와 더불어 전체 시스템에 잘 통합해야 한다.

→ SSAN: **S**tructured **S**elf-**A**ttention **N**etwork

SSAN: Entity Structure

Coreference

같은 entity를 지칭하는 mention끼리 True(coref), 서로 다른 entity를 지칭하는 mention끼리 False(relate)

Co-occurrence

같은 문장 내 존재하는 mention끼리 True(intra), 서로 다른 문장에 존재하는 mention끼리 False(inter)

		Coreference	
		True	False
Co-occurence	True	*intra+coref*	*intra+relate*
	False	*inter+coref*	*inter+relate*

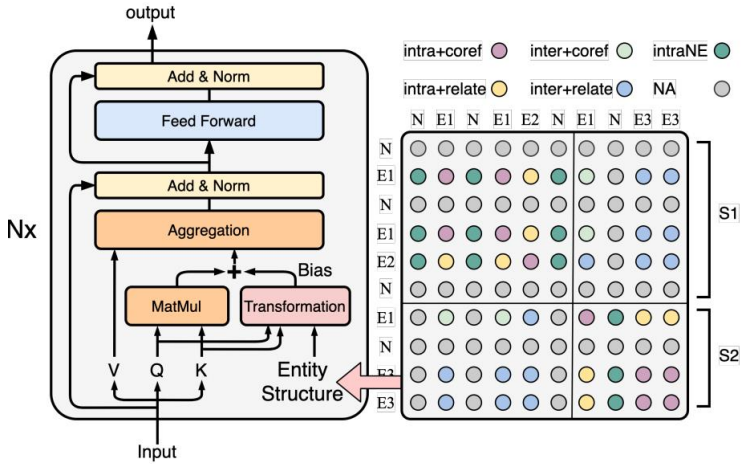
SSAN: SSAN Network

Transformer

기존 transformer와 똑같은 구조를 가지며, query와 key를 기반으로 attention을 구하는 과정부터 다르다.

Attention

기존 transformer로 구하는 attention (unstructured attention score)에 entity structure를 transformation한 결과를 value로 하여 구해지는 attention 점수가 최종 attention이 된다.
이를 최종 반영하여 aggregate하여 contextual representation을 얻는다.



$$e_{ij}^{l} = \frac{\boldsymbol{q}_i^l {\boldsymbol{k}_j^l}^T}{\sqrt{d}}$$

$$\tilde{e}_{ij}^{l} = e_{ij}^{l} + \frac{transformation(\boldsymbol{q}_i^l, \boldsymbol{k}_j^l, s_{ij})}{\sqrt{d}}$$

$$\boldsymbol{z}_i^{l+1} = \sum_{j=1}^{n} \frac{exp\ \tilde{e}_{ij}^{l}}{\sum_{k=1}^{n} exp\ \tilde{e}_{ik}^{l}} \boldsymbol{v}_j^l$$

SSAN: Transformation / Classification

structure와 학습하는 모델을 잘 통합하기 위해, 해당 구조를 모델의 파라미터로 연결하는 과정을 거친다.

Biaffine Transformation $bias_{ij}^{l} = \boldsymbol{q}_i^l \boldsymbol{A}_{l,s_{ij}} {\boldsymbol{k}_j^l}^T + b_{l,s_{ij}}$

Decomposed Linear Transformation $bias_{ij}^{l} = \boldsymbol{q}_i^l \boldsymbol{K}_{l,s_{ij}}^T + \boldsymbol{Q}_{l,s_{ij}} {\boldsymbol{k}_j^l}^T + b_{l,s_{ij}}$

Classification

entity representation: 해당하는 representation의 average pooling

prediction: bilinear

$$P_r(\boldsymbol{e}_s, \boldsymbol{e}_o) = sigmoid(\boldsymbol{e}_s \boldsymbol{W}_r \boldsymbol{e}_o)$$

$$L = \sum_{<s,o>} \sum_{r} CrossEntropy(P_r(\boldsymbol{e}_s, \boldsymbol{e}_o), \overline{y}_r(\boldsymbol{e}_s, \boldsymbol{e}_o))$$

SSAN: Results

Model	Dev Ign F1 / F1	Test Ign F1 / F1
ContexAware (2019)	48.94 / 51.09	48.40 / 50.70
EoG*(2019)	45.94 / 52.15	49.48 / 51.82
BERT Two-Phase (2019a)	- / 54.42	- / 53.92
GloVe+LSR (2020)	48.82 / 55.17	52.15 / 54.18
HINBERT (2020)	54.29 / 56.31	53.70 / 55.60
CorefBERT Base (2020)	55.32 / 57.51	54.54 / 56.96
CorefBERT Large (2020)	56.73 / 58.88	56.48 / 58.70
BERT+LSR (2020)	52.43 / 59.00	56.97 / 59.05
CorefRoBERTa (2020)	57.84 / 59.93	57.68 / 59.91
BERT Base Baseline	56.29 / 58.60	55.08 / 57.54
$SSAN_{Decomp}$	56.68 / 58.95	**56.06 / 58.41**
$SSAN_{Biaffine}$	**57.03 / 59.19**	55.84 / 58.16
BERT Large Baseline	58.11 / 60.18	57.91 / 60.03
$SSAN_{Decomp}$	58.42 / 60.36	57.97 / 60.01
$SSAN_{Biaffine}$	**59.12 / 61.09**	**58.76 / 60.81**
RoBERTa Base Baseline	57.47 / 59.52	57.27 / 59.48
$SSAN_{Decomp}$	58.29 / 60.22	**57.72** / 59.75
$SSAN_{Biaffine}$	**58.83 / 60.89**	57.71 / **59.94**
RoBERTa Large Baseline	58.45 / 60.58	58.43 / 60.54
$SSAN_{Decomp}$	59.54 / 61.50	59.11 / 61.24
$SSAN_{Biaffine}$	**60.25 / 62.08**	**59.47 / 61.42**
+ Adaptation	**63.76 / 65.69**	**63.78 / 65.92**

Adaptation?

DocRED의 distantly supervised set을 이용하여, SSAN을 사전 훈련한 다음 supervised set으로 fine-tuning

사전 훈련하는 과정이 새로운 파라미터가 사전 훈련된 Transformer 모델과의 적응하는 단계이기 때문에 adaptation이라고 표현

Results

Decomp, Biaffine 모두 baseline 대비 좋은 성능을 보여줌

adaptation은 아주 큰 성능 차이를 보여줌 (현재 SOTA)

adaptation을 제외한다면, 다른 논문에 비해서는 약한 성능 (ex. SIRE, ATLOP)

SSAN: Ablation Study

Dependency	Ign F1	F1
$\text{SSAN}_{\text{Biaffine}}$ (RoBERTa Large)	60.25	62.08
– *intra+coref*	59.59	61.57
– *intra+relate*	59.92	61.91
– *inter+coref*	59.87	61.74
– *inter+relate*	59.92	61.84
– *intraNE*	59.96	61.97
– all	58.45	60.58

Entity structure의 중요성

Entity structure를 어떻게 가져가는지에 따른 성능 비교
제안한 entity structure는 도움이 된다
intra+coref 가 가장 큰 영향이 있다

Bias Term	Ign F1	F1
RoBERTa Large baseline (w/o bias)	58.45	60.58
$+b_{s_{ij}}$	58.62	60.59
$+\boldsymbol{Q}_{s_{ij}}\boldsymbol{k}_j^T$	58.79	60.65
$+\boldsymbol{q}_i\boldsymbol{K}_{s_{ij}}^T$	59.26	61.31
$+\boldsymbol{q}_i\boldsymbol{K}_{s_{ij}}^T + \boldsymbol{Q}_{s_{ij}}\boldsymbol{k}_j^T + b_{s_{ij}}$	59.54	61.50
$+\boldsymbol{q}_i\boldsymbol{A}_{s_{ij}}\boldsymbol{k}_j^T$	59.83	61.75
$+\boldsymbol{q}_i\boldsymbol{A}_{s_{ij}}\boldsymbol{k}_j^T + b_{s_{ij}}$	60.25	62.08

Bias terms of two transformation modules

transformation module에서 bias를 어떻게 가져가는지에 따른 성능 비교
Decomposed: key conditioned bias가 query conditioned bias보다 더 좋은 결과를 보여준다

정리

Document-level Relation Extraction - 문서 내 서로 다른 entity 간 관계를 예측하는 task

GAIN (graph)
문서를 mention-level graph로 구축하여 각 entity에 대한 feature를 모델링하고, entity-level graph를 구축하여 각 entity pair에 대한 path 정보를 기반으로 예측하는 모델 제안

SIRE (graph)
두 mention 간의 관계가 같은 문장 내 존재하는 지 여부에 따라 서로 다른 모델링을 하여 예측
(top K words / mention-level graph)

ATLOP (transformer)
각 클래스 별 adaptive한 threshold 적용 및 각 entity pair 별 attention이 적용된 pooling

SSAN (transformer)
두 mention 간의 관계를 '같은 문장 내 존재'와 '같은 entity 지칭'으로 구분하여 Transformer 모델에 녹임